Published in Journals: Algorithms, AppliedMath, Computation and Symmetry

Topic Reprint

Intelligent Optimization Algorithm

Theory and Applications

Edited by
Shi Cheng, Chaomin Luo and Shangce Gao

mdpi.com/topics

MDPI

Intelligent Optimization Algorithm: Theory and Applications

Intelligent Optimization Algorithm: Theory and Applications

Topic Editors

Shi Cheng
Chaomin Luo
Shangce Gao

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Topic Editors

Shi Cheng
School of Computer Science
Shaanxi Normal University
Xi'an
China

Chaomin Luo
Department of Electrical and Computer Engineering
Mississippi State University
Mississippi State, MS
USA

Shangce Gao
Faculty of Engineering
University of Toyama
Toyama-shi
Japan

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Topic, published open access by the journals *Algorithms* (ISSN 1999-4893), *AppliedMath* (ISSN 2673-9909), *Computation* (ISSN 2079-3197) and *Symmetry* (ISSN 2073-8994), freely accessible at: https://www.mdpi.com/topics/50828CCC6Y.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-8839-9 (Hbk)
ISBN 978-3-7258-8840-5 (PDF)
https://doi.org/10.3390/books978-3-7258-8840-5

© 2026 by the authors. Articles in this reprint are Open Access and distributed under the Creative Commons Attribution (CC BY) license. The reprint as a whole is distributed by MDPI under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) license (https://creativecommons.org/licenses/by-nc-nd/4.0/).

Contents

About the Editors . . . . . **vii**

Xuan Liu, Chenyan Wang, Shan Jiang, Yutong Gao, Chaomurilige and Bo Cheng
Multi-Objective Cauchy Particle Swarm Optimization for Energy-Aware Virtual Machine Placement in Cloud Datacenters
Reprinted from: *Symmetry* **2025**, *17*, 742, https://doi.org/10.3390/sym17050742 . . . . . **1**

Xujiang Qin, Qi He, Xin Zhang and Xiang Yang
Data Value Assessment in Digital Economy Based on Backpropagation Neural Network Optimized by Genetic Algorithm
Reprinted from: *Symmetry* **2025**, *17*, 761, https://doi.org/10.3390/sym17050761 . . . . . **22**

Chenbo Shi, Delin Wang, Xiangyu Zhang, Chun Zhang, Jia Yan, Changsheng Zhu and Xiaobing Feng
Laser Stripe Segmentation Network Based on Evidential Uncertainty Theory Modeling Fine-Tuning Optimization Symmetric Algorithm
Reprinted from: *Symmetry* **2025**, *17*, 1280, https://doi.org/10.3390/sym17081280 . . . . . **52**

Chenyang Zhao, Guangyu Zhou, Junsheng Zhang, Zhijie Zhang, Gang Huang and Qianfang Xie
Dynamic Error Correction for Fine-Wire Thermocouples Based on CRBM-DBN with PINN Constraint
Reprinted from: *Symmetry* **2025**, *17*, 1831, https://doi.org/10.3390/sym17111831 . . . . . **71**

Fengsheng Jia, Tao Jin, Ruizhou Guo, Xinghua Yuan, Zihao Guo and Chengbing He
Intelligent Identification Method of Valve Internal Leakage in Thermal Power Station Based on Improved Kepler Optimization Algorithm-Support Vector Regression (IKOA-SVR)
Reprinted from: *Computation* **2025**, *13*, 251, https://doi.org/10.3390/computation13110251 . . . **98**

Ibrahim Aljarah, Abdullah Alzaqebah, Nailah Al-Madi, Ala' M. Al-Zoubi and Amro Saleh
mDA: Evolutionary Machine Learning Algorithm for Feature Selection in Medical Domain
Reprinted from: *Computation* **2025**, *13*, 292, https://doi.org/10.3390/computation13120292 . . . **124**

Tianlong Su, Linxin Zhang, Xuzhao Yuan, Xiaoquan Li, Xuefeng Li, Xuxing Huang, et al.
PS-InSAR Monitoring Integrated with a Bayesian-Optimized CNN–LSTM for Predicting Surface Subsidence in Complex Mining Goafs Under a Symmetry Perspective
Reprinted from: *Symmetry* **2025**, *17*, 2152, https://doi.org/10.3390/sym17122152 . . . . . **144**

Wilson Chango, Ana Salguero, Tatiana Landivar, Roberto Vásconez, Geovanny Silva, Pedro Peñafiel-Arcos, et al.
SARIMA vs. Prophet: Comparative Efficacy in Forecasting Traffic Accidents Across Ecuadorian Provinces
Reprinted from: *Computation* **2026**, *14*, 5, https://doi.org/10.3390/computation14010005 . . . . . **172**

Qingling He, Yifan Feng, Lin Ma, Xiaojuan Lu, Jiamei Zhang and Changxi Ma
Dynamic Interval Prediction of Subway Passenger Flow Using a Symmetry-Enhanced Hybrid FIG-ICPO-XGBoost Model
Reprinted from: *Symmetry* **2026**, *18*, 288, https://doi.org/10.3390/sym18020288 . . . . . **201**

Zhixin Wang, Shi Cheng, Yifei Sun, Sicheng Hou and Mingming Zhang
Solving Bilevel Multi-Robot Cooperative Path Planning Problems via a Memetic Framework
Reprinted from: *Symmetry* **2026**, *18*, 499, https://doi.org/10.3390/sym18030499 . . . . . **228**

Wei Gao, Bingnan Wu, Jianhua Liu and Daoming Tang
Flight Schedule Problem Optimization Based on Discrete Memory-Enhanced Restructured Particle Swarm Optimization Algorithm
Reprinted from: *Algorithms* **2026**, *19*, 233, https://doi.org/10.3390/a19030233 . . . . . **253**

Douglas Kwasi Boah, Suleman Abudu Fiele and Christian John Etwire
A New Algorithm for Finding Initial Basic Feasible Solutions of Transportation Problems
Reprinted from: *AppliedMath* **2026**, *6*, 58, https://doi.org/10.3390/appliedmath6040058 . . . . . **272**

Tian Yao, Yong Xu, Yue Ma, Hongtao Yan, Haihang Xu and An Wang
Object Re-Identification Method for Air-to-Ground Targets Based on Neighborhood Feature Centralization Attention
Reprinted from: *Computation* **2026**, *14*, 96, https://doi.org/10.3390/computation14050096 . . . . **308**

About the Editors

Shi Cheng

Shi Cheng received a bachelor's degree in Mechanical and Electrical Engineering from Xiamen University, Xiamen; a master's degree in Software Engineering from Beihang University (BUAA), Beijing, China; a Ph.D. degree in Electrical Engineering and Electronics from University of Liverpool, Liverpool, United Kingdom; and a Ph.D. degree in Electrical and Electronic Engineering from Xi'an Jiaotong-Liverpool University, Suzhou, China in 2005, 2008, and 2013, respectively. He is currently an associate professor in the School of Artificial Intelligence and Computer Science, Shaanxi Normal University, Xi'an, China. His current research interests include swarm intelligence, automated planning, and big data mining techniques, as well as their applications. He has authored more than 40 refereed papers, including those published in *IEEE Transactions on Evolutionary Computation and IEEE Transactions on Cybernetics*. Dr. Cheng serves as an Associate Editor for *Expert Systems with Applications*, *Computers & Industrial Engineering*, and other publications.

Chaomin Luo

Chaomin Luo received a Ph.D. degree in Electrical and Computer Engineering from the University of Waterloo, Canada, an M.Sc. degree in Engineering Systems and Computing from the University of Guelph, Canada, and a B.Eng. degree in Electrical Engineering from Southeast University, China. He is currently a Full Professor in the Department of Electrical and Computer Engineering at Mississippi State University. His research interests include computational intelligence, intelligent systems, robotics, autonomous systems, and AI-driven engineering education.

Dr. Luo serves as an Associate Editor for the *IEEE Transactions on Cognitive and Developmental Systems*, *Biomimetic Intelligence and Robotics* and *Intelligence and Robotics*. He has also served in numerous conference leadership positions, including Program Chair, Regional Program Chair, and General Co-Chair, and has organized more than 20 special sessions at leading international conferences. Dr. Luo is the recipient of multiple Best Paper Awards, including those from the *American Society for Engineering Education (ASEE)*, *Biomimetic Intelligence and Robotics*, *IEEE CARS*, *IEEE ICIA*, and the International Conference on Swarm Intelligence.

Shangce Gao

Shangce Gao received his Ph.D. degree in Innovative Life Science from the University of Toyama, Toyama, Japan, in 2011. He is currently a Professor in the Faculty of Engineering at the University of Toyama. His research interests focus on brain-inspired neural networks and their applications in medical diagnosis and drug discovery. He serves as an Associate Editor for several international journals, including *IEEE Transactions on Neural Networks and Learning Systems* and the *IEEE/CAA Journal of Automatica Sinica*.

symmetry

MDPI

Article

Multi-Objective Cauchy Particle Swarm Optimization for Energy-Aware Virtual Machine Placement in Cloud Datacenters

Xuan Liu [1,2], Chenyan Wang [1], Shan Jiang [1,*], Yutong Gao [1,2], Chaomurilige [1] and Bo Cheng [3]

[1] Key Laboratory of Ethnic Language Intelligent Analysis and Security Governance of MOE, Minzu University of China, Beijing 100081, China; liuxuan@muc.edu.cn (X.L.); 23302121@muc.edu.cn (C.W.); ytgao92@muc.edu.cn (Y.G.); chaomurilige@muc.edu.cn (C.)
[2] Key Laboratory of Big Data & Artificial Intelligence in Transportation, Ministry of Education (Beijing Jiaotong University), Beijing 100044, China
[3] School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100081, China; chengbo@bupt.edu.cn
* Correspondence: jshan@muc.edu.cn

Abstract: With the continuous expansion of application scenarios for cloud computing, large-scale service deployments in cloud data centers are accompanied by a significant increase in resource consumption. Virtual machines (VMs) in data centers are allocated to physical machines (PMs) and require the resources provided by PMs to run various services. Apparently, a simple solution to minimize energy consumption is to allocate VMs as compactly as possible. However, the above virtual machine placement (VMP) strategy may lead to system performance degradation and service failures due to imbalanced resource load, thereby reducing the robustness of the cloud data center. Therefore, an effective VMP solution that comprehensively considers both energy consumption and other performance metrics in data centers is urgently needed. In this paper, we first construct a multi-objective VMP model aiming to simultaneously optimize energy consumption, resource utilization, load balancing, and system robustness, and we then build a joint optimization function with resource constraints. Subsequently, a novel energy-aware Cauchy particle swarm optimization (EA-CPSO) algorithm is proposed, which implements particle asymmetric disturbances and an energy-efficient population iteration strategy, aiming to minimize the value of the joint optimization function. Finally, our extensive experiments demonstrated that EA-CPSO outperforms existing methods.

Keywords: energy efficient; Cauchy particle swarm optimization; cloud computing; multi-objective optimization; virtual machine placement

1. Introduction

1.1. Background

Recent years have witnessed the rapid development of cloud computing, which has been a hot topic in academia and industry [1,2]. A large number of cloud services bring incremental wide-area applications and economies of scale computing models [3], and more and more applications are deployed in cloud data centers [4]. The average size of data centers has also continued to expand. A cloud data center usually runs thousands of PMs with hardware resources such as CPU, bandwidth, and memory that are shared through virtualization technology to achieve scalable and resilient computing capabilities [5]. VMs operate independently on PMs, enabling efficient utilization and flexible management of physical resources. The VMP problem aims to allocate VMs to appropriate PMs based on

established objectives and constraints [6]. An efficient VMP strategy can ensure the stable and efficient operation of data centers [7] and has become a key research focus in the field of cloud computing.

1.2. Motivation

The issue of energy consumption in data centers is becoming increasingly severe. As early as 2010, data centers worldwide consumed as much as 201.8 terawatt-hours of electricity, an amount sufficient to power 19 million average U.S. households for a year, and this figure has continued to grow since then [8]. In addition, in order to meet user demands for quality of services (QoS) and cope with the dynamics and uncertainty of VM resource requirements [9], cloud service providers often adopt an over-provisioning strategy to ensure the stability and reliability of services [10]. However, this approach also brings problems of excess resources and energy waste. Moreover, energy consumption is directly linked to carbon emissions [11], and the increase in carbon emissions will have a significant negative impact on the global environment. Consequently, reducing the energy consumption of cloud data centers while ensuring system performance has become an urgent and essential task [4].

Beyond energy conservation, cloud providers must also pay attention to resource utilization and load balancing among PMs to ensure the performance of customer VMs [12], thereby enhancing overall user experience, market competitiveness, and operational efficiency. Additionally, the dynamic and stochastic characteristics of VM demand in cloud environments calls for robust VMP strategies that can maintain system performance and QoS even under load fluctuations, unexpected resource demands, or hardware failures [13]. As a result, designing more robust VMP strategies has become an increasing focus for cloud providers. Nevertheless, most existing research [13,14] on VMP problems primarily targets energy efficiency and resource utilization, with limited consideration given to the robustness of VM layout schemes.

1.3. Author's Contribution

Based on the aforementioned analysis, comprehensively considering those optimization objectives that are really concerned about practical applications and seeking the optimal solution for VMP is of great significance for service cluster deployment and cloud computing development. This study models the VMP problem as a constrained multi-objective optimization problem, taking into account the various resources within cloud data centers. To address this problem, we establish a novel multi-objective VMP optimization model and design a joint objective optimization function. We also propose the EA-CPSO algorithm to efficiently solve this optimization problem.

In summary, the contributions of this paper are as follows:

- Driven by practical application scenarios and in response to the diversity and complexity of resources within cloud data centers, we construct a multi-dimensional optimization model that is applicable to various cloud data centers. This model focuses on four optimization directions: firstly, reducing energy consumption through global VM regulation; secondly, achieving optimal load balancing by optimizing resource allocation and deployment strategies; thirdly, enhancing resource utilization efficiency based on resource consumption tracking during allocation processes; lastly, strengthening the robustness of the system to cope with various uncertainties and dynamic changes.
- Based on the existing PSO, EA-CPSO is proposed to have the ability to solve discrete optimization problems. The EA-CPSO algorithm implements a Cauchy mutation operation and an elite solution preservation operation to ensure that iterative individuals

tend to high-quality solutions while preventing the algorithm from converging to the local optimal. A Cauchy mutation leverages its heavy-tailed distribution property, allowing particles to have a higher probability of making large jumps, thereby increasing the likelihood of asymmetric transitions. The elite solution preservation mechanism retains the historically best solutions in each generation, preventing high-quality solutions from being disrupted or degraded by random perturbations or mutation operations during the population evolution process. Additionally, EA-CPSO demonstrates an ability to perceive changes in energy and resource allocation within the system, minimizing the value of the optimization function while ensuring all constraints are met.

- We developed a simulation platform for data center resource scheduling and designed a series of comparative experiments. The experimental results demonstrate that EA-CPSO outperforms existing approaches in solving multi-objective VM deployment optimization problems, while also validating the rationality and effectiveness of the proposed model in practical application scenarios.

1.4. Organization of the Paper

The structure of this paper is as follows: Section 2 reviews the related work. Section 3 describes the system model and problem statement. Section 4 focuses on the development of the algorithm, while Section 5 presents and analyzes the experimental results. Lastly, Section 6 concludes the paper with a summary.

2. Related Works

VMP is a research problem with various optimization objectives and has garnered significant attention in both industrial and academic fields. Numerous optimization algorithms have been developed to address this problem, aiming to find effective solutions. The following provides an overview of relevant research achievements in this field.

High-level comprehensive surveys on VMP problems have been discussed in [2,7]. The research conducted by Alashaikh et al. [2] emphasized the pivotal role of preferences in VMP decisions. They demonstrated how preferences influence the solutions to the VMP problem by conducting a detailed analysis of the specific applications of preferences in various scenarios, such as considering communication costs between VMs, resource utilization, and the specific needs of users or administrators. Study [7] conducted a comprehensive review and in-depth exploration of optimization methods for VMP and migration. This work encompassed approaches based on various optimization objectives such as cost, QoS, and energy efficiency, and discussed VMs management strategies in distributed and decentralized architectures.

Study [15] addressed the issues of task scheduling and load balancing in cloud computing by proposing two hybrid scheduling strategies based on genetic algorithms, one combining the genetic algorithm with First-Come-First-Serve, and the other with Round Robin. By integrating classical scheduling policies, the proposed approach enhances the quality of the initial population and accelerates the convergence of the genetic algorithm. Using real-world asymmetric workload traces, simulation experiments demonstrated the superiority of the proposed methods in terms of scheduling efficiency, resource utilization, and system throughput.

The work of [16] emphasized load balancing within cloud data centers, aiming to evenly allocate tasks and resources across all PMs to prevent any node from becoming overloaded or underutilized. It solved the problem of insufficient efficiency in reducing response time, energy consumption, and execution cost in existing load-balancing methods. This study proposed a load balancing model and used a hybrid optimization algorithm to solve it.

Study [17] primarily focused on the issue of energy consumption optimization in VM allocation within cloud data centers, particularly addressing the demands of Internet of Things (IoT) applications. It proposed a novel framework called AFED-EF, which classifies servers into five types by setting four thresholds and dynamically adjusts VM allocation and migration strategies based on these categories, aiming to jointly tackle the challenges of load fluctuations in VM configurations and enhance energy efficiency.

In their latest study [18], Hasanein D. Rjeib and Gabor Kecskemeti focused on optimizing energy consumption and minimizing resource wastage through reducing active PMs. They proposed models for energy consumption and resource wastage, defining PM energy usage as linearly proportional to CPU utilization. Additionally, they quantified the potential cost of unused resources within the servers and defined SLA violations as instances where the host fails to provide services to VMs during specific resource-request periods.

The research in [19] modeled the VMP problem as a multi-dimensional bin-packing task. Considering resources like CPU, memory, and disk, it designed a novel fitness function to optimize load balancing, resource utilization, and active PMs. The constraint of this problem ensures that the resource demands of VMs assigned to each PM remain within the PM's capacity limits.

The study in [20] addressed the VMP problems by formulating a multi-objective VMP model aimed at minimizing energy consumption while maximizing load balancing, resource utilization, and robustness. To tackle these challenges, the study proposed an improved algorithm named energy-efficient KnEA, which builds on the Knee point-driven evolutionary algorithm (KnEA) and incorporates a population initialization strategy focused on energy efficiency.

Ma et al. [21] proposed a PSO-based algorithm for joint network selection and service deployment, called PSO-JNSSP. In this algorithm, the network selection and service deployment of each particle are treated as potential solutions, and the speed and position of the particles are iteratively updated to converge to the optimal solution. Firstly, this study addressed the contradiction between the continuous updates of particle positions and the discrete nature of base station locations by introducing a transition probability mechanism, where a transition probability is defined based on the distance between the particle's calculated continuous pseudo-position and each base station as well as the current queue length of each base station, comprehensively considering both distance and load. Secondly, the fitness function was also improved by not simply optimizing delay or energy consumption alone but by combining both through weighted summation, allowing particles to balance task execution delay and energy consumption during the search process.

Study [22] proposed an innovative algorithm based on Quantum PSO to optimize service placement. The goal was to achieve optimal service placement while improving system throughput, energy consumption, delay, and load. Specifically, this paper improved particle representation by introducing the concept of quantum particles where each particle can represent multiple potential solutions simultaneously through quantum superposition, thereby enhancing the coverage of the search space. This paper also designed a double hashing technique to ensure that the encoding observed from the quantum state can be uniformly and effectively mapped to the actual edge node identifiers.

In [23], the service placement problem for IoT applications on fog resources was formulated as an optimization problem, with a hierarchical structure consisting of fog cells and fog orchestration control nodes. Additionally, the study introduced a heuristic approach based on the genetic algorithm (GA) to address the service deployment problem, and it evaluated the performance of the GA, the exact optimization method, and the greedy algorithm on the service deployment problem.

Study [24] proposed an Availability-aware Virtual Cluster Allocation algorithm based on Biogeography-based Optimization. In this study, the calculation methods for risk cost and total bandwidth usage were defined, and a joint optimization function was constructed for evaluating the performance of various virtual cluster allocation schemes.

Unlike previous works, we aimed to achieve the joint optimization of energy consumption, resource utilization, load balancing, and robustness in the VMP problem, addressing the limitations of most existing research that considers fewer than three objectives, which is often insufficient for many practical applications. Moreover, we improved existing models and used real-world test data as much as possible to ensure that the experimental results are more applicable to real scenarios. In terms of algorithm design, the proposed EA-CPSO algorithm effectively handles constrained multi-objective optimization problems and enhances population optimization through energy-aware Cauchy mutation.

3. System Models and Problem Formulation

In this section, we formulate the VMP problem as an NP-hard multi-objective optimization problem. Then, we construct branch models for the four optimization objectives mentioned earlier, precisely expressing these objectives through mathematical formulas. Based on this, we establish a multi-objective VMP optimization model designed to optimize all the set objectives simultaneously. For clarity in understanding the formulations in subsequent sections, Table 1 provides explanations for all the mathematical notations:

Table 1. Notations.

Symbol	Meaning
P, P_j	PMs and PM j
V, V_i	VMs and VM i
n	number of VMs
m	number of PMs in the data center
X	allocation scheme and X is a $m \times n$ binary matrix
x_{ij}	if the VM i is allocated in the PM j, and $x_{ij} \in \{0, 1\}$
C_j^k	resource k capacity of PM j, $k \in$ {CPU, mem, band}
D_i^k	resource k demand of VM i, $k \in$ {CPU, mem, band}
S	virtual machine placement scheme
U_j^k	resource k utilization of the sum hosted VMs, $k \in$ {CPU, mem, band}
$\overline{U^k}$	resource k average utilization
U	overall resource utilization ratio of S
$E\left(D_i^k\right)$	expectation of resource k demand
$\sigma\left(D_i^k\right)$	standard deviations of resource k demand
P'	the set of PMs already used in S
m'	number of PMs in the set P'
E	whole energy consumption of all PMs in the data center
E_j	energy consumption of the PM j
$U_j^{cpu}(t)$	CPU utilization as a function of time
RE_j^k	reserved resource k of PM j
W_j^k	weight of reserved resource space on PM j
R_j	total robustness measure of PM j
R	average of the robustness measures of all PMs (the robustness measure of S)
L	all PMs' resource load imbalance degree
f	optimization objective
$\gamma_1, \gamma_2, \gamma_3, \gamma_4$	variable weight coefficient
F	joint optimization objective (fitness function)

3.1. Problem Description

In cloud architecture, numerous PMs are interconnected through the network [25] and virtualized into multiple independent VM instances, each capable of running different applications or services [16]. The model of the VM placement problem, as illustrated in Figure 1, revolves around the rational allocation of VMs with varying resource requirements to their corresponding PMs. The objectives of this allocation include minimizing energy consumption and maximizing resource utilization, load balance, and robustness. PMs are equipped with multiple types of resources [16], which are requested by VMs based on their operational needs. Specifically, the main modules include the following:

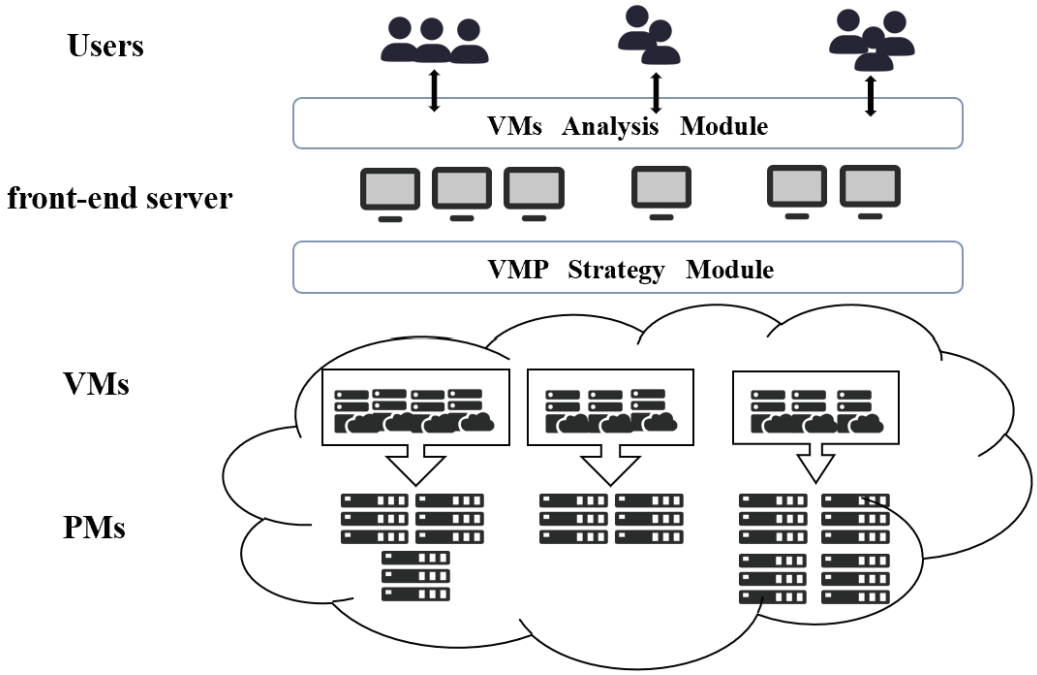


Figure 1. Virtual machine placement model.

- Users and frontend server: Cloud users submit requests to the frontend, which is the frontend server that handles these requests. The frontend server, used for processing client requests and providing user interfaces and services, plays a bridge role in connecting users and cloud services in cloud computing.
- VMs analysis module: This module determines the million instructions per second (MI/s) and memory for each VM request and calculates the number of active VMs during each time interval based on the frontend server.
- VMP strategy module: The role of this module is to determine the optimal strategy for deploying VMs in a cloud computing environment. It relies on a set of algorithms, rules, and policies to make decisions in order to meet user requirements, maximize resource utilization, and achieve high performance goals.
- VMs: The workload is deployed and executed on VMs. Based on the workload and resource requirements, VMs are optimally placed and allocated to PMs through placement algorithms.
- PMs: The infrastructure provides hardware support for provisioning virtualized resources to fulfill user requests.

3.1.1. Physical Machine

The set of PMs is represented as P, and P can be represented as $P = \{P_1, P_2, \cdots, P_m\}$, m being the number of available PMs. Each PM $P_j\left(P_j \in P\right)$ has a resource capacity vector

$C_j^k = \left\langle C_j^{cpu}, C_j^{mem}, C_j^{band} \right\rangle$, $j = 1, 2, ...m$, where C_j^{cpu}, C_j^{mem}, and C_j^{band} denote the CPU, memory, and bandwidth capacities of P_j, respectively.

3.1.2. Virtual Machine

The set of VMs is represented as V, and V can be represented as $V = \{V_1, V_2, \cdots, V_n\}$, n representing the number of VMs. Each VM $V_i (V_i \in V)$ is characterized by its resource demand vector $D_i^k = \left\langle D_i^{cpu}, D_i^{mem}, D_i^{band} \right\rangle$, $i = 1, 2, ...n$, where D_i^{cpu}, D_i^{mem}, and D_i^{band} indicate the resource requirements for the CPU, memory, and bandwidth of V_i, respectively.

Subsequently, the constraint on the PM's resource capacity (i.e., the demands D^k of the hosted VMs for each resource type should not exceed the resource capacity C^k of the host PM) is formulated as follows:

$$\begin{gathered} \sum_{i=1}^{n} D_i^k x_{ij} \leq C_j^k, \\ \forall i \in \{1, \cdots, n\}, \forall k \in \{cpu, mem, band\} \end{gathered} \tag{1}$$

3.1.3. Placement Relationship

The placement relationship between VMs and PMs can be depicted by a two-dimensional matrix X of size $m \times n$, as shown by Formulas (2) and (3):

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \tag{2}$$

$$x_{ij} = \begin{cases} 1 & \text{if } V_i \text{ is placed in } P_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where a_{ij} is a binary variable; $x_{ij} = 1$ indicates that V_i is allocated to P_j, whereas $x_{ij} = 0$ signifies that V_i is not allocated to P_j. The following condition ensures that each VM is assigned to no more than one PM:

$$\sum_{j}^{m} x_{ij} \leq 1, \forall i \in \{1, \cdots, n\} \tag{4}$$

Optimizing resource utilization reduces resource wastage and lowers the number of required PMs [26], thereby directly decreasing energy consumption and operating costs in data centers. According to [27], overall resource utilization is defined as the average utilization across different resource types, and the resource utilization vector $U_j^k = \left\langle U_j^{cpu}, U_j^{mem}, U_j^{band} \right\rangle$ of P_j is computed as follows:

$$U_j^{cpu} = \frac{\sum_{i}^{n} D_i^{cpu} \cdot x_{ij}}{C_j^{cpu}} \tag{5}$$

$$U_j^{mem} = \frac{\sum_{i}^{n} D_i^{mem} \cdot x_{ij}}{C_j^{mem}} \tag{6}$$

$$U_j^{band} = \frac{\sum_{i}^{n} D_i^{band} \cdot x_{ij}}{C_j^{band}} \tag{7}$$

where U_j^{cpu}, U_j^{mem}, and U_j^{band} denote the utilization ratios of the CPU, memory, and bandwidth for P_j. The average utilization ratio of each resource across all PMs can be calculated from the utilization ratio of each resource for every PM. The resource utilization of scheme S is then obtained, as illustrated in (8)–(11):

$$\overline{U^{cpu}} = \frac{1}{m'} \sum_{j \in P'} U_j^{cpu} \tag{8}$$

$$\overline{U^{mem}} = \frac{1}{m'} \sum_{j \in P'} U_j^{mem} \tag{9}$$

$$\overline{U^{band}} = \frac{1}{m'} \sum_{j \in P'} U_j^{band} \tag{10}$$

$$U = \left(\overline{U_{cpu}} + \overline{U_{mem}} + \overline{U_{band}}\right)/3 \tag{11}$$

In scheme S, the set P' denotes the utilized PMs and m' indicates the number of PMs within P'. The average utilization ratios of the CPU, memory, and bandwidth are represented by $\overline{U^{cpu}}$, $\overline{U^{mem}}$, and $\overline{U^{band}}$, respectively. Additionally, U denotes the overall resource utilization ratio for S.

3.2. Energy Consumption Model

According to most research [28], the CPU is the primary energy drain when compared to the other components. Consequently, the energy consumption of a PM is often represented by the CPU workload. The energy consumption of PM Pj is represented by Ej, while the total energy consumption of all PMs is denoted as E. We drew upon empirical data derived from SPECpower benchmarking [29] to inform our analysis. Table A1 shows the energy consumption of the HP ProLiant G4 and G5 servers under varying workload levels.

3.3. Robustness Measure

In data centers, different types of applications run on VMs. Given the inherent differences among these applications and the varying times and frequencies of user access, the requirements for computing resources, storage resources, and network resources by VMs show significant diversity and variability [25]. This dynamic and random nature of demand implies that the resource load of VMs will change dynamically accordingly [30].

In practical applications [31], deploying VMs based on peak-hour load conditions often leads to a significant underutilization of resources during off-peak hours, which leads to problems such as resource waste, cost escalation, and energy consumption increase. On the contrary, deploying VMs according to average load can reduce resource idleness but the performance requirements may not be met during specific periods, thereby compromising QoS and user experience [20]. Therefore, there is an urgent need for an efficient and robust VMP strategy that can flexibly address the dynamic fluctuations in resource load, ensuring a balance among effective cost and energy control, maintenance of high-level QoS, and user satisfaction.

Regarding the dynamic workload of VMs, we can calculate the expectation and standard deviation of a VM load based on historical log data. The expectation represents the average level of demand for a certain type of resource when the VM requests resources from the cloud data center. In contrast, the standard deviation reveals the degree of fluctuation in the VM's load around this average value, essentially quantifying the uncertainty of the load. Therefore, to enhance the robustness of VM placement strategies, we take the expected resource demand of VMs as the baseline while thoroughly considering the variation in

their load levels [16]. This approach ensures that sufficient buffer space is allocated during planning to maintain system stability, even under conditions of significant load fluctuations, thereby averting resource shortages or overload scenarios.

Furthermore, the likelihood of PM overload is related to both the reserved space on the PMs and the standard deviation of the resource loads of the VMs assigned to them [20]. Specifically, a larger reserved resource space corresponds to a reduced risk of PM overload, albeit potentially at the cost of decreased resource utilization efficiency. Conversely, inadequate reserved resource space, coupled with significant fluctuations in the resource loads of the VMs hosted, renders the PMs more susceptible to reaching an overloaded state. In summary, to accurately evaluate the robustness of a PM, it is necessary to comprehensively consider its reserved resource space and assign an appropriate weighting factor to it.

Based on the aforementioned concept, the robustness metrics for VMP schemes are constructed as follows:

$$RE_j^{cpu} = C_j^{cpu} - \sum_i E(D_i{}^{cpu}) x_{ij}, j = 1, 2, \cdots, m \tag{12}$$

$$RE_j^{mem} = C_j^{mem} - \sum_i E(D_i{}^{mem}) x_{ij}, j = 1, 2, \cdots, m \tag{13}$$

$$RE_j^{band} = C_j^{band} - \sum_i E\left(D_i{}^{band}\right) x_{ij}, j = 1, 2, \cdots, m \tag{14}$$

where REj^{cpu}, REj^{mem}, and REj^{band} denote the reserved resources for CPU, memory, and bandwidth, respectively. Let $\sigma(Di^{cpu})$, $\sigma(Di^{mem})$, and $\sigma\left(Di^{band}\right)$ represent the standard deviations of the resource demands for Vi. The weight of the reserved resource on Pj is then defined by Formulas (15)–(17):

$$W_j^{cpu} = \frac{\sum_i^n \sigma\left(D_i^{cpu}\right) x_{ij}}{\sum_i^n \sigma\left(D_i^{cpu}\right)} \tag{15}$$

$$W_j^{mem} = \frac{\sum_i^n \sigma(D_i^{mem}) x_{ij}}{\sum_i^n \sigma(D_i^{mem})} \tag{16}$$

$$W_j^{band} = \frac{\sum_i^n \sigma\left(D_i^{band}\right) x_{ij}}{\sum_i^n \sigma\left(D_i^{band}\right)} \tag{17}$$

Furthermore, the robustness measure for CPU, memory, and bandwidth can be defined as follows:

$$R_j^{cpu} = W_j^{cpu} \cdot RE_j^{cpu} = \frac{\sum_i^n \sigma\left(D_i^{cpu}\right) x_{ij}}{\sum_i^n \sigma\left(D_i^{cpu}\right)} \left(C_j^{cpu} - \sum_{i=1}^n E(D_i{}^{cpu}) x_{ij}\right) \tag{18}$$

$$R_j^{mem} = W_j^{mem} \cdot RE_j^{mem} = \frac{\sum_i^n \sigma(D_i^{mem}) x_{ij}}{\sum_i^n \sigma(D_i^{mem})} \left(C_j^{mem} - \sum_{i=1}^n E(D_i{}^{mem}) x_{ij} \right) \tag{19}$$

$$R_j^{band} = W_j^{band} \cdot RE_j^{band} = \frac{\sum_i^n \sigma\left(D_i^{band}\right) x_{ij}}{\sum_i^n \sigma\left(D_i^{band}\right)} \left(C_j^{band} - \sum_{i=1}^n E\left(D_i{}^{band}\right) x_{ij} \right) \tag{20}$$

The total robustness measure of P_j is defined in (21). The robustness measure of S is represented by the average of the robustness measures across all the PMs, as shown in (22):

$$R_j = R_j^{cpu} \cdot R_j^{mem} \cdot R_j^{band} \tag{21}$$

$$R = \frac{1}{m'} \sum_{j \in P'} R_j \tag{22}$$

3.4. Load Balance

Load balancing aims to ensure fair utilization and effective allocation of PM resources, thereby meeting the QoS requirements of users [9]. Considering CPU, memory, and bandwidth resources, we use L in this paper to measure the degree of resource load imbalance for PM P_i, as shown in formula (23):

$$L = \left(\sqrt{\sum_{j \in P'} \left(U_j^{cpu} - \overline{U^{cpu}} \right)^2} + \sqrt{\sum_{j \in P'} \left(U_j^{mem} - \overline{U^{mem}} \right)^2} + \sqrt{\sum_{j \in P'} \left(U_j^{band} - \overline{U^{band}} \right)^2} \right) / 3 \tag{23}$$

3.5. VMP Problem Model

Based on the above analysis, we model the optimization model of the VMP problem as follows:

$$min : f_1(X) = E \tag{24}$$

$$min : f_2(X) = c - R \tag{25}$$

$$min : f_3(X) = 1 - U \tag{26}$$

$$min : f_4(X) = L \tag{27}$$

Expressions (24)–(27) define four optimization objectives, including minimizing the energy consumption of physical machines, achieving optimal robustness, maximizing resource utilization, and minimizing load variance. Specifically, in Expression (25), the symbol c introduced represents a constant term. Notably, for the purpose of normalizing the model, a fixed constant is subtracted from robustness and resource utilization, shifting their optimization direction towards minimization. Since there is a trade-off between the above four optimization objectives, we measure a combined metric called $F(X)$, as the formula of (28) and (29) shows, where $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ are the variable weight coefficient:

$$\min : F(X) = \gamma_1 f_1 + \gamma_2 f_2 + \gamma_3 f_3 + \gamma_4 f_4 \tag{28}$$

$$s.t. \quad Constraints\,(1)\text{–}(4) \tag{29}$$

4. Algorithm

In this section, we propose an improved algorithm based on PSO, namely EA-CPSO. The improvements focus on two aspects. Firstly, an efficient particle encoding method is designed based on the characteristics of the VMP problem. Secondly, an energy-saving-oriented particle mutation and population iteration strategy is proposed. Table 2 complements related notations about the algorithm description:

Table 2. Notations.

Symbol	Meaning
X	position of particle
V	velocity of particle
i	currently accessed particle
k	current iteration number
ω	inertia weight
ξ_1, ξ_2, ξ	random variables lie in $(0, 1)$
C_1	individual cognitive factor
C_2	social cognitive factor
$pbest$	individual historical optimal solution
$gbest$	global optimal solution
ω_{max}	maximum inertia weight
ω_{min}	minimum inertia weight
K	maximum number of iterations
I	number of particles
p	a probability of particle mutation
η	tunable weight
$\mathcal{C}$	standard Cauchy-distributed random value
X_{max}	upper bound of the particle's position
X_{min}	lower bound of the particle's position

4.1. Particle Swarm Optimization

Particle swarm optimization (PSO) [21] is a swarm intelligence-based optimization algorithm, which simulates the behavior of biological organisms in nature that collaborate within a group to locate food. Each particle's position represents a solution, and its velocity determines its direction and speed of movement within the search space. The particles iteratively adjust their velocities and positions to seek out more optimal solutions. At the k iteration, the velocity of particle i is given by v_i^k and the position is denoted by x_i^k. The position update is formulated as shown in Formula (30), while the velocity update is given by Formula (31):

$$X_i^{k+1} = X_i^k + V_i^k \tag{30}$$

$$V_i^{k+1} = \omega V_i^k + c_1 \cdot \xi_1 (pbest_i^k - X_i^k) + c_2 \cdot \xi_2 (gbest^k - X_i^k) \tag{31}$$

where x_i^{k+1} is the position of particle i of the $k+1$ iteration, and where V_i^{k+1} is the velocity of particle i of the $k+1$ iteration. is a local optimal solution; $pbest_i^k$ represents the individual historical optimal solution of particle i up to the k iteration, while $gbest^k$ represents the global optimal solution of the population up to the k iteration; c_1 represents the individual cognitive factor, while c_2 represents the social cognitive factor; ξ_1, ξ_2 represent random variables that lie in $(0, 1)$. Specifically, ω is the inertia weight used to control the detection capabilities of the algorithm. The expression for ω is shown in Formula (32). Here, ω_{max}

denotes the maximum inertia weight, ω_{min} represents the minimum inertia weight, and K signifies the maximum number of iterations.

$$\omega = \omega_{max} - (\omega_{max} - \omega_{min})\frac{k}{K} \tag{32}$$

Given the importance of the inertia weight, we aim for particles to exhibit differentiated search characteristics at various stages of the algorithm. Therefore, we employ a linearly decreasing inertia weight throughout the iterations. Specifically, as the number of iterations increases, ω gradually decreases. This design allows the population to maintain strong global exploration capabilities during the initial stages, which helps it to avoid becoming trapped in the local optimal. In the later stages of iterations, the particles' local exploitation abilities are enhanced, facilitating more refined searches and thereby improving the convergence rate of the algorithm.

4.2. Cauchy Mutation

It is difficult to solve complex optimization problems effectively using standard PSO algorithms. To overcome this limitation, many researchers have incorporated mutation strategies into the standard PSO [21,22], with Gaussian and Cauchy mutations being the primary mutation operators introduced. Figure 2 illustrates the comparison between Gaussian and Cauchy density functions [32]. It can be observed that the Gaussian distribution exhibits a characteristic bell-shaped curve with a narrow peak and rapid decay. In contrast, although the Cauchy distribution also has a high peak near the center, its tails decay more slowly, resulting in a broader distribution range. More importantly, whereas the Gaussian distribution introduces symmetric disturbances, the Cauchy distribution introduces asymmetric disturbances through its long tail, thereby increasing the probability of particles undergoing asymmetric transitions. Consequently, particles mutated with the Cauchy distribution possess stronger exploration capabilities.

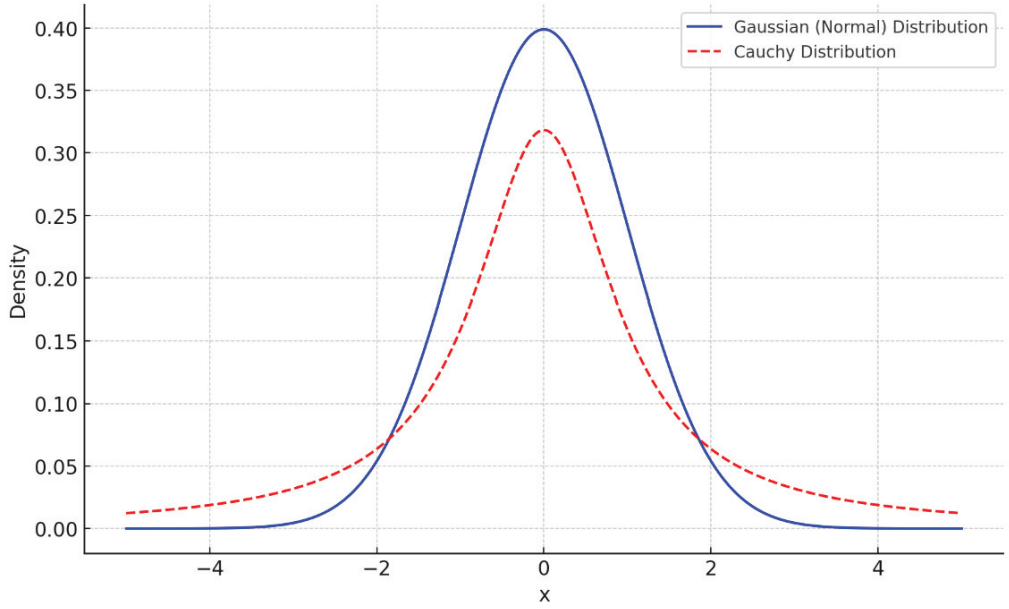


Figure 2. Comparison of Gaussian and Cauchy density functions.

We propose the EA-CPSO algorithm, which applies the Cauchy mutation to selected particles with a certain probability p. This approach aims to enhance the diversity of the

swarm and reduce the risk of premature convergence to the local optimal. The Cauchy mutation process for X_i is implemented as follows:

$$X_i = X_i(1 + \eta \cdot \mathcal{C}) \tag{33}$$

$$\mathcal{C} = tan((\xi - 0.5) \cdot \pi) \tag{34}$$

$$X_i = \begin{cases} X_{max} & X_i > X_{max} \\ X_i & X_{main} < X_i < X_{max} \\ X_{min} & X_i < X_{min} \end{cases} \tag{35}$$

where η is the tunable weight and ξ is a random value between 0 and 1. $\mathcal{C}$ represents a random perturbation value drawn from a standard Cauchy distribution with a location parameter of 0 and a scale parameter of 1; while max and min signify the upper and lower bounds, respectively, for the particle position variable.

4.3. Implementation Scheme of EA-CPSO Algorithm

In Section 3, the objective of the VMP optimization is identifying an optimized placement solution S. Consequently, each particle retains a set of solutions during each iteration, and VMP subsequently determines the optimal mapping of VMs to PMs through comparison. Furthermore, in practical scenarios, cloud service providers offer a limited number of VM types [10]. Inspired by this, the particle encoding proposed in this chapter solely considers the mapping of VM types (rather than VM instances) to PMs. The VM scheduling sequence is illustrated in Figure 3. In the figure, each black rectangle within the solid box on the left represents a PM, while each white rectangle within the dashed box on the right represents a VM. The number in the bottom-right corner of each rectangle indicates the ID of the corresponding PM or VM. The arrows in the figure represent the mapping relationships between VMs and PMs. Taking the topmost arrow as an example, the PM with ID 1 is associated with VMs with ID 3 and ID 5, meaning that VMs V_3 and V_5 are allocated to P_1:

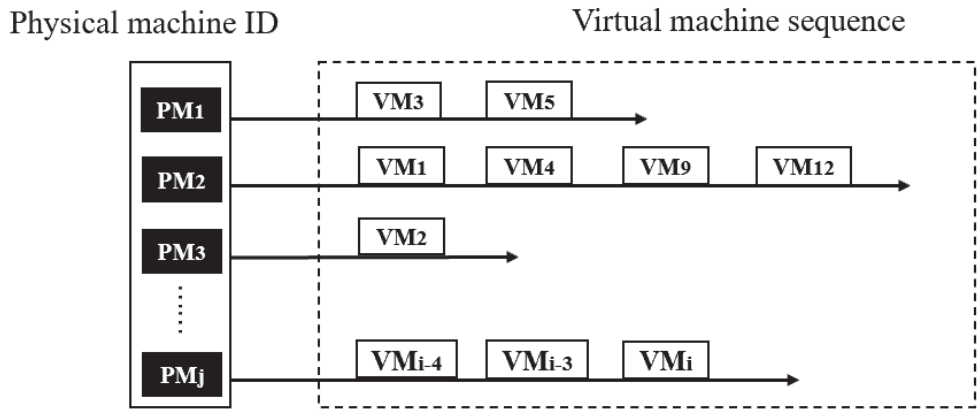


Figure 3. Placement solution example.

Figure 4 illustrates the search mechanism of the PSO algorithm. Each particle represents a potential solution, and its movement is guided by both its own search history and the collective experience of the swarm. The position of a particle is updated according to three components: the inertia component, which maintains its previous motion direction; the cognitive component, which drives it toward its personal best position; and the social component, which directs it toward the global best position. By combining these three

driving forces, particles achieve a balance between exploration and exploitation within the search space, continuously updating their positions based on individual and global experiences and gradually approaching the optimal solution.

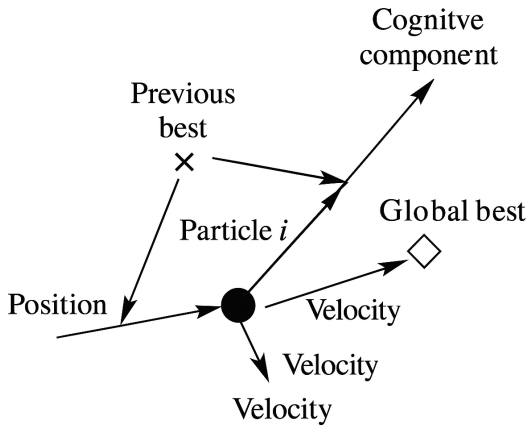


Figure 4. Schematic diagram of the PSO search mechanism.

As shown in Algorithm 1, we present the implementation scheme of the VMP approach using EA-CPSO. Lines 1 to 2 are initialization; we define K as the maximum number of iterations, and we define I as the number of particles; then, the optimization loop begins. During iteration, the velocity and position of each particle are updated based on its *pbest* and *gbest* values. Particles are sorted according to their fitness, and a subset is selected for Cauchy mutation, aiming to enhance diversity and avoid getting trapped in the local optimal. After each iteration, the global optimal solution is updated, and, ultimately, the algorithm returns the optimal position and fitness value.

Algorithm 2 elaborates on the VM allocation process within Algorithm 1. For each particle in Algorithm 1, Algorithm 2 handles the assignment of VMs to PMs by evaluating available resources and updating the allocation scheme. Specifically, for each VM, it checks if a PM has sufficient resources; if not, it performs a linear search for a suitable PM and assigns the VM accordingly. Essentially, Algorithm 2 serves as the resource allocation mechanism invoked by Algorithm 1 for each particle, ensuring feasible and optimized VM-to-PM allocation within the EA-CPSO framework.

Algorithm 1 VMP approach with EA-CPSO

1: Initialize particles with random X and V
2: Initialize F, *pbest*, *gbest*, and other parameters
3: **for** $k = 1$ to K **do**
4: **for** $i = 1$ to I **do**
5: Update V_i based on pbest and gbest by the Formula (31)
6: Update X_i based on the V_i by the Formula (30)
7: Compute $F(X_i)$ of particle by Formulas (24) and (29)
8: **if** $F(X_i) < F(pbest_i)$ **then**
9: Set $pbest_i$ to a copy of X_i
10: **end if**
11: **end for**
12: Sort particles by F, select particles with probability p
13: **for** each selected particle **do**
14: Apply Cauchy mutation by Formulas (33)–(35)
15: update X
16: **end for**
17: update *gbest*
18: **end for**
19: **return** *gbest* and $F(gbest)$

Algorithm 2 VM allocation

```
Initialize PMs and V
Initialize particles, scheme S, and other parameters
for each particle do
    update S based on available resources by Formulas (1) and (4)
    for each VM in VMs do
        if PM resources are insufficient then
            perform linear search for an available PM
        end if
        Assign VM to PM
    end for
end for
return S
```

5. Experimental Evaluations

5.1. Experiment Settings

In all our experiments, the following scenarios were used. Forty PMs were provided for the placement of 50 VMs. Typically, cloud data centers offer a variety of PMs to meet diverse requirements [29]. Meanwhile, the standard deviation of the VM resource requests was randomly set to reflect their diversity. Each host had a capacity of two CPU cores with MIPS values of either 1880 or 2660, 4 GB of RAM, and 1 TB of storage space. Power models were used from either the HP ProLiant ML110 G4 or HP ProLiant ML110 G5, and these models indicated the energy consumption of the hosts at various utilization rates, as shown in Table A1. For the VMs configurations [33], we considered four types of VMs with MIPS ratings of 250, 500, 1000, and 1250, with the number of each type being randomly generated. Detailed specifications for both the hosts and the VMs can be found in Table 3:

Table 3. Host/VM types and capacity.

Name	CPU (MIPS)	Cores	Memory	Bandwidth	Storage
PM Type 1	1.86 GHz	4	4 GB	1 Bbit/s	1 TB
PM Type 2	2.66 GHz	4	4 GB	1 Bbit/s	1 TB
VM Type 1	1.25 GHz	1	0.6 GB	100 MBbit/s	1 GB
VM Type 2	1.0 GHz	1	1.7 GB	200 MBbit/s	1 GB
VM Type 3	0.5 GHz	1	3.75 GB	100 MBbit/s	1 GB
VM Type 4	0.25 GHz	1	0.85 GB	200 MBbit/s	1 GB

The parameter values of the EA-CPSO algorithm are as follows: the population size is 40; the number of iterations is 60 or 120; the Cauchy mutation probability is 0.2; the individual cognitive factor is 1.8; and the social cognitive factor is 2.2. The stopping criterion of the algorithm is reaching the predefined maximum number of iterations. We used a weighted summation method to calculate the joint optimization objective function [21], as shown in Formula (28). The weight coefficients for each indicator were set as follows: $\gamma_1 = 0.0004$, $\gamma_2 = 0.02$, $\gamma_3 = 1$, $\gamma_4 = 0.5$. The design of the weight coefficients aimed to normalize the value ranges of each optimization indicator to the $[0, 1]$ interval, thereby avoiding bias due to dimensional differences and ensuring the comparability of the objectives during the optimization process.

To assess the effectiveness of our approach, we performed experiments comparing the EA-CPSO algorithm with these alternative methods.

- RFF [24]: The RFF (random first-fit) algorithm is a typical greedy algorithm. Its core idea is to randomly select a VM and attempt to assign it to the first PM that satisfies the constraints.

- MGGA [23]: The MGGA (multi-objective grouping genetic) algorithm is an optimization method based on the GA. It utilizes the global search capability of the GA, combined with the characteristics of grouping problems, to gradually approach the Pareto optimal solution set through population evolution.
- PSO [21]: a heuristic optimization algorithm based on population intelligence, which has been introduced in Section 4.1.

5.2. Experimental Results and Evaluation

5.2.1. Comparison of Joint Optimization Result

In this paper, we define the optimization objective as $F(X)$ (as shown in Equations (28) and (29)). The fitness function $F(X)$ comprehensively considers four objectives: robustness, resource utilization, energy consumption, and load balancing. By employing a weighted summation method, the multi-objective optimization problem is converted into a single-objective optimization problem. This approach improves computational efficiency and provides a unified evaluation standard for comparing different algorithms.

Figure 5 illustrates the joint optimization results of four algorithms (PSO, RFF, MGGA, and EA-CPSO) under 60 and 120 iterations. It is evident that EA-CPSO demonstrated superior optimization performance under both conditions. Specifically, its joint optimization result was approximately 2.90 after 60 iterations and further reduced to 2.87 after 120 iterations, showcasing the algorithm's efficiency in handling joint optimization problems. Taking the 60-iteration case as an example, EA-CPSO outperformed PSO by 14.66%, RFF by 33.82%, and MGGA by 16.18%.

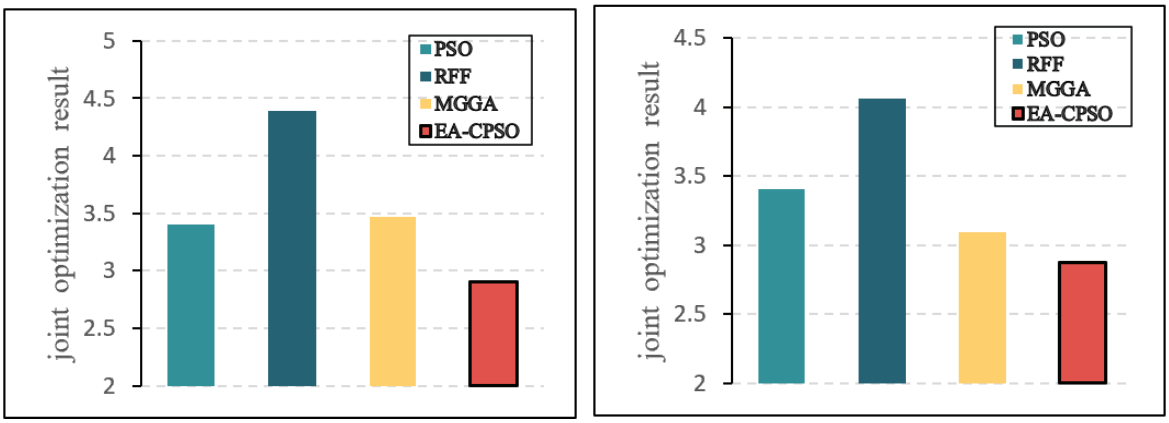


Figure 5. Joint optimization results comparison for 60 and 120 iterations.

The results indicate that EA-CPSO performs consistently well in both fewer and more iterations, making it highly suitable for efficiently solving complex optimization problems. MGGA exhibited notable improvement with increased iterations, indicating it to be more suitable for scenarios requiring higher computational resources. In contrast, PSO and RFF showed relatively lower optimization capabilities, with RFF, in particular, demonstrating limitations in addressing such problems effectively.

The experimental data have been summarized in Appendix A. In addition to the three comparison experiments discussed above, we also included the artificial bee colony (ABC) algorithm, the tree seed algorithm (TSA), and the knee point-driven evolutionary algorithm (KnEA) for further comparison, in order to more comprehensively validate the effectiveness of the proposed method. Among them, ABC simulates the foraging behavior of bees; STA models the seed dispersal mechanism of trees for both local and global search; and KnEA addresses multi-objective optimization by identifying knee points on the Pareto front.

5.2.2. Comparison of Convergence

To verify the convergence performance of different algorithms during the optimization process, this study plotted the convergence curves of EA-CPSO and other alternative algorithms within 60 iterations, as shown in Figure 6. We highlighted the convergence points in red to better illustrate the convergence behavior of the algorithm. Specifically, since the RFF algorithm does not have the concepts of population size and iteration, its values were directly used for comparison:

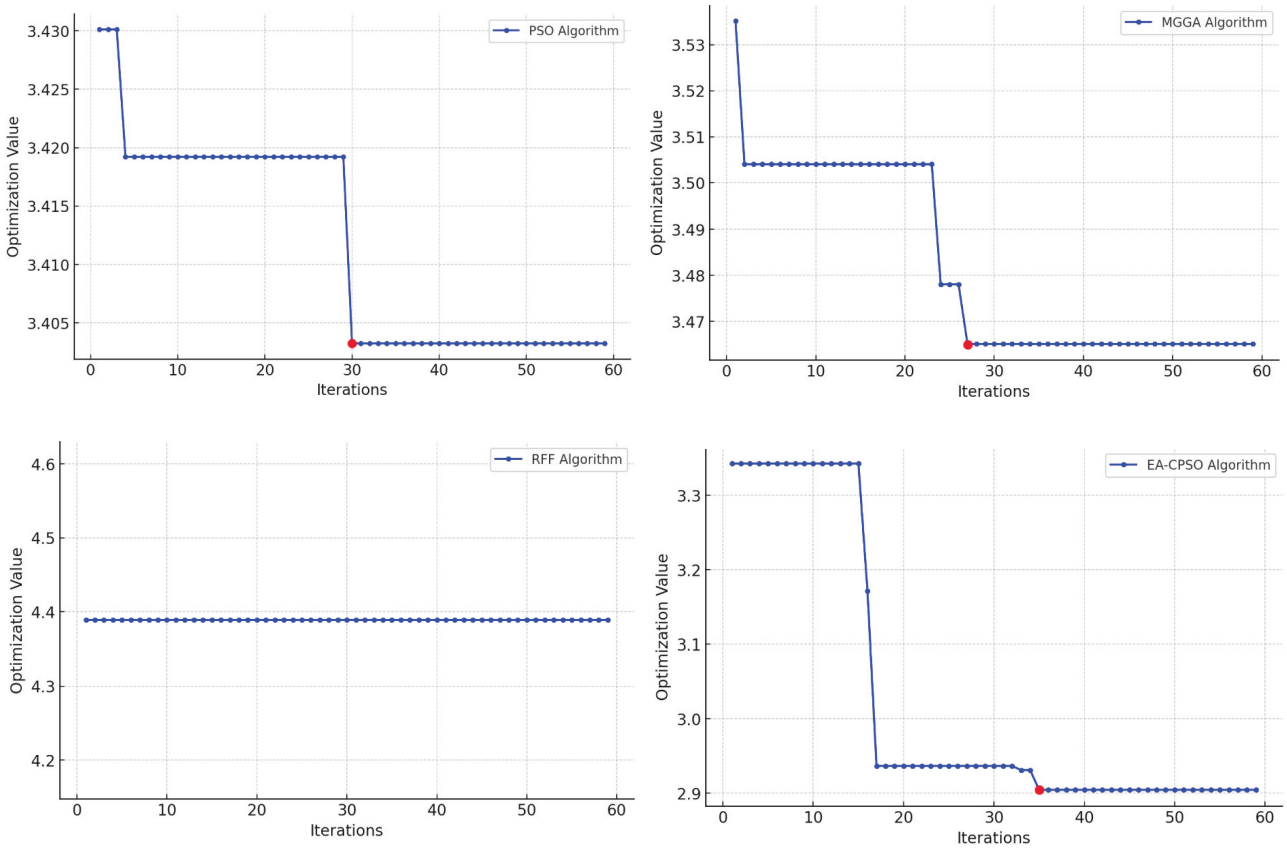


Figure 6. Algorithms convergence comparison (PSO, MGGA, RFF, and EA-CPSO).

The curves show that EA-CPSO demonstrated significant advantages in both optimization efficiency and final results. It exhibited a rapid convergence trend in the early stages of iteration, with the optimization value significantly decreasing within the first 20 iterations and stabilizing after about 30 iterations. The Cauchy mutation significantly enhanced EA-CPSO's jumping ability in the global search phase, effectively overcoming the limitation of PSO, which tends to get trapped in the local optimal.

In contrast, the other algorithms showed certain limitations. While MGGA gradually converged to a better result in the later stages of optimization, its convergence speed in the early iterations was noticeably slower than that of EA-CPSO, leading to lower optimization efficiency. PSO showed a fast decline in optimization value during the initial iterations, but due to the lack of an effective global search mechanism it eventually became stuck in the local optimal. RFF, as a classic greedy approximation algorithm, was mainly used as a reference in this experiment, and its optimization performance was relatively weak.

5.2.3. Comparison of Different Algorithms for Each Objective

To further compare the performance of different algorithms in multi-objective optimization, this paper conducted a quantitative analysis of the optimization values for each

objective in the different algorithms and presents the results in the bar charts shown in Figure 7:

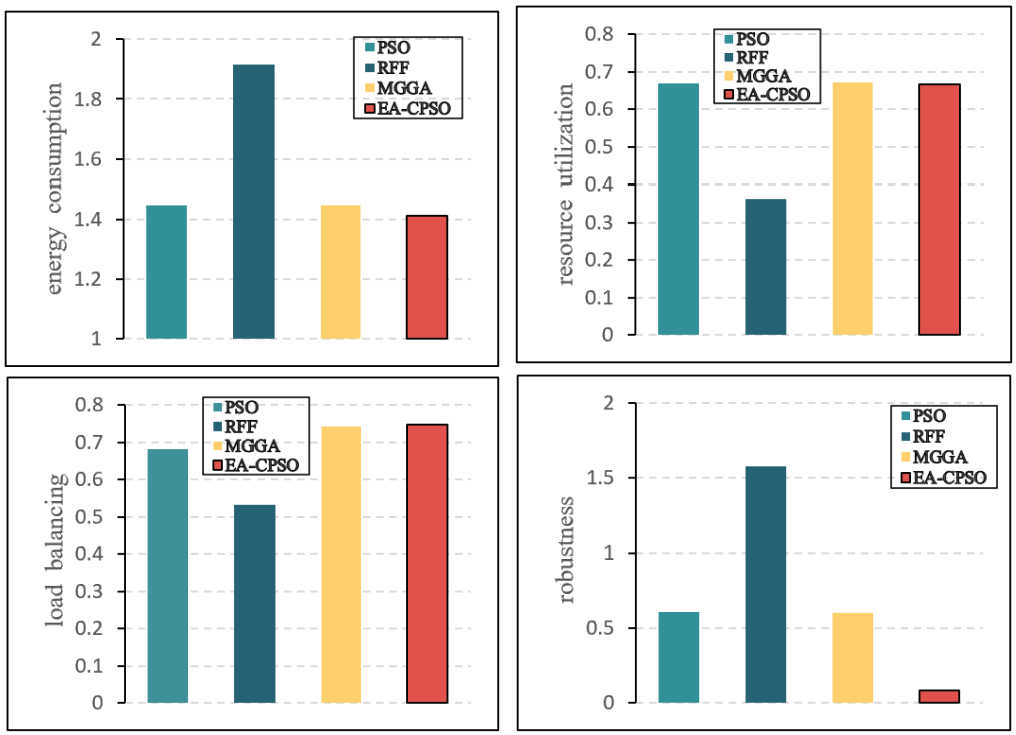


Figure 7. Each optimization objective value comparison by different algorithms.

Firstly, evaluated for energy consumption and robustness, EA-CPSO significantly outperformed the other algorithms, demonstrating its advantages in global optimization and resource allocation balance. However, EA-CPSO did not excel in the objective of load balancing. On the one hand, EA-CPSO achieved greater optimization for the other objectives but sacrificed the adequate optimization of load balancing. On the other hand, this was related to the diversity and imbalance in the resource demands of the VMs and the resource capacities of the PMs in the experimental environment. Additionally, although RFF performed well for certain individual objectives, its lack of global search capability led to notable shortcomings in multi-objective trade-offs.

6. Conclusions

In this paper, we focused on the VMP problem in data centers and proposed a multi-objective placement model that comprehensively considers energy consumption, resource utilization, load balancing, and robustness. Based on this model, a joint optimization objective function was constructed, and a population-based heuristic algorithm, EA-CPSO, was developed to efficiently address the problem. By incorporating an energy-aware Cauchy mutation strategy, EA-CPSO significantly enhances the global search capability of the population. Subsequently, EA-CPSO was compared with existing algorithms, and our experimental results demonstrate that it outperforms other methods, achieving better trade-offs among all optimization objectives.

EA-CPSO achieves a dynamic balance between exploration and exploitation within the search space by integrating the inertia component, the cognitive component, and the social component, while introducing a mutation mechanism. This makes it suitable for solving complex problems such as multi-objective optimization and constrained optimization. The mutation mechanism leverages the heavy-tailed property of the Cauchy distribution, enabling particles to perform large-scale jumps with a higher probability and thereby

enhancing the global search capability. The elite solution preservation mechanism ensures that the best solutions in each generation are retained, improving the overall optimization performance. However, EA-CPSO exhibits sensitivity to parameter settings in practical applications. To address this issue, future research will consider incorporating an adaptive parameter control mechanism to further enhance the robustness and optimization performance of EA-CPSO across different application scenarios.

Author Contributions: Conceptualization, X.L., C.W. and B.C.; methodology, X.L. and B.C.; software, C.W.; validation, Y.G.; formal analysis, C.; investigation, Y.G.; resources, X.L.; data curation, S.J.; writing—original draft preparation, X.L. and C.W.; writing—review and editing, C.W.; visualization, C.; supervision, B.C.; project administration, S.J. and B.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Grant of Open Project of Key Laboratory of Ethnic Language Intelligent Analysis and Security Governance, Ministry of Education (NO. OPC-202401) and Foundation of Key Laboratory of Big Data & Artificial Intelligence in Transportation (Beijing Jiaotong University), Ministry of Education (No.BATLAB202403)

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1. Energy consumption in watts at varying CPU utilization levels.

Server	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
G4	86	89.4	92.6	96	99.5	102	106	108	112	114	117
G5	93.7	97	101	105	110	116	121	125	129	133	135

Table A2. Data from experimental results.

Algorithms	PSO	RFF	MGGA	EA-CPSO	ABC	TSA	KnEA
$F(X)$	3.403	4.389	3.465	2.904	3.893	3.357	3.176
$\gamma_1 f_1$	1.448	1.916	1.448	1.412	1.442	1.447	1.479
$\gamma_2 f_2$	0.604	1.576	0.603	0.078	1.022	0.521	0.303
$\gamma_3 f_3$	0.669	0.36	0.671	0.667	0.667	0.670	0.666
$\gamma_4 f_4$	0.682	0.533	0.743	0.747	0.763	0.719	0.729

References

1. Al Hwaitat, A.K.; Fakhouri, H.N. Multi-Cloud Security Optimization Using Novel Hybrid JADE-Geometric Mean Optimizer. *Symmetry* **2025**, *17*, 503. [CrossRef]
2. Alashaikh, A.; Alanazi, E.; Al-Fuqaha, A. A Survey on the Use of Preferences for Virtual Machine Placement in Cloud Data Centers. *ACM Comput. Surv.* **2022**, *54*, 1–39. [CrossRef]
3. Liu, S.; Han, W.; Zhang, Z.; Chan, F.T.S. An analysis of performance, pricing, and coordination in a supply chain with cloud services: The impact of data security. *Comput. Ind. Eng.* **2024**, *192*, 110237. [CrossRef]
4. Hosseini Shirvani, M.; Ramzanpoor, Y. Multi-objective QoS-aware optimization for deployment of IoT applications on cloud and fog computing infrastructure. *Neural Comput. Appl.* **2023**, *35*, 19581–19626. [CrossRef]
5. Jin, H.; Jiang, C.; Lv, S. A Hybrid Whale Optimization Algorithm for Quality of Service-Aware Manufacturing Cloud Service Composition. *Symmetry* **2023**, *16*, 46. [CrossRef]
6. Feng, H.; Zhou, T.; Deng, Y.; Yang, L.T. A Holistic Energy-Aware and Probabilistic Determined VMP Strategy for Heterogeneous Data Centers. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 1852–1866. [CrossRef]
7. Silva Filho, M.C.; Monteiro, C.C.; Inácio, P.R.M.; Freire, M.M. Approaches for optimizing virtual machine placement and migration in cloud environments: A survey. *J. Parallel Distrib. Comput.* **2018**, *111*, 222–250. [CrossRef]

8. Hussain, M.; Wei, L.-F.; Rehman, A.; Hussain, A.; Ali, M.; Javed, M.H. An electricity price and energy-efficient workflow scheduling in geographically distributed cloud data centers. *J. King Saud Univ.—Comput. Inf. Sci.* **2024**, *36*, 102170. [CrossRef]
9. Herrera, J.L.; Berrocal, J.; Forti, S.; Brogi, A.; Murillo, J.M. Continuous QoS-aware adaptation of Cloud-IoT application placements. *Computing* **2023**, *105*, 2037–2059. [CrossRef]
10. Li, Y.; Yao, X.; Wei, S.; Xiao, W.; Yin, Z. Cloud Manufacturing Service Composition Optimization Based on Improved Chaos Sparrow Search Algorithm with Time-Varying Reliability and Credibility Evaluation. *Symmetry* **2024**, *16*, 772. [CrossRef]
11. Tripathi, S.; Kumar, P.; Gupta, P.; Misra, R.; Singh, T.N. Workload Shifting Based on Low Carbon Intensity Periods: A Framework for Reducing Carbon Emissions in Cloud Computing. In Proceedings of the 2023 IEEE International Conference on Big Data (BigData), Sorrento, Italy, 15–18 December 2023; pp. 3387–3395.
12. Regaieg, R.; Koubàa, M.; Ales, Z.; Aguili, T. Multi-objective optimization for VM placement in homogeneous and heterogeneous cloud service provider data centers. *Computing* **2021**, *103*, 1255–1279. [CrossRef]
13. Hu, L.; Wu, X.; Che, X. HICA: A Hybrid Scientific Workflow Scheduling Algorithm for Symmetric Homogeneous Resource Cloud Environments. *Symmetry* **2025**, *17*, 280. [CrossRef]
14. Absardi, Z.N.; Javidan, R. IoT traffic management using deep learning based on osmotic cloud to edge computing. *Telecommun. Syst.* **2024**, *87*, 419–435. [CrossRef]
15. Naz, I.; Naaz, S.; Agarwal, P.; Alankar, B.; Siddiqui, F.; Ali, J. A Genetic Algorithm-Based Virtual Machine Allocation Policy for Load Balancing Using Actual Asymmetric Workload Traces. *Symmetry* **2023**, *15*, 1025. [CrossRef]
16. Geetha, P.; Vivekan, S.J.; Yogitha, R.; Jeyalakshmi, M.S. Optimal load balancing in cloud: Introduction to hybrid optimization algorithm. *Expert Syst. Appl.* **2024**, *237*, 121450. [CrossRef]
17. Zhou, Z.; Shojafar, M.; Alazab, M.; Abawajy, J.; Li, F. AFED-EF: An Energy-Efficient VM Allocation Algorithm for IoT Applications in a Cloud Data Center. *IEEE Trans. Green Commun. Netw.* **2021**, *5*, 658–669. [CrossRef]
18. Rjeib, H.D.; Kecskemeti, G. VMP-ER: An Efficient Virtual Machine Placement Algorithm for Energy and Resources Optimization in Cloud Data Center. *Algorithms* **2024**, *17*, 295. [CrossRef]
19. Wen, C.; Jiang, W. Research on Virtual Machine Layout Strategy Based on Improved Particle Swarm Optimization Algorithm. In Proceedings of the 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Zhangjiajie, China, 10–12 August 2019; pp. 1343–1349.
20. Ye, X.; Yin, Y.; Lan, L. Energy-Efficient Many-Objective Virtual Machine Placement Optimization in a Cloud Computing Environment. *IEEE Access* **2017**, *5*, 16006–16020. [CrossRef]
21. Ma, S.; Song, S.; Zhao, J.; Zhai, L.; Yang, F. Joint Network Selection and Service Placement Based on Particle Swarm Optimization for Multi-Access Edge Computing. *IEEE Access* **2020**, *8*, 160871–160881. [CrossRef]
22. Bey, M.; Kuila, P.; Naik, B.B.; Ghosh, S. Quantum-inspired particle swarm optimization for efficient IoT service placement in edge computing systems. *Expert Syst. Appl.* **2024**, *236*, 121270. [CrossRef]
23. Skarlat, O.; Nardelli, M.; Schulte, S.; Borkowski, M.; Leitner, P. Optimized IoT service placement in the fog. *Serv. Oriented Comput. Appl.* **2017**, *11*, 427–443. [CrossRef]
24. Liu, J.; Wang, S.; Buyya, R.; Yang, F. Availability-Aware Virtual Cluster Allocation in Bandwidth-Constrained Datacenters. *IEEE Trans. Serv. Comput.* **2020**, *13*, 425–436. [CrossRef]
25. Shanmugam, V.; Ling, H.-C.; Gopal, L.; Eswaran, S.; Chiong, C.W.R. Network-aware virtual machine placement using enriched butterfly optimisation algorithm in cloud computing paradigm. *Clust. Comput.* **2024**, *27*, 8557–8575. [CrossRef]
26. Ni, W.; Zhang, Y.; Li, W.W. An Optimal Strategy for Resource Utilization in Cloud Data Centers. *IEEE Access* **2019**, *7*, 158095–158112. [CrossRef]
27. Liu, X.; Cheng, B.; Wang, S. Availability-Aware and Energy-Efficient Virtual Cluster Allocation Based on Multi-Objective Optimization in Cloud Datacenters. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 972–985. [CrossRef]
28. Dasgupta, G.; Sharma, A.; Verma, A.; Neogi, A.; Kothari, R. Workload management for power efficiency in virtualized data centers. *Commun. ACM* **2011**, *54*, 131–141. [CrossRef]
29. Farahnakian, F.; Ashraf, A.; Pahikkala, T.; Liljeberg, P.; Plosila, J.; Porres, I.; Tenhunen, H. Using Ant Colony System to Consolidate VMs for Green Cloud Computing. *IEEE Trans. Serv. Comput.* **2015**, *8*, 187–198. [CrossRef]
30. Seddiki, D.; Galán, S.G.; Expósito, J.E.M.; Ibañez, M.V.; Marciniak, T.; de Prado, R.J.P. Sustainable expert virtual machine migration in dynamic clouds. *Comput. Electr. Eng.* **2022**, *102*, 108257. [CrossRef]
31. Clarkson, J.; Fumero, J.; Papadimitriou, M.; Xekalaki, M.; Kotselidis, C. Towards practical heterogeneous virtual machines. In Proceedings of the Conference Companion of the 2nd International Conference on Art, Science, and Engineering of Programming, Nice, France, 9–12 April 2018; Marr, S., Sartor, J.B., Eds.; ACM: New York, NY, USA, 2018; pp. 46–48.

32. Mlotshwa, T.; van Deventer, H.; Bosman, A.S. Cauchy Loss Function: Robustness Under Gaussian and Cauchy Noise. In *Artificial Intelligence Research*; Pillay, A., Jembere, E., Gerber, A., Eds.; Springer Nature: Cham, Switzerland, 2022; pp. 123–138.
33. Liu, X.; Cheng, B.; Wang, S.; Chen, J. Joint Availability Enhancement and Traffic Optimization of Virtual Cluster Allocation in Cloud Datacenters. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 1554–1567. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

symmetry

MDPI

Article

Data Value Assessment in Digital Economy Based on Backpropagation Neural Network Optimized by Genetic Algorithm

Xujiang Qin [1], Qi He [2,3], Xin Zhang [1,*] and Xiang Yang [2,3,*]

[1] Guangxi Information Center, Nanning 530221, China
[2] College of Computer Science and Engineering, Guilin University of Technology, Guilin 541006, China
[3] Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, Guilin 541004, China
* Correspondence: zhangxin@gxi.gov.cn (X.Z.); 1994022@glut.edu.cn (X.Y.)

Abstract: As a new form of economic activity driven by data resources and digital technologies, the digital economy underscores the strategic significance of data as a core production factor. This growing importance necessitates accurate and robust valuation methods. Data valuation poses core modeling challenges due to its nonlinear nature and the instability of neural networks, including gradient vanishing, parameter sensitivity, and slow convergence. To overcome these challenges, this study proposes a genetic algorithm-optimized BP (GA-BP) model, enhancing the efficiency and accuracy of data valuation. The BP neural network employs a symmetrical architecture, with neurons organized in layers and information transmitted symmetrically during both forward and backward propagation. Similarly, the genetic algorithm maintains a symmetric evolutionary process, featuring symmetric operations in both crossover and mutation. The empirical data used in this study are sourced from the Shanghai Data Exchange, comprising 519 data samples. Based on this dataset, the model incorporates 9 primary indicators and 21 secondary indicators to comprehensively assess data value, optimizing network weights and thresholds through the genetic algorithm. Experimental results show that the GA-BP model outperforms the traditional BP network in terms of mean squared error (*MSE*), root mean squared error (*RMSE*), mean absolute error (*MAE*), and coefficient of determination (R^2), achieving a 47.6% improvement in prediction accuracy. Furthermore, GA-BP exhibits faster convergence and greater stability. When compared to other models such as long short-term memory (LSTM), convolutional neural networks (CNNs), and optimization-based BP variants like particle swarm optimization BP (PSO-BP) and whale optimization algorithm BP (WOA-BP), GA-BP demonstrates superior generalization and robustness. This approach provides valuable insights into the commercialization of data assets.

Keywords: genetic algorithm; value evaluation; BP neural network; optimization model

1. Introduction

In the era of rapid digital economic growth, data have become a central production factor, profoundly transforming economic growth models and social governance systems. China's "14th Five-Year Plan for Digital Economy Development" and the "Opinions on Establishing a Data Infrastructure System to Better Leverage the Role of Data Elements" (commonly known as the "Data Twenty Articles") explicitly highlight the need to accelerate the marketization of data elements and refine data value assessment mechanisms to unlock

the economic and social potential of data assets. The core modeling challenge in this process lies in addressing the nonlinearity of data and the sensitivity of neural networks to initial parameters, which complicates the accurate quantification of data's unstructured value. Moreover, the rapid advancement of artificial intelligence has greatly facilitated the in-depth exploration of data elements, playing a crucial role in data analysis, intelligent forecasting, and value assessment. Despite these advancements, effectively measuring and evaluating the value of data remains a significant challenge in the ongoing process of data marketization.

Existing research on data value assessment primarily follows two approaches. The first approach adapts conventional intangible asset valuation methods—such as the income approach [1], market approach [2], and cost approach [3]—to account for the unique characteristics of data. Wang et al. reviewed current data valuation methodologies, highlighting the importance of incorporating data's natural increment, externalities, and multidimensional attributes. They also emphasized the need to consider spillover effects in data circulation and evaluate value based on specific application scenarios [4]. Jonathan argued that traditional valuation methods should account for market dynamics as external economic conditions significantly influence data valuation [5]. Xu Xianchun et al. refined the cost approach to improve its applicability in data accounting [6], while Lei et al. enhanced the income approach to increase evaluation accuracy. Among these methods, the cost approach is generally preferred due to its practicality and lower operational complexity, whereas the market and income approaches are more susceptible to external market fluctuations [7]. Yang classified the enterprise data assetization process into three stages—accumulation, application, and assetization—applying the income approach to assess case enterprises [8]. Zhang et al. [9] suggested that the income approach better reflects the profitability of data, yielding results that are widely accepted. Meanwhile, Lu et al. [10] proposed mitigating the limitations of the traditional cost approach by incorporating precise market premium estimations and accurate replacement cost calculations. Lin contended that most studies on traditional valuation methods overlook practical applicability, advocating for the adoption of modern evaluation techniques instead [11].

The second approach to data valuation involves developing specialized derivative models. For instance, Longstaff et al. introduced the least squares Monte Carlo (LSM) model, which integrates least squares regression with the Monte Carlo algorithm for valuation purposes [12]. Zuo et al. proposed replacing the analytic hierarchy process with a multidimensional preference linear programming method to enhance data valuation accuracy [13]. Despite these advancements, no consensus has been reached within the academic community regarding a standardized methodology for data value assessment. Most studies remain anchored in intangible asset valuation frameworks while advocating for the selection of evaluation approaches tailored to the specific attributes of the data [14].

Recent advancements in artificial intelligence have introduced new perspectives on data value assessment. Researchers have increasingly explored the integration of machine learning and deep learning techniques with traditional valuation methods to optimize assessment models and improve predictive accuracy. For example, D. Niyato et al. incorporated the Sternberg model from economics with machine learning classification algorithms to evaluate data utility from a data science perspective [15]. Zhang proposed a deep learning-based model for analyzing both intrinsic and extrinsic data value, validating its effectiveness using production data from port enterprises [16]. Wang et al. developed a fuzzy evaluation model for big data value assessment, employing artificial neural networks to determine indicator weights and applying fuzzy comprehensive evaluation to derive final valuation results [17]. Additionally, Ren constructed a data evaluation system leveraging ensemble machine learning techniques to quantify data value [18]. Wang further

highlighted the capability of artificial neural networks to objectively assess data application value, mitigating the subjectivity and variability inherent in manual evaluations [19]. Yan's research demonstrated that machine learning-based models outperform traditional multiple linear regression methods, with the random forest model exhibiting the highest accuracy in data valuation [20].

Building on these developments, deep learning-based neural network models have gained prominence in data value assessment. Among them, the backpropagation (BP) neural network [21], recognized for its strong nonlinear mapping capabilities and adaptability, has demonstrated significant advantages in data prediction and analysis.

The backpropagation (BP) algorithm has demonstrated significant potential in data valuation applications, particularly in optimizing computational efficiency and enhancing predictive accuracy. For example, Huo et al. [22] implemented a three-layer BP neural network to classify retail stores for fresh food sales forecasting. Their model exhibited accelerated convergence rates and robust resistance to data redundancy and noise during Matlab simulations, achieving superior accuracy in short-term stock price predictions. Further showcasing its versatility, Chen et al. [23] employed BP neural networks during the COVID-19 pandemic to predict user suitability for online education platforms, achieving a classification accuracy of 77.5%. Similarly, Kalinić et al. [24] demonstrated the BP algorithm's predictive superiority over traditional linear models in analyzing mobile commerce consumer attitudes, highlighting its enhanced ability to capture complex behavioral patterns.

However, traditional BP neural networks exhibit inherent limitations in practical applications, such as a susceptibility to local optima and a hypersensitivity to initial weight-threshold configurations, all of which reduce model stability and predictive reliability. Feng et al. [25] emphasized the algorithm's critical reliance on initial weights and biases, showing that improper initialization directly degrades valuation accuracy. Deng et al. [26] identified fundamental weaknesses in BP neural networks for elemental prediction in aquatic systems, noting that their tendency toward local minima restricts global optimization potential, thus limiting practical applicability despite satisfactory internal validation performance. To address these limitations, researchers have proposed alternative architectures. For instance, an RBF (radial basis function) neural network model was introduced to overcome the predictive deficiencies of conventional BP networks. Wang et al. [27] systematically analyzed the inherent contradictions between BP's learning rate and stability, as well as the absence of systematic methods for determining optimal hidden-layer neuron quantities. Similarly, Ye et al. [28] confirmed BP's convergence instability and structural ambiguity while implementing a modified LM-BP (Levenberg–Marquardt backpropagation) neural network to reduce model error rates through enhanced optimization mechanisms.

To address these challenges, researchers have increasingly integrated metaheuristic optimization algorithms with backpropagation neural networks (BPNNs) in recent years. Notable advancements include hybrid architectures that combine BPNNs with a sparrow search algorithm (SSA) [29], a whale optimization algorithm (WOA) [30], a particle swarm optimization (PSO) [31], and genetic algorithms (GAs) [32]. These integrations enable BPNNs to overcome the constraints of local optima and achieve global optimization capabilities. Among these approaches, the genetic algorithm (GA) has proven particularly effective as a global search mechanism for optimizing the initial weights and thresholds of BPNNs. This GA-BPNN hybridization effectively resolves issues related to instability and local optima entrapment during system training, thereby enhancing both convergence efficiency and predictive performance [33]. When applied to data valuation modeling, the GA-BPNN framework significantly improves prediction efficiency while eliminating the subjective biases inherent in traditional fitting processes, thus enhancing the objectivity

and reliability of results [34]. Zhang et al. [35] optimized a BP neural network using an improved genetic algorithm (GA) to model the relationship between weld appearance and molten pool shadow features. Through experiments conducted at welding speeds of 4.5 m/min and 3 m/min, the predictive performance of the model was validated, with the mean absolute percentage error (MAPE) for weld height and width remaining below 4.95%, 4.81%, 5.3%, and 1.4%, respectively. These results demonstrate the model's high accuracy and stability. Song et al. [36] proposed a BP neural network prediction model integrated with the AdaBoost algorithm to improve prediction accuracy. The average mean absolute error (*MAE*) decreased from 17,760.1 in the standard BP model to 5,230.6 in the AdaBoost_BP model, representing an approximate 70.5% reduction in error. Liang et al. [37] developed a GA-optimized BP neural network (GA-BP) model to predict the effects of polymer fissure grouting. Compared with the conventional BP model, the GA-BP model improved the coefficient of determination (R^2) from 0.975 to 0.991 while reducing *MAE* by 56.6% and root mean square error (*RMSE*) by 40.4%, indicating significantly enhanced predictive accuracy and fitting performance. Zhang et al. [38] applied the GA-BP neural network to improve the energy efficiency assessment of crude oil gathering and transportation systems. The GA-BP model increased the R^2 for energy utilization efficiency, thermal energy utilization efficiency, and electrical energy utilization efficiency by 1.87%, 0.56%, and 1.37%, respectively. Additionally, the R^2 for comprehensive energy consumption, gas consumption, and electricity consumption improved by 2.63%, 0.77%, and 0.31%, respectively, demonstrating that the GA-BP model significantly enhanced prediction accuracy while reducing computational workload. Li et al. [39] employed an interpretable GA-BP strategy to optimize the prediction of interactions at the membrane–sludge particle interface. The mean squared errors (*MSEs*) for three types of interactions were reduced to 1.0816×10^{-6}, 5.0089×10^{-9}, and 9.0432×10^{-7}, respectively, with the maximum regression coefficient (R) reaching 0.99990 and prediction errors controlled within 0.01%, effectively improving the accuracy of membrane fouling quantification. Li et al. [40], based on construction engineering data from Guangdong Province, proposed a GA-BP model for project cost prediction. The R^2 of the testing set increased from 0.87 to 0.94, and the *RMSE* decreased from 1907.203 to 1281.422, significantly enhancing prediction accuracy and stability.

This study represents the first application of a GA-optimized BP neural network in data value assessment. A comprehensive data value evaluation framework was established based on different "dataset" types, consisting of 9 primary indicators—data quality, data accessibility, data coverage, data diversity, data adaptability, data volume, technical capability, market factors, and data source—and 21 secondary indicators, including producer reputation, supplier qualifications, data quality certification, timeliness, and update speed, among the 21 indicators. The model designed in this study incorporates symmetrical structural features, which contribute to the stability and generalization ability of data value prediction. The GA-BP model effectively optimizes network weights and thresholds, accelerating convergence and improving prediction accuracy while substantially reducing mean squared error (*MSE*), root mean squared error (*RMSE*), and mean absolute error (*MAE*). Simultaneously, it enhances the determination coefficient (R^2) [41]. Furthermore, comparative analysis of predicted and actual values, alongside error distribution visualizations, thoroughly validated the advantages of GA-BP in terms of convergence speed, prediction accuracy, and model stability. Experimental results demonstrate that the GA-BP model effectively handles data complexity and heterogeneity, outperforming BP neural networks, long short-term memory (LSTM), convolutional neural networks (CNNs), random forest (RF), support vector machines (SVMs), particle swarm optimization BP (PSO-BP), whale optimization algorithm BP (WOA-BP), and sparrow search algorithm BP (SSA-BP) in both prediction accuracy and model robustness.

2. Materials and Methods

2.1. BP Neural Network

The backpropagation neural network (BPNN) is a multilayer feedforward neural network trained using the error backpropagation algorithm. It consists of an input layer, hidden layers, and an output layer. The BP neural network is characterized by a symmetrical structure where the number of neurons gradually increases and decreases around the hidden layers, forming a symmetric topology. The signal transmission from input to output and the error propagation from output to input also form a symmetrical process. The core principle of a BPNN is to minimize the error between the network's output and the desired output by adjusting the network's weights and thresholds through gradient descent. The BP algorithm iteratively optimizes the model through two phases: forward propagation to compute the output and backward propagation to correct the error. It is widely used in classification, regression, and nonlinear modeling tasks [21].

To better illustrate the training process of a BP neural network, we take a three-layer BP neural network structure shown in Figure 1 as an example. The training process of a BP neural network generally consists of two main stages: the forward computation of errors and backpropagation.

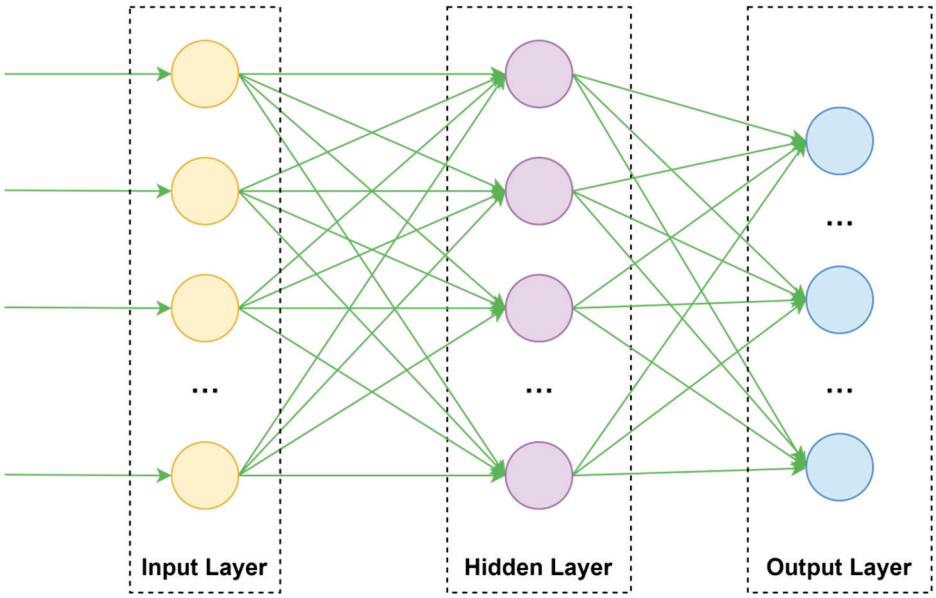


Figure 1. BP neural network structure.

In the forward computation phase, the output of the hidden layer H_j is first calculated based on the input feature vector X_i. Then, the output layer O_k is computed using H_j. Finally, the prediction error E_k of the BP neural network is determined by comparing the computed output O_k with the expected output Y_k. The mathematical formulations for the hidden layer and output layer are given as follows:

$$H_j = f_1(\sum_{i=1}^{n} W_{ij} X_i - U_j) \tag{1}$$

$$O_k = f_2(\sum_{j=1}^{n} H_j W_{jk} - V_k) \tag{2}$$

$$E_k = |O_k - Y_k| \tag{3}$$

where H_j represents the output of the j-th hidden-layer neuron, f_1 and f_2 are the activation function where f_1 is 'tansig' for the hidden layer and f_2 is 'purelin' for the output layer, and W_{ij} denotes the weight between the i-th input neuron and the j-th hidden-layer neuron.

X_i corresponds to the input data of the i-th input neuron, while U_j represents the bias term associated with the j-th hidden-layer neuron. Similarly, O_k signifies the output of the k-th output layer neuron, W_{jk} represents the weight between the j-th hidden neuron and the k-th output neuron, and V_k denotes the bias term of the k-th output neuron.

After obtaining the prediction error of the neural network, the backpropagation phase is initiated to update the weights between the input layer and the hidden layer (W_{ij}) and those between the hidden layer and the output layer (W_{jk}), as well as the thresholds of the hidden layer (U_j) and the output layer (V_k). The specific process can be expressed as follows:

$$W_{ij} = W_{ij} + \alpha H_j(1 - H_j) X_i \sum_{k=1}^{m} W_{jk} Y_k \tag{4}$$

$$W_{jk} = W_{jk} + \alpha H_j Y_k \tag{5}$$

$$U_j = U_j + \alpha H_j(1 - H_j) X_i \sum_{k=1}^{m} W_{jk} Y_k \tag{6}$$

$$V_k = V_k + \alpha Y_k \tag{7}$$

where α was empirically determined to be 0.01 after testing several values (e.g., 0.1, 0.01, 0.001) to ensure model convergence and generalization performance.

During training, forward computation and backpropagation iterate continuously until the *MSE* requirement is met or the maximum number of epochs is reached. The *MSE* is propagated back in the form of gradients to adjust the weights and biases of neurons in each layer. The genetic algorithm, inspired by Darwin's natural selection mechanism, theoretically ensures the continuous optimization of the BP neural network's weights and biases throughout the training process.

2.2. GA-BP Neural Network

Traditional BP neural networks suffer from slow convergence, a tendency to fall into local optima, and high sensitivity to initial weights and thresholds, all of which significantly impact model stability and prediction accuracy. To address these limitations, genetic algorithms (GAs), known for their efficiency and strong global optimization capabilities, are introduced to optimize the initial parameters of the BP neural network (BPNN). The genetic algorithm (GA) is a global optimization technique inspired by the principles of biological evolution, first proposed by Holland in 1975. Its core mechanism mimics natural selection, crossover, and mutation to explore the solution space and identify optimal solutions. Through iterative population evolution, GA progressively converges toward the global optimum of complex problems, making it particularly suitable for nonlinear, high-dimensional, and multi-modal optimization tasks.

The construction of the GA-BP data value evaluation model can be summarized as follows:

(1) In this study, individuals in the population are encoded using a real number encoding scheme. Each consists of a string of real numbers representing the neuron thresholds and inter-layer connection weights of the BP neural network, forming the initial population. The BPNN is then utilized to predict data value, with the minimization of mean squared error (*MSE*) on both training and testing sets serving as the fitness function. The fitness of each individual in the initial population is calculated using the following equation:

$$MSE_{train} = \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} (y_{train,i} - \hat{y}_{train,i})^2 \tag{8}$$

$$MSE_{test} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (y_{test,i} - \hat{y}_{test,i})^2 \tag{9}$$

$$Fitness = \frac{1}{2}(MSE_{train} + MSE_{test}) \tag{10}$$

where MSE_{train} and MSE_{test} represent the mean squared errors of the training and test sets, respectively; $y_{train,i}$ and $y_{test,i}$ denote the actual values of the training and test sets; $\hat{y}_{train,i}$ and $\hat{y}_{test,i}$ correspond to the predicted values; and N_{train} and N_{test} indicate the number of samples in the training and test sets, respectively. The fitness function, $Fitness$, is a comprehensive measure of the prediction errors on both the training and testing datasets. By minimizing $Fitness$ the algorithm optimizes the BP neural network's fitting ability on the training set and its generalization performance on the testing set, thereby preventing overfitting.

(2) The selection process in the genetic algorithm employs the roulette wheel selection method, where the selection probability of each chromosome is proportional to its fitness. Chromosomes with higher fitness values have a greater likelihood of being selected. The selection probability is calculated using the following equation:

$$P(c_i) = \frac{Fitness(c_i)}{\sum_{i=1}^{n} Fitness(c_i)} \tag{11}$$

where $P(c_i)$ represents the selection probability of chromosome c_i; $Fitness(c_i)$ denotes the fitness value of chromosome c_i; and n is the total number of chromosomes in the population.

(3) The crossover operation in the genetic algorithm is performed using a two-point crossover operator. Two crossover points are randomly selected and the gene segments between these points are exchanged. This process can be described as follows: Suppose we have two parent chromosomes, c_k and c_l, represented as follows:

$$c_k = [g_1, g_2, g_3, \ldots, g_m] \tag{12}$$

$$c_l = [h_1, h_2, h_3, \ldots, h_m] \tag{13}$$

① Randomly selecting crossover points: Two crossover points, p_1 and p_2, are randomly chosen such that $1 \leq p_1 < p_2 \leq m$;

② Exchanging gene segments: The gene segments between p_1 and p_2 in the parent chromosomes c_k and c_l are swapped to generate offspring chromosomes.

As a result, the parent chromosomes c_k and c_l exchange gene segments between p_1 and p_2, producing the new offspring chromosomes c'_k and c'_l, which can be represented as follows:

$$c'_k = \left[g_1, g_2, \ldots, g_{p_i-1}, h_{p_1}, h_{p_1+1}, \ldots, h_{p_2-1}, g_{p_2}, g_{p_2+1}, \ldots, g_m\right] \tag{14}$$

$$c'_l = \left[h_1, h_2, \ldots, h_{p_i-1}, g_{p_1}, g_{p_1+1}, \ldots, g_{p_2-1}, h_{p_2}, h_{p_2+1}, \ldots, h_m\right] \tag{15}$$

where the gene segments between the crossover points p_1 and p_2 are exchanged, while the remaining segments remain unchanged.

(4) The mutation operation is carried out using the Gaussian mutation operator, which allows the gene values of an individual to be randomly perturbed within a specified range. This enhances the diversity of the genetic algorithm and improves the convergence quality. The mutation operation for the j-th gene a_{ij} of the i-th individual is performed as follows:

$$a'_{ij} = a_{ij} + \mathcal{N}\left(0, \sigma^2\right) \tag{16}$$

where $\mathcal{N}(0, \sigma^2)$ represents a Gaussian-distributed random variable with a mean of 0 and a variance of σ^2; σ denotes the mutation amplitude, which is typically dynamically

adjusted during iterations to ensure the algorithm's balance between global search and local optimization capabilities.

After performing the key operations of roulette wheel selection, two-point crossover, and Gaussian mutation in the genetic algorithm, a new population is generated and the individual fitness values are recalculated according to the fitness function. If the fitness of the optimal individual satisfies the termination condition, this best individual is selected and its corresponding initial weight and threshold values are adopted as the initial parameters for the BP neural network. The optimized BP neural network is then used to train the model, calculate the prediction errors for both the training and testing datasets, and evaluate the model using the mean squared error (*MSE*). This iterative optimization process continues until the error meets the specified threshold or the maximum number of iterations is reached, at which point the optimal BP neural network model is obtained. It is worth noting that the two-point crossover operator symmetrically exchanges gene segments between parent chromosomes, ensuring a balanced transfer of genetic information and preserving diversity within the solution space. Additionally, Gaussian mutation introduces symmetric perturbations, enabling further exploration of diverse regions in the solution space. These symmetric mechanisms are crucial for maintaining a balance between exploration and exploitation, thereby allowing the genetic algorithm to effectively search for the optimal solution. The flow of the GA-BP algorithm is shown in Figure 2.

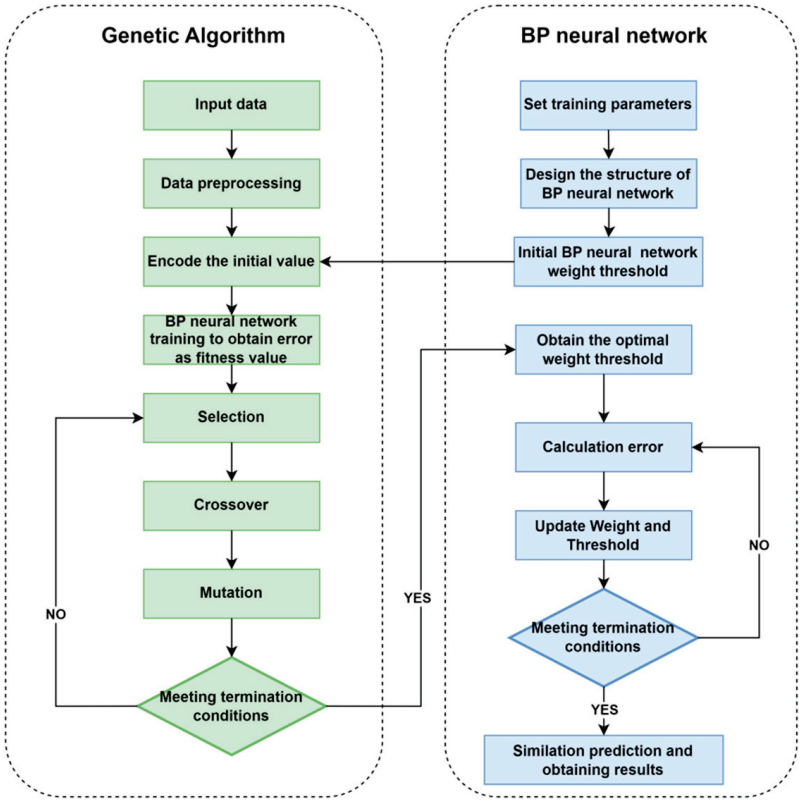


Figure 2. Flow chart of BP neural network optimized by GA.

2.3. Data Sources and Processing

2.3.1. Data Sources

The data used in this study were sourced from the Shanghai Data Exchange, where it was manually collected, organized, and quantified. Based on the data quality evaluation indicators outlined by the National Information Technology Standardization Technical Committee (GB/T 36344-2018 [42]) and the Data Value Assessment Guidelines (DB61/T 1809-2024 [43]), and further informed by an extensive review of relevant literature in the

data industry, the study focuses on data value evaluation, analyzing the key factors influencing data value. Nine primary indicators were identified: data quality, data accessibility, data coverage, data diversity, data adaptability, data volume, technical capability, market factors, and data source. From these, 21 secondary indicators were selected, including producer reputation, supplier qualifications, data quality certification, timeliness, update speed, spatial coverage, scenario coverage, multi-source heterogeneity, application scenario diversity, industry adaptability, data subject relevance, data scale, data increment, data dimensional richness, data update technical capability, functional complexity, pricing model, product type, self-production, open collection, and agreement acquisition. The dataset spans multiple sectors, including financial services, transportation, urban governance, healthcare, trade, industrial manufacturing, technological innovation, modern agriculture, and cultural tourism. To ensure data integrity and accuracy, we conducted data cleaning and deduplication on the original dataset. This process involved removing records with excessive missing values, incorrect formats, or duplicates, ultimately retaining 519 valid data entries. As a result, a data value assessment dataset was constructed, comprising 21 distinct data features, as outlined in Table 1. The sample data are shown in Table 2.

Table 1. Selection of indicators for the value evaluation model.

Target Layer	Number	Indicator Layer	Indicator Definition
Data quality	X1	Producer reputation	The reputation and credibility of the data provider in the marketplace, assessed based on historical transaction records, user feedback, and industry evaluation reports.
	X2	Supplier qualifications	The legal qualification of the data provider, evaluated through documentation such as business licenses and industry-specific certifications.
	X3	Data quality certification	Whether the dataset has been certified by authoritative institutions for accuracy, completeness, and reliability, based on official certification documents.
Data accessibility	X4	Timeliness	The temporal sensitivity and responsiveness of data updates, evaluated by the alignment between release timestamps and practical application demands.
	X5	Update speed	The actual frequency of data updates (e.g., daily, weekly, monthly), quantified based on specifications provided in the product documentation (e.g., updates per month).
Data coverage	X6	Spatial coverage	The spatial coverage of the dataset, measured by the number and density of geographic regions included (e.g., cities, provinces, countries).
	X7	Scenario coverage	The breadth of application domains supported by the dataset, determined by the number and diversity of use cases outlined in product documentation (e.g., urban development, transportation, tourism).
Data diversity	X8	Multi-source heterogeneity	Whether the dataset integrates heterogeneous sources (e.g., remote sensing, linguistic corpora, statistical data, IoT) and comprises diverse data structures (e.g., text, images, tabular data), based on source descriptions and underlying schema.
	X9	Application scenario diversity	The extent to which the dataset supports various business needs, evaluated by the number and type of application scenarios demonstrated in practical use cases (e.g., query-based, mapping, analytical tasks).

Table 1. *Cont.*

Target Layer	Number	Indicator Layer	Indicator Definition
Data adaptability	X10	Industry adaptability	Whether the dataset conforms to the standards and requirements of specific industries, assessed via keyword matching and domain categorization (e.g., cultural tourism, housing and urban development).
	X11	Data subject relevance	The thematic relevance of the dataset to specific domains (e.g., remote sensing, satellite, transportation, population), assessed through metadata and topic annotations.
Data volume	X12	Data scale	The total volume of the data resource, directly obtained from metadata fields (e.g., "storage size" = 2 GB).
	X13	Data increment	The incremental data volume per update, derived from metadata descriptors such as "incremental storage size" (e.g., 0.2 GB).
	X14	Data dimensional richness	The number of structured fields or metric dimensions, extracted from the underlying schema of the dataset.
Data volume	X15	Data update technical capability	The technical capability of the provider for data updating, evaluated by the adoption of advanced techniques such as automation or remote sensing interpretation.
	X16	Functional complexity	Functional diversity and operational complexity of the data product, determined by the presence of advanced features such as analytical tools, real-time querying, and data visualization.
Market factors	X17	Pricing model	The pricing model employed for data transactions (e.g., annual subscription, pay-per-use, one-time purchase), based on transaction policies specified by the platform.
	X18	Product type	The fundamental attributes or classification of the data product, reflecting its organizational format and intended use (e.g., static, dynamic, structured, or unstructured data).
Data source	X19	Self-production	Whether the data are independently collected or processed by the provider, determined by source attribution and qualification documentation.
	X20	Open collection	Whether the data are acquired via web scraping or open government sources, based on acquisition methods explicitly noted in the source description.
	X21	Agreement-based acquisition	Whether the data are lawfully acquired through contractual agreements with data sources, confirmed via indications of "agreement-based acquisition" or third-party partnerships.
-	-	Data value	The transaction price on data exchange platforms, equivalent to the listed price, is extracted and quantified from published pricing records.

Table 2. Partial sample data.

Number	Indicator Layer	Sample 1	Sample 2
X1	Producer reputation	79.6	82.4
X2	Supplier qualifications	0.79	0.5
X3	Data quality certification	8.6	8.7
X4	Timeliness	3.3	0
X5	Update speed	5	1
X6	Spatial coverage	5	5
X7	Scenario coverage	2	2
X8	Multi-source heterogeneity	0.85	0.72
X9	Application scenario diversity	7.8	5.7
X10	Industry adaptability	8.6	8.4
X11	Data subject relevance	0.87	0.8
X12	Data scale (GB)	4.9	800
X13	Data increment (GB)	0.02	0
X14	Data dimensional richness	122	500
X15	Data update technical capability	0.95	0
X16	Functional complexity	0.7	0.7
X17	Pricing model	4	10
X18	Product type	1	2
X19	Self-production	1	1
X20	Open collection	0	1
X21	Agreement-based acquisition	0	1
-	Data value	750,000	800,000

2.3.2. Correlation Analysis

Correlation analysis is a fundamental technique for feature selection, aiming to quantify the relationship between each feature and the target variable, thereby identifying features that significantly impact the target variable. In high-dimensional datasets some features may be redundant or noisy. Directly incorporating all features can increase computational complexity, exacerbate the curse of dimensionality, and negatively affect the model's stability and generalization capability. By performing correlation analysis we can effectively eliminate features that contribute little or have no relationship with the target variable, reduce data dimensionality, lower computational costs, and enhance model training efficiency.

In this study, we conducted a correlation analysis between each feature and the overall value to identify key factors influencing data value assessment. Feature selection methods can be categorized into three types: filter methods, wrapper methods, and embedded methods. Considering time efficiency and computational costs, we employed the filter method and used the Pearson correlation coefficient [44] for feature selection. The Pearson correlation coefficient is a widely used statistical metric for measuring the linear relationship between two continuous variables. It quantifies correlation by computing the ratio of the

covariance between two variables to the product of their respective standard deviations. The equation is as follows:

$$r = \frac{\sum (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum (X_i - \overline{X})^2 \sum (Y_i - \overline{Y})^2}} \tag{17}$$

where r is the Pearson correlation coefficient and the range of r is between $[-1, 1]$; X_i and Y_i represent the i-th data points of the variables X and Y, respectively; and $\overline{X}$ and $\overline{Y}$ are the means of variables X and Y, respectively.

2.4. Assessment of Model Performance

To evaluate the performance of the GA-BP and BP models, this study employs the mean absolute error (*MAE*), root mean square error (*RMSE*), and the coefficient of determination (R^2) to assess the prediction performance of each model. The MAE represents the average absolute error between the predicted values and the actual values, indicating the average level of model error. A smaller *MAE* indicates a smaller deviation between the predicted and actual values, with the unit being CNY (Chinese Yuan). The lower the *MAE*, the smaller the model's prediction bias regarding data value, reflecting the accuracy and reliability of the evaluation results. In data value assessment, a smaller *MAE* signifies that the model can more accurately predict the monetary value of data, thereby assisting decision-makers in better understanding and utilizing data resources. The root mean square error (*RMSE*) is the square root of the mean squared difference between predicted and actual values, with the unit being CNY (Chinese Yuan). A smaller *RMSE* indicates better overall model performance with more stable errors, thereby enhancing the reliability of data value assessment. This suggests that the model's predictions are generally closer to the actual values. The R^2 value reflects the degree of correlation between the predicted results and the actual outcomes. A value closer to 1 indicates better model fitting. The formulas for each of these metrics are provided in Equations (18)–(20) and are as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{18}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{19}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \overline{y}_i)^2} \tag{20}$$

where y_i denotes the actual value of the i-th sample; $\hat{y}_i$ represents the predicted value of the i-th sample; and $\overline{y}_i$ is the mean of all actual values.

3. Predictive Model Construction

3.1. Data Selection and Normalization

3.1.1. Feature Selection

As shown in Figure 3, by calculating the Pearson correlation coefficient between each feature and the overall value and performing significance testing using p-values, the following features were found to have significant correlation ($p < 0.05$): X2, X3, X4, X5, X8, X9, X10, X12, X13, X14, and X17. These 11 features with strong correlation were selected as the input parameters.

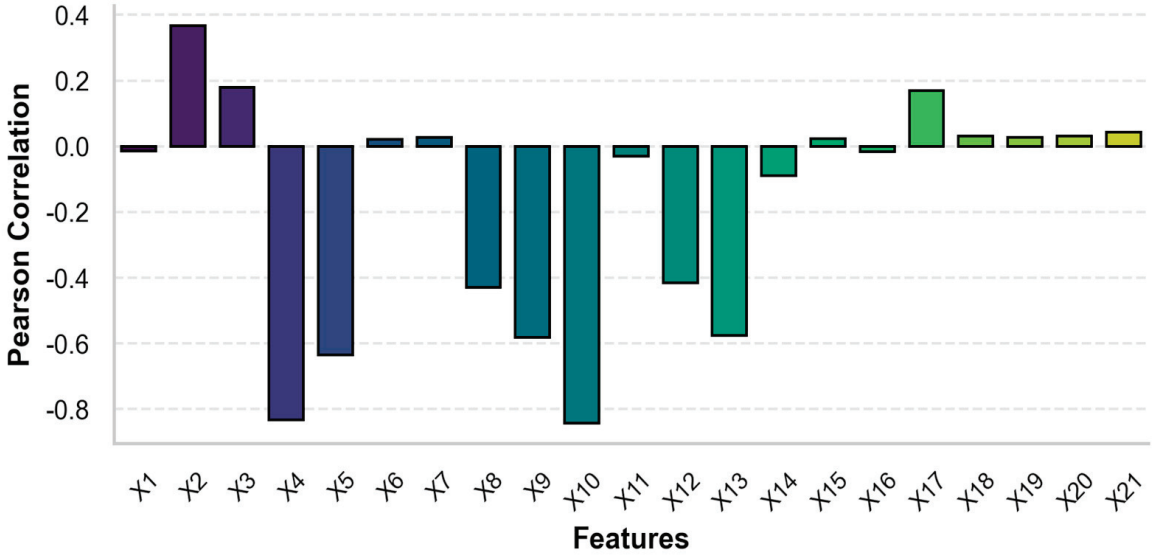


Figure 3. Pearson correlation coefficient.

3.1.2. Data Processing

(1) The primary objective of data normalization is to eliminate the influence of varying feature scales, ensuring that all data points are distributed within a common range. This process enhances model stability and accelerates convergence. In machine learning and neural network training, significant differences in feature value ranges can slow down optimization and negatively impact model performance. Therefore, normalization is a crucial step in data preprocessing. Common normalization techniques include min–max normalization and Z-score normalization [45], among others. Min–max normalization linearly transforms data into a specified range, improving the model's adaptability to different features. In this study, the mapminmax function in MATLAB R2022b is employed for data preprocessing and normalization, scaling the data within the range [0, 1]. The transformation equation is as follows:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{21}$$

where X_{norm} represents the standardized data, X denotes the original data, X_{min} is the minimum value of the original data, and X_{max} is the maximum value of the original data.

(2) The dataset consists of 519 samples, representing transaction records collected between 2022 and 2024. The data are stored in .xlsx format, with each record represented as structured textual data. Of these samples, 70% are randomly selected as the training set. The remaining 30% is evenly divided, with 15% assigned to the test set and 15% to the validation set. This partitioning strategy ensures that the model is trained on a sufficiently large dataset for effective learning while enabling an objective evaluation through independent test and validation sets.

3.2. Backpropagation Neural Networks

In this study, we assessed the performance of neural networks with varying numbers of hidden layers. The results revealed that the three-layer network achieved the highest performance, with an R^2 value of 0.90645. Increasing the number of layers resulted in diminished performance: the four-layer network had an R^2 of 0.87591 and the five-layer network performed at an R^2 of 0.8192. These findings suggest that adding more layers does not yield substantial improvements and may, in fact, reduce the model's effectiveness. Consequently, we selected the three-layer network as it offers the best balance between

model complexity and predictive accuracy. In light of these results, a three-layer BP neural network is constructed using the newff function, and the trainlm function is employed for training. The default Levenberg–Marquardt algorithm [46] is used for optimization as it is well-suited for medium-sized networks that require substantial memory but exhibit fast convergence. Several alternative activation functions were tested, and the combination of tansig and purelin delivered the best performance, achieving an R^2 of 0.90645. In comparison, the logsig and purelin combination yielded an R^2 of 0.87119 while the tansig–tansig combination resulted in an R^2 of 0.88207. The poslin (ReLU) and purelin combination achieved an R^2 of 0.90359. Therefore, the transfer function between the input layer and the hidden layer is the tansig S-function, whereas the transfer function between the hidden layer and the output layer is the purelin linear function.

This study initially considers a total of 21 indicators. After feature selection, 11 indicators with significant correlations are retained. Consequently, the number of nodes in the input layer is set to 11. To determine the optimal number of hidden-layer nodes, methods such as the node removal method and the expansion method are commonly used to balance network complexity and error minimization. In this experiment, the trial-and-error approach is adopted; starting with a small number of hidden-layer nodes, the number is gradually increased until the configuration yielding the lowest network error is identified. Additionally, the following empirical formula (Equation (22)) is used to provide an initial estimate for the number of hidden-layer nodes:

$$m = \sqrt{n + s} + \alpha \tag{22}$$

where n is the number of nodes in the input layer, s is the number of nodes in the output layer, m is the number of nodes in the hidden layer, and α is a constant chosen between [1, 10].

After calculation, the number of nodes in the hidden layer was chosen from the range [4, 13]. Through repeated experiments, as shown in Figure 4, the optimal number of hidden-layer nodes is 8.

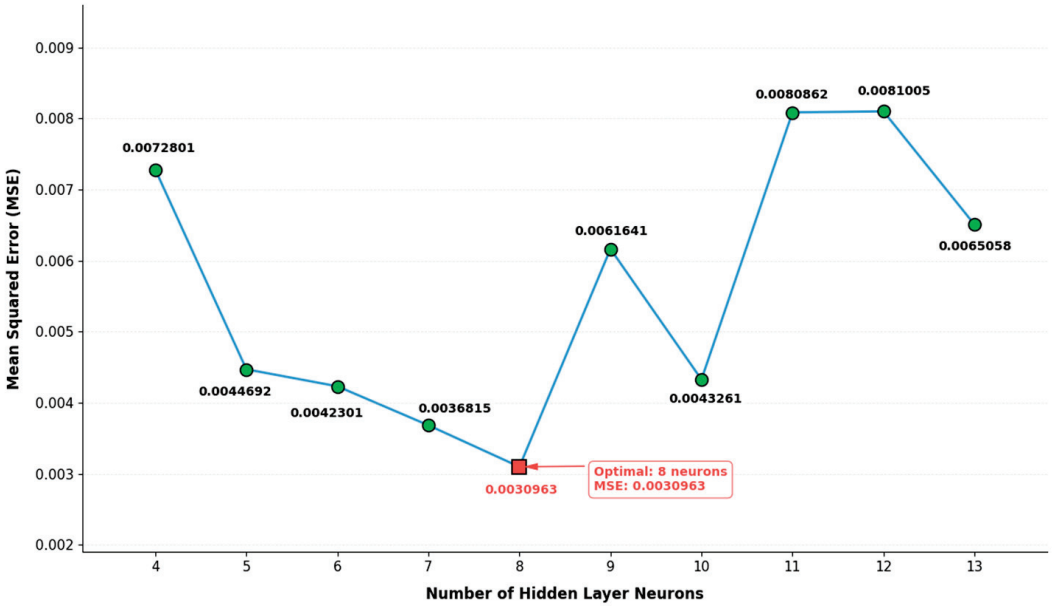


Figure 4. Impact of different hidden-layer node counts on error.

The output layer consists of a single node, representing the predicted data value. The neural network structure used in this study is illustrated in Figure 5. The final configuration

comprises an input layer with 11 nodes, a hidden layer with 8 nodes, and an output layer with 1 node. The specific network parameters are detailed in Table 3.

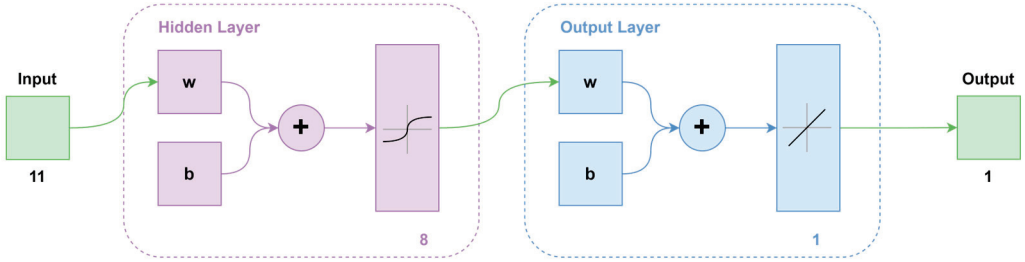


Figure 5. Flow chart of BP neural network.

Table 3. BPNN parameter settings.

Network Parameters	Value
Maximum number of iterations	1000
Learning rate	0.01
Convergence error	0.00001
Input layer threshold	11
Hidden layer threshold	8
Output layer threshold	1

3.3. Genetic Algorithm Implementation

In the genetic algorithm, a sensitivity analysis was performed to identify the optimal crossover rate for the genetic algorithm (GA), with a focus on the R^2 metric. Crossover rates ranging from 0.5 to 0.9 were evaluated. The results indicated that a crossover rate of 0.7 yielded the highest R^2 value of 0.97938, demonstrating the best model performance. In contrast, crossover rates of 0.6, 0.8, and 0.5 produced slightly lower R^2 values of 0.97935, 0.97895, and 0.97769, respectively. A crossover rate of 0.9 resulted in a more significant decline, with an R^2 value of 0.97395. Accordingly, a crossover rate of 0.7 was selected as the optimal configuration to achieve the highest prediction accuracy. In addition, a mutation rate of 0.2 was selected based on preliminary experiments. This value was found to maintain adequate genetic diversity while preserving convergence stability. Higher mutation rates introduced excessive randomness, which degraded model performance, whereas lower mutation rates led to premature convergence and suboptimal solutions. Therefore, a mutation rate of 0.2 was chosen to achieve an effective balance between exploration and exploitation during the evolutionary process.

Additionally, given the absence of a unified standard for population size, this study adopted a population size of 50 [37]. To validate this selection, population sizes of 5 [47], 20 [38], 30 [34], and 50 [37] were compared based on experimental results. Among these, a size of 50 yielded the highest R^2 value (0.97938), while sizes of 20, 30, and 5 produced lower R^2 values of 0.97155, 0.96892, and 0.95457, respectively. These results support the choice of a population size of 50 as optimal for achieving superior predictive performance. Therefore, the number of iterations was set to 50. As shown in Figure 6, after the 23rd generation, the individual fitness values no longer changed significantly, indicating that the genetic algorithm had reached a state of convergence.

Ultimately, the selection operator uses the roulette wheel selection method to select individuals with higher fitness. The crossover probability is set to 0.7, and the crossover operator uses two-point crossover. The mutation probability is set to 0.2, with the mutation operator employing Gaussian mutation. After multiple iterations of crossover and mutation the best-fitting individual is obtained and its data become the optimal initial weights and

thresholds for the BP neural network. These optimal weights and thresholds are then decoded, and the optimal connection weights between the input layer and the hidden layer, the hidden layer threshold, the connection weights between the hidden layer and the output layer, and the output layer threshold are assigned to the original BP neural network, resulting in the GA-BP neural network. The genetic algorithm is implemented using MATLAB's built-in ga toolbox, which provides efficient functions for optimization. The parameter settings for the genetic algorithm used in this study are shown in Table 4.

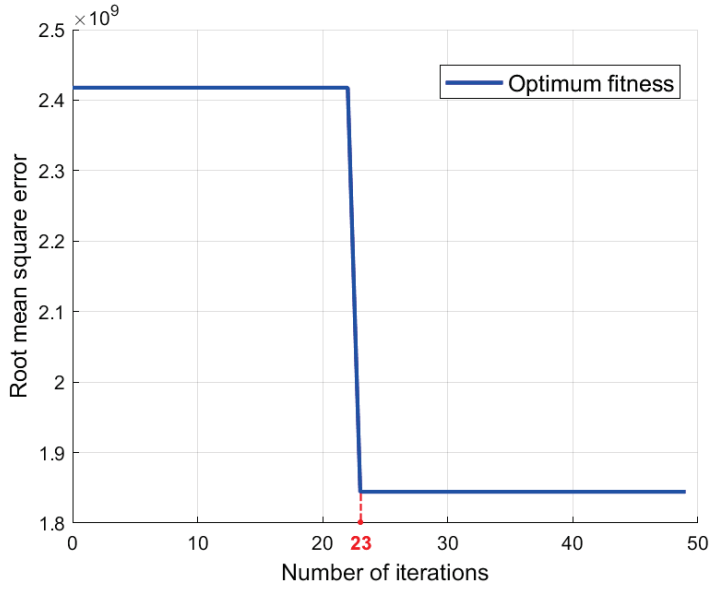


Figure 6. Individual fitness curve of genetic algorithm.

Table 4. Genetic algorithm parameters.

Population Size	Evolution Iteration	Crossover Factor	Mutation Factor
50	50	0.7	0.2

4. Analysis and Discussion

4.1. BP Neural Network Model Results

Without applying genetic algorithm optimization to initialize the weights and thresholds of the BP neural network, the network was trained directly using the 11 selected correlation features and the parameters from Table 3. The performance of the BP network was then evaluated, yielding the following results. Figure 7 illustrates the variation in the mean squared error (*MSE*) across training cycles. As training progresses, the *MSE* gradually decreases and reaches its optimal value at the 10th training cycle, indicating that the BP neural network performs optimally at this point. However, due to the gradient descent method used for weight updates, the BP neural network is prone to becoming trapped in local optima. Figure 8 presents the correlation coefficient (R) of the BP network across the training set, validation set, test set, and overall dataset. As shown in the figure, the correlation coefficient (R) for the test set is 0.96314, while the overall correlation coefficient is 0.97704. These results indicate that most of the predicted values closely align with the training set, though some deviations persist in certain data points. Further analysis reveals that the BP neural network achieves a mean absolute error (*MAE*) of 63,420.8503 and a coefficient of determination (R^2) of 0.90645. While the model can capture the overall data trend its generalization ability remains limited, leaving room for further improvement.

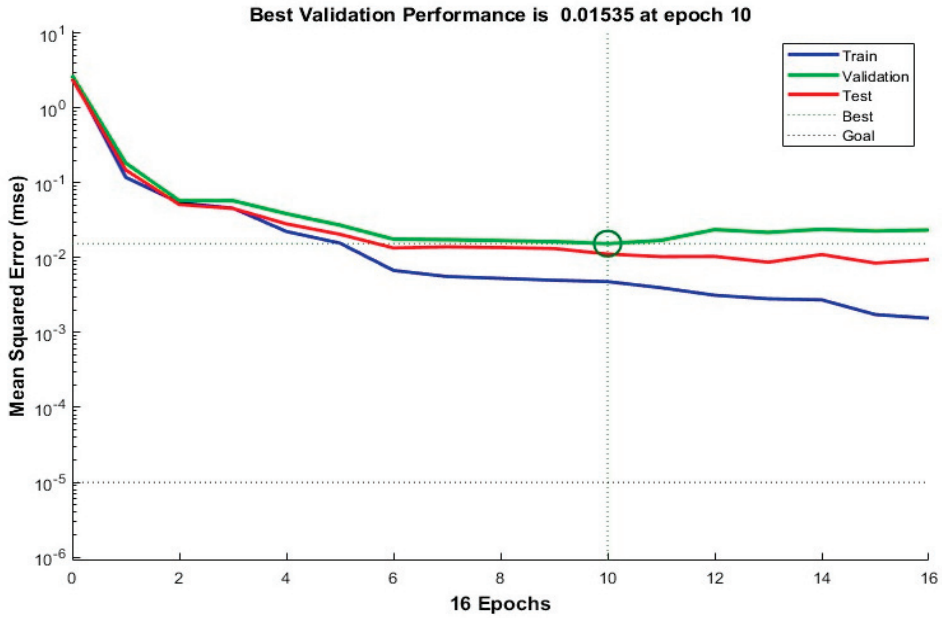


Figure 7. Network performance of BP model.

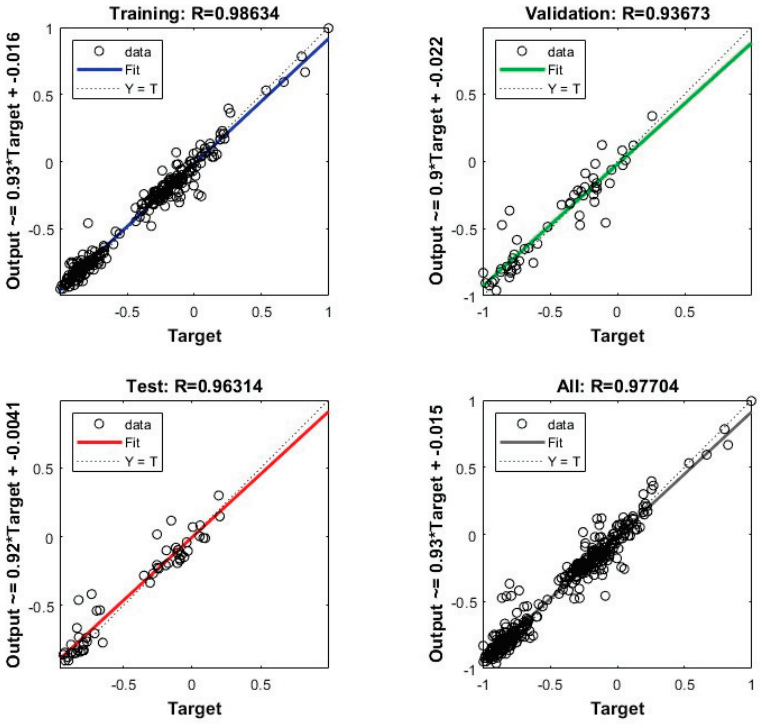


Figure 8. BP linear regressions of training sets, validation sets, test sets, and the overall data.

4.2. GA-BP Model Results

By applying the genetic algorithm to optimize the BP neural network, the GA-BP model was trained using the 11 selected correlation features and the parameters from Table 4. The results are as follows. Figure 9 illustrates the trend of mean squared error (*MSE*) during training. As training progresses, the *MSE* gradually decreases and converges, reaching its optimal value at the 34th training cycle. This indicates that the GA-BP network achieves its best performance at this point. Figure 10 presents the correlation coefficient (*R*) of the GA-BP model across the training set, validation set, test set, and overall dataset. As shown in the figure, the correlation coefficient (*R*) for the test set is 0.9915, while the overall correlation coefficient is 0.99288. These values suggest that the GA-BP model maintains

consistent performance across different datasets and exhibits strong generalization ability. Furthermore, the GA-BP model achieves a mean absolute error (*MAE*) of 33,221.7576 and a coefficient of determination (R^2) of 0.97938, demonstrating high precision and reliability in data value prediction.

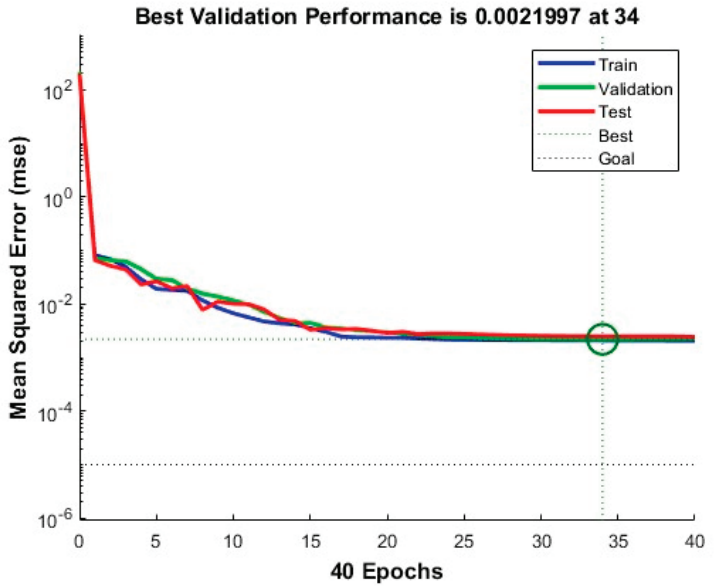


Figure 9. Network performance of GABP model.

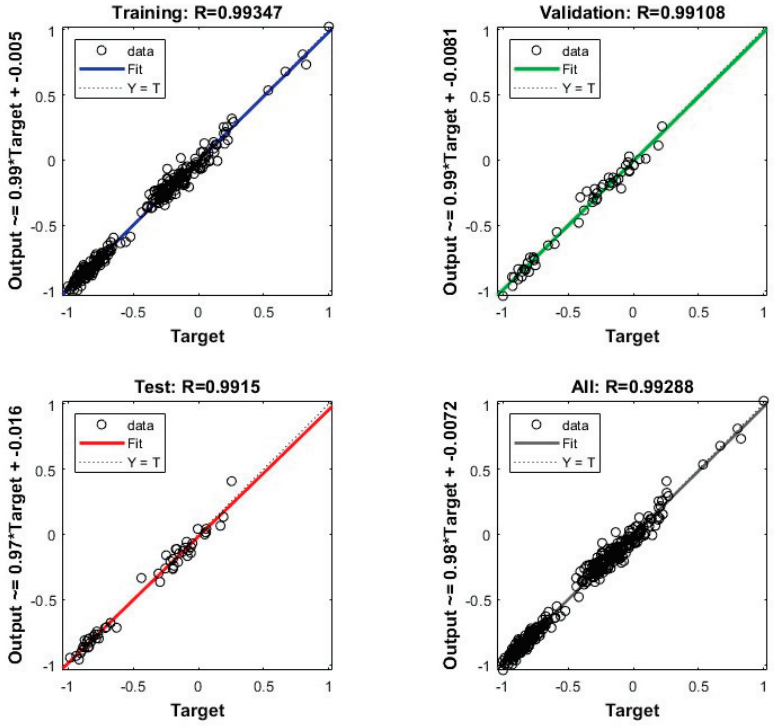


Figure 10. GABP linear regressions of training sets, validation sets, test sets, and the overall data.

4.3. Model Comparison

4.3.1. Comparison with Other Baseline Models

To assess the effectiveness and superiority of the GA-BP algorithm in data value evaluation, we compared it with LSTM, CNN, RF, and SVM models. The dataset was processed in the same way as for the GA-BP model. The parameter settings for each model are shown in Table 5.

Table 5. Model Parameter Settings for LSTM, SVM, RF, and CNN.

Model	Parameter	Setting
LSTM	Maximum Epochs	1000
	Learning Rate	0.01
	Activation Function	ReLU
	Optimizer	Adam
	Learning Rate Decay Factor	0.1
	Decay Period	1000
SVM	Penalty Factor (C)	4
	Gamma (Radial Basis Function)	0.8
	Kernel Type	RBF
	Epsilon (Slack Variable Tolerance)	3
RF	Number of Decision Trees	100
	Minimum Leaf Size	5
	Minimum Samples per Leaf	5
CNN	Mini-Batch Size	32
	Maximum Epochs	1000
	Learning Rate	0.01
	Learning Rate Decay Factor	0.1
	Decay Period	1000

For each model, a comprehensive hyperparameter tuning process was conducted to optimize performance. To maintain consistency with the GA-BP model, the maximum number of iterations (epochs) and the learning rate were standardized across all models. Specifically, the maximum number of iterations was set to 1000 and the learning rate was set to 0.01 for the LSTM and CNN models.

The remaining hyperparameters for each model were tuned to achieve optimal performance. For the LSTM model, hyperparameters such as the activation function, optimizer, and learning rate decay factor were optimized using grid search. Learning rate decay factors were tested within the range of 0.01 to 0.5, and model performance was evaluated based on validation loss. Additionally, the ReLU activation function and Adam optimizer were selected for their proven effectiveness in training deep learning models. The best combination of these hyperparameters was determined through cross-validation. For the SVM model, a grid search was performed to fine-tune the penalty factor (C) and gamma parameter. The penalty factor was tested between 1.0 and 10.0, while gamma values ranged from 0.1 to 1.0. The optimal parameter combination, which yielded the best classification performance as determined by cross-validation, was selected. This process enhanced the decision boundary and improved the SVM's ability to generalize to unseen data. For the RF model, the number of decision trees and the minimum number of leaf samples were tuned using random search. The number of trees was tested between 50 and 200, while the minimum number of leaf samples was adjusted between 1 and 10. The goal of the random search was to balance model complexity and accuracy, ensuring the model would not overfit while providing robust predictions on the test data. For the CNN model, we focused on tuning the batch size, learning rate decay factor, and decay period. Using grid search, batch sizes between 16 and 64 and learning rate decay factors between 0.1 and 0.5 were tested. The objective was to minimize training loss and enhance the model's generalization ability, thereby preventing overfitting and improving predictive performance.

Figures 11–16 provide a visual comparison of the predicted and actual values across different models. The results demonstrate that the GA-BP model achieves the highest fitting accuracy, with its prediction curve closely following the 45-degree line indicating strong performance in capturing underlying data patterns. In contrast, the traditional BP

model exhibits significant fluctuations, with its curve deviating notably from the 45-degree line, highlighting inconsistent accuracy. The LSTM model shows divergence in some areas, reflecting difficulties in handling the dataset's complexities and its limited ability to generalize, particularly due to its challenges in modeling non-sequential relationships. The SVM model experiences the largest divergence, with substantial fluctuations from the 45-degree line a result of its sensitivity to data noise and the regularization effect, both of which compromise stability and generalization. The RF model initially fits well but begins to diverge in later stages as it fails to capture long-term dependencies and complex patterns, leading to performance degradation. The CNN model performs reasonably well at first but shows increasing divergence as the predictions progress as its focus on local patterns, typically suited for image data, does not align with the structure of the current dataset, causing accuracy to decline over time.

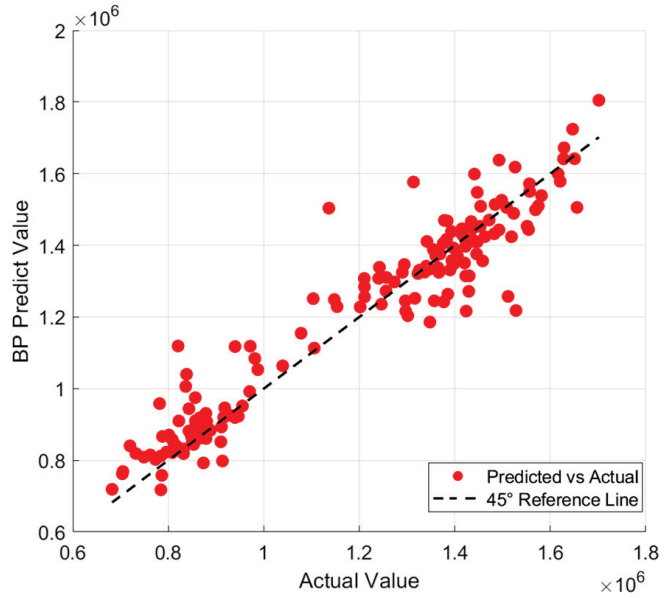


Figure 11. Comparison of predicted values and true values of BP model.

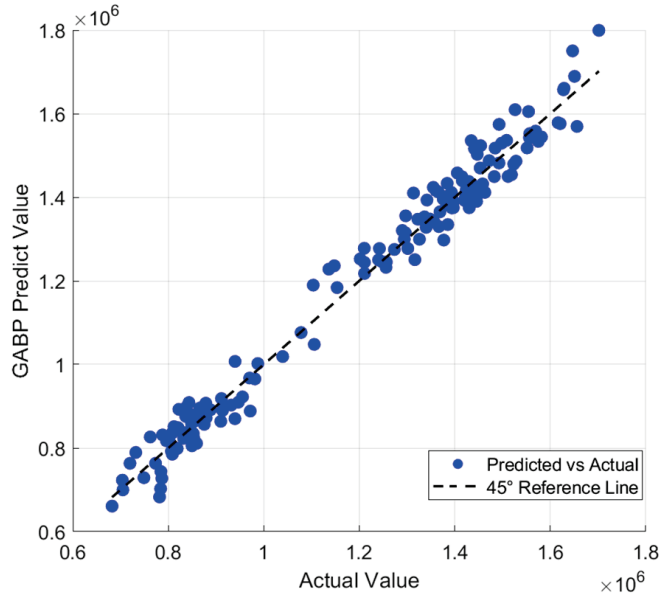


Figure 12. Comparison of predicted values and true values of GA-BP model.

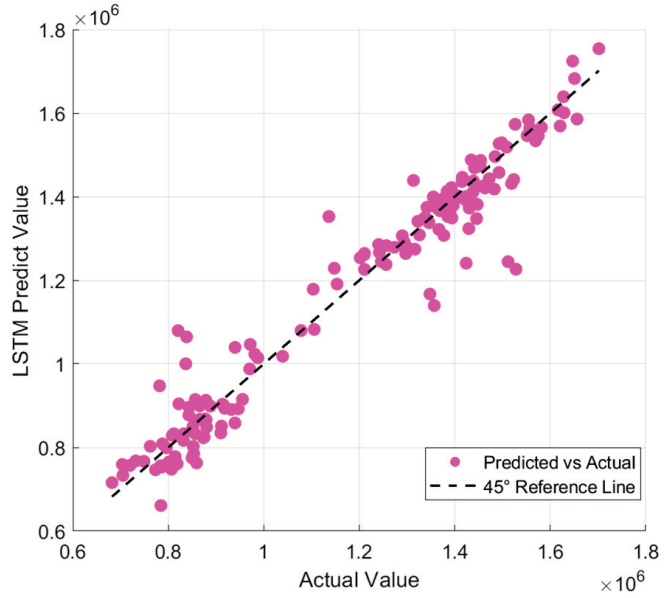


Figure 13. Comparison of predicted values and true values of LSTM.

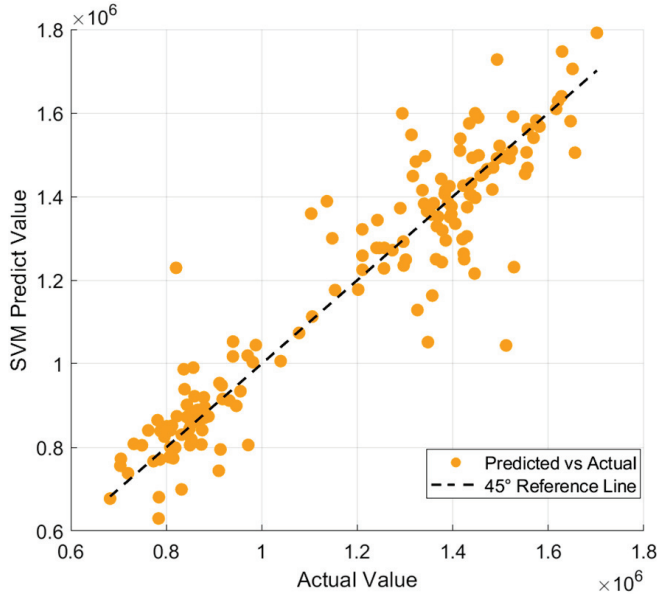


Figure 14. Comparison of predicted values and true values of SVM.

Figure 17 provides additional validation of the GA-BP model's superiority. In terms of error distribution, the GA-BP model exhibits significantly lower fluctuation levels than other models, indicating greater stability. The traditional BP model shows the widest error distribution, suggesting weaker generalization capability. The SVM and CNN models display highly dispersed errors, implying greater volatility in prediction accuracy. While the LSTM and RF models maintain relatively stable error distributions, they do not reach the performance level of the GA-BP model. These results highlight the GA-BP model's distinct advantage in error control, further reinforcing its reliability and precision in data value assessment.

As shown in Table 6, the GA-BP model demonstrates significant advantages in prediction error reduction and fitting capability. Compared to other models, GA-BP achieves the lowest *MAE* and *RMSE*, reducing these metrics by 23.7% and 28.8%, respectively, relative to the second-best RF model. Additionally, its R^2 is the highest, showing a 2.1% improvement over RF, which strongly indicates that GA-BP offers superior prediction accuracy and

fitting performance. Among all the compared models the traditional BP neural network performs relatively poorly, with its *MAE* and *RMSE* values only marginally better than those of the worst-performing SVM model. After optimization with the genetic algorithm, the GA-BP model improves significantly, reducing the *MAE* by 47.6%, the *RMSE* by 53.0%, and increasing the R^2 by 8.0%. When compared to deep learning models, GA-BP achieves a 35.7% reduction in *MAE* compared to LSTM and a 48.1% reduction compared to CNN. Its *RMSE* is also 49.5% lower than that of LSTM and 55.0% lower than that of CNN. In comparison to other machine learning models, GA-BP reduces *RMSE* by 60.5% relative to SVM and by 28.8% relative to RF. These results demonstrate that GA-BP effectively mitigates the BP network's tendency to fall into local optima by leveraging the global search capabilities of the genetic algorithm. Among all the models evaluated in this study, GA-BP exhibits the strongest optimization effect and generalization performance.

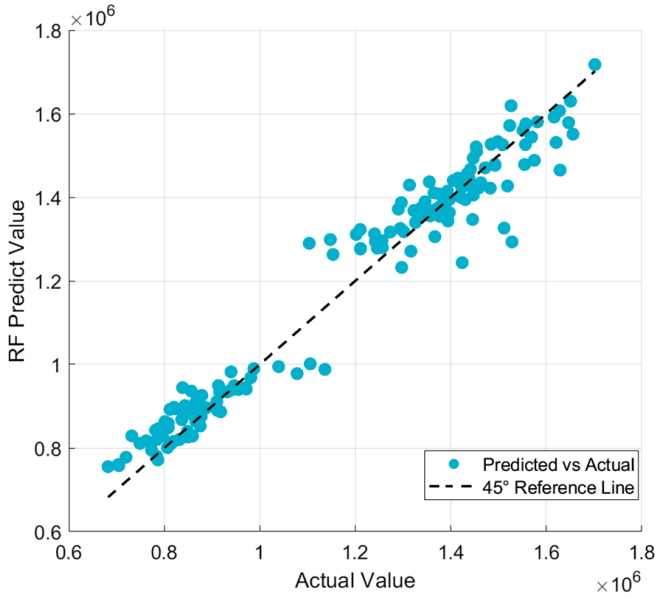


Figure 15. Comparison of predicted values and true values of RF.

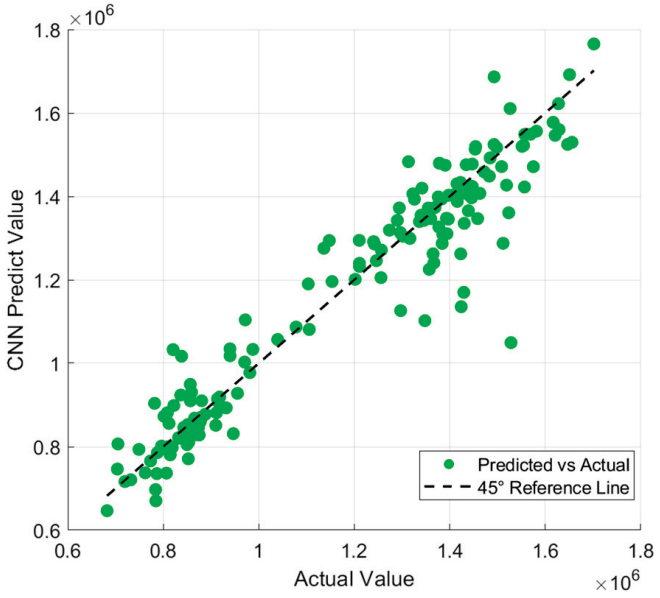


Figure 16. Comparison of predicted values and true values of CNN.

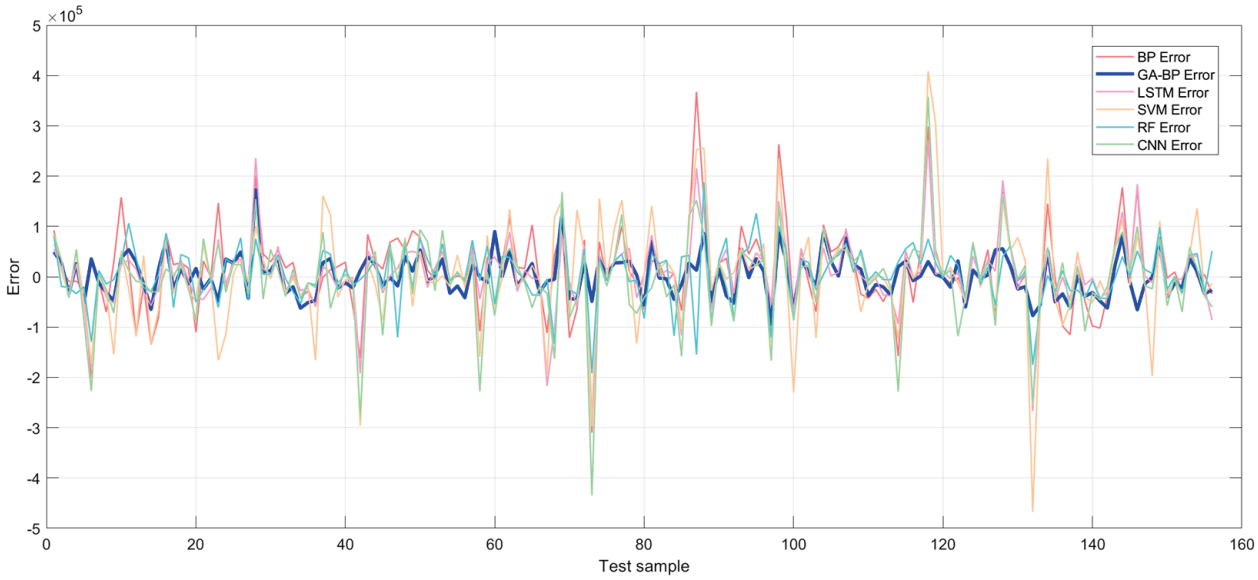


Figure 17. Error comparison with other foundation models.

Table 6. Comparison of evaluation metrics for different baseline models.

Model	*MAE*	*RMSE*	R^2
BP	63,420.8503	89,562.8324	0.90645
LSTM	51,652.8086	83,295.2656	0.91909
CNN	64,022.9688	93,487.3047	0.89807
RF	43,509.3184	59,049.8405	0.95934
SVM	71,727.9614	106,348.2779	0.8681
GA-BP	**33,221.7576**	**42,050.7613**	**0.97938**

4.3.2. Comparison with Other Optimization Models

To further evaluate the effectiveness of the genetic algorithm (GA) in optimizing the backpropagation (BP) neural network, this section compares the GA-BP model with other BP-based optimization algorithms, including particle swarm optimization BP (PSO-BP), sparrow search algorithm BP (SSA-BP), and whale optimization algorithm BP (WOA-BP). The dataset used for all models is preprocessed through Pearson correlation analysis to select 11 relevant features. To ensure a fair comparison, the BP network parameters for each optimization algorithm are kept consistent with those listed in Table 3. All comparative experiments employed the same dataset partitioning strategy as the GA-BP model. The population size and maximum number of iterations for SSA-BP, PSO-BP, and WOA-BP were identical to those in GA-BP, both set to 50.

For SSA-BP, three key parameters—population distribution (PD), search space variation (SD), and selection probability (ST)—were optimized. A grid search was employed to assess the impact of varying values for each parameter on convergence speed and accuracy. The PD parameter, affecting population distribution, was tested between 0.3 and 0.5; SD, controlling search space variation, between 0.2 and 0.4; and ST, which influences selection probability, between 0.6 and 0.8. The optimal values were PD = 0.4, SD = 0.3, and ST = 0.7, balancing exploration and exploitation for improved convergence and accuracy while avoiding premature convergence. Similarly, for PSO-BP, the two critical parameters—cognitive learning factor (C1) and social learning factor (C2)—were optimized. These parameters control how particles adjust their positions based on individual experience (C1) and the experience of neighboring particles (C2). A grid search was conducted to evaluate various combinations of C1 and C2 values, both tested in the range of 1.5 to 3.0. After testing several combinations, the optimal values were found to be C1 = 2.0 and

C2 = 2.0. These settings facilitated a balanced exploration–exploitation trade-off, leading to faster convergence without compromising the quality of the solution. In contrast, WOA-BP did not require parameter tuning as the whale optimization algorithm (WOA) inherently manages exploration and exploitation through its own mechanism, without the need for user-defined parameters.

Figures 18–20 present a visual comparison of the predicted and actual values across different models. The results demonstrate that the GA-BP model achieves the highest fitting accuracy, with its prediction curve aligning more closely with the 45-degree line, reflecting superior performance in capturing underlying data patterns. In contrast, the SSA-BP model shows significant deviations, with some data points moving far from the 45-degree line. This is due to the model's tendency to become trapped in local optima, impairing its capacity to accurately fit key transitions in the data. The PSO-BP model, although moderately aligned with the 45-degree line, exhibits a moderate level of divergence, reflecting the limited convergence capabilities of particle swarm optimization which results in less refined predictions. The WOA-BP model also displays noticeable deviations, especially at extreme values, as it struggles to capture the full range of data variations, leading to a poor fit at these points. Overall, the GA-BP model consistently demonstrates better alignment with the 45-degree line, ensuring more stable and accurate predictions and effectively addressing the challenges faced by the other models.

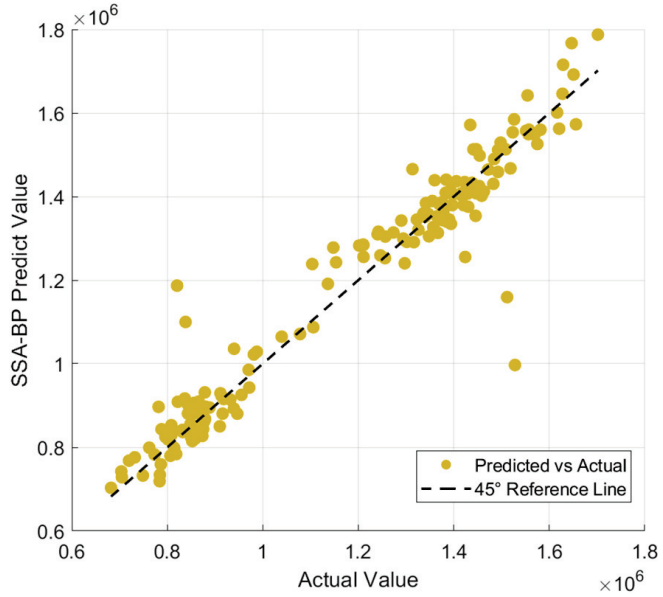


Figure 18. Comparison of predicted values and true values of SSA-BP model.

Additionally, Figure 21 illustrates that the error fluctuation ranges of the PSO-BP, SSA-BP, and WOA-BP models are significantly larger than that of GA-BP. The prediction errors of PSO-BP exhibit periodic oscillations across samples, while SSA-BP, despite having a lower overall error magnitude, experiences sudden error spikes in highly nonlinear segments, indicating a lack of robustness. WOA-BP, on the other hand, displays the largest error amplitude, with persistent high deviations near extreme points suggesting premature convergence in its parameter optimization process. By contrast, the GA-BP model maintains a consistently low error range, demonstrating significantly improved uniformity and stability in error distribution. These findings quantitatively validate the genetic algorithm's superiority in neural network optimization.

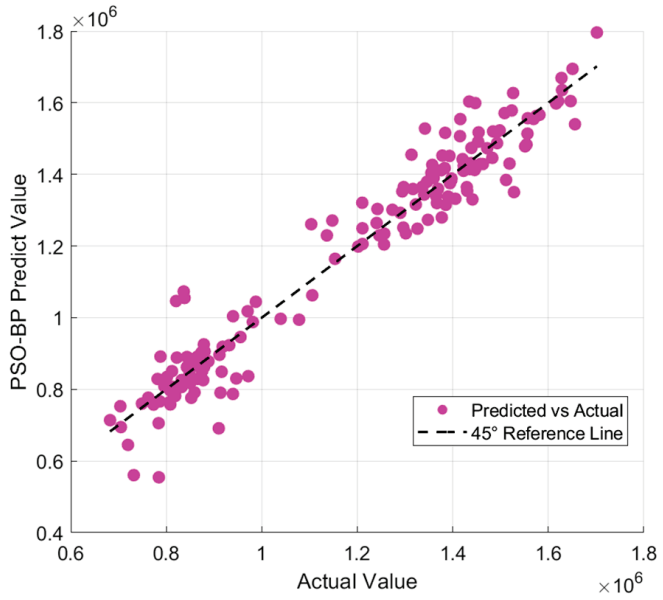


Figure 19. Comparison of predicted values and true values of PSO-BP model.

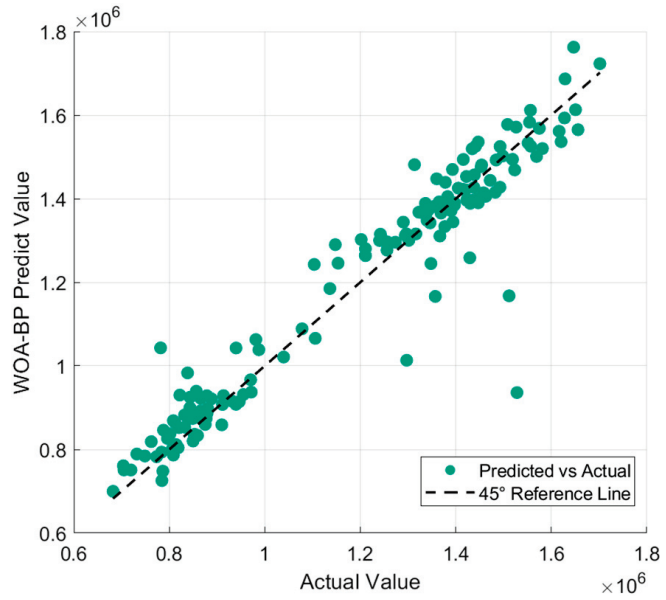


Figure 20. Comparison of predicted values and true values of WOA-BP model.

Table 7 compares the performance of different optimization models in terms of prediction errors and goodness-of-fit. The SSA-BP model showed improvements across all three metrics—*MAE*, *RMSE*, and R^2—with *MAE* decreasing by 17.0%, *RMSE* by 13.7%, and R^2 increasing to 0.9304. In comparison, the PSO-BP model showed limited optimization effects, with *MAE* reduced by only 6.3%, *RMSE* improved by 1.3%, and a slight increase in R^2 to 0.9089. Although the WOA-BP model reduced *MAE* by 18.9%, its *RMSE* increased by 1.4%, and R^2 slightly decreased to 0.90385, revealing the instability of its optimization performance. In contrast, the GA-BP model demonstrated notable improvements across all evaluation metrics. *MAE* was reduced by 47.6%, *RMSE* improved by 53.0%, and R^2 increased to 0.97938, significantly outperforming the other models and highlighting the advantages of the genetic algorithm in neural network optimization.

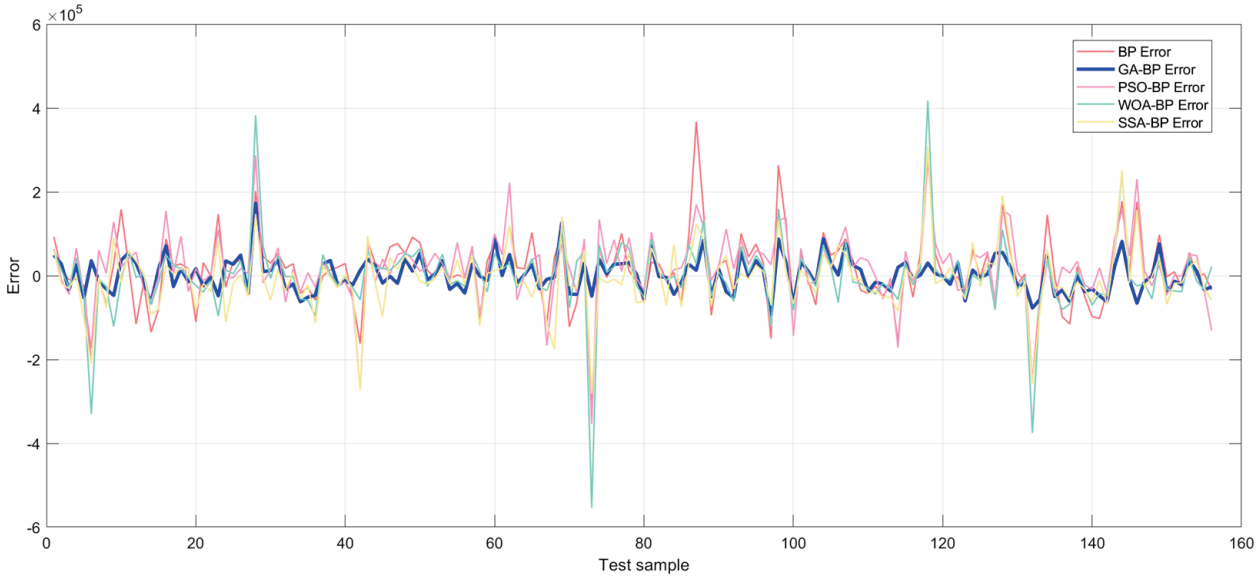


Figure 21. Error comparison with other optimization algorithms.

Table 7. Comparison of evaluation metrics for different optimization models.

Model	*MAE*	*RMSE*	R^2
BP	63,420.8503	89,562.8324	0.90645
PSO-BP	55,265.0438	75,869.5022	0.93287
SSA-BP	52,651.3975	77,253.3347	0.9304
WOA-BP	51,732.5201	83,761.3627	0.91818
GA-BP	**33,221.7576**	**42,050.7613**	**0.97938**

The experimental results indicate that the genetic algorithm, through its global search capability, effectively reduces the BP neural network's tendency to fall into local optima. Among all the models evaluated, GA-BP showed the best prediction accuracy and model fitting, providing strong support for its application in complex prediction tasks.

4.4. Statistical Analysis of Model Performance

To determine whether the differences in prediction performance between the models are statistically significant an analysis of variance (ANOVA) was conducted. We compared the performance of the following models: the standard BP neural network (BP), the GA-optimized BP neural network (GABP), the SSA-optimized BP (SSA-BP), the PSO-optimized BP (PSO-BP), the WOA-optimized BP (WOA-BP), the long short-term memory (LSTM) network, the support vector machine (SVM), the random forest (RF), and the convolutional neural network (CNN).

The ANOVA results produced an F-statistic of 22.14 and a p-value of 1.77×10^{-17}. Since the p-value is well below the commonly accepted significance threshold of 0.05 we reject the null hypothesis, indicating that significant differences exist between the models. This suggests that the GA-optimized BP neural network (GABP) outperforms the other models. To further validate these findings, we conducted a Tukey honest significant difference (HSD) test, which confirmed that GABP shows significant performance differences compared to all other models, with adjusted p-values consistently below 0.05. These results support the conclusion that GA optimization substantially improves the predictive performance of the BP neural network, making GABP the superior model in this comparison.

4.5. Limitations and Future Work

While this study offers valuable insights into data value assessment, several limitations must be acknowledged. First, the relatively small sample size (n = 519) may limit the generalizability of the findings. Second, potential source bias could arise from the exclusive use of data collected from the Shanghai Data Exchange, which may constrain the representativeness of the dataset. Third, the study was conducted within a single country, potentially limiting the applicability of the results across different regulatory, economic, and cultural environments.

Future research should seek to overcome these limitations. Increasing the sample size and incorporating data from multiple exchanges and platforms would enhance the robustness and generalizability of the results. Moreover, cross-national comparative studies are encouraged to validate the model's applicability in diverse market contexts. Further research could also explore the temporal dynamics of data value evolution and investigate the integration of emerging technologies, such as federated learning and privacy-preserving computation, to further advance the field. Additionally, exploring the use of synthetic data to simulate broader market conditions and test the model's performance in scenarios where real-world data are limited could provide valuable insights and help in validating the model across diverse, hypothetical market environments.

5. Conclusions

This study introduces a BP neural network model optimized by a genetic algorithm (GA). Through comparative experimental analysis, the results demonstrate that the GA-BP model significantly outperforms traditional BP neural networks in both prediction accuracy and stability. Furthermore, the GA-BP model exhibits clear advantages over popular machine learning models, including long short-term memory (LSTM), convolutional neural networks (CNNs), random forest (RF), and support vector machines (SVMs), as well as optimization algorithms such as particle swarm optimization (PSO), sparrow search algorithm (SSA), and whale optimization algorithm (WOA). These advantages are especially evident when addressing highly nonlinear and complex tasks, such as data value assessment. This is primarily attributed to the effective integration of the genetic algorithm's global search capability with the BP neural network's ability to model nonlinear relationships, which collectively overcome the issue of local optima that often hampers traditional BP networks.

In terms of application value, this study contributes significantly beyond simply applying the GA-BP neural network model to the novel field of data value assessment. Specifically, it introduces a new approach by integrating data indicators from various industries to construct a more objective and comprehensive data value assessment model. This approach addresses the subjectivity often found in traditional evaluation methods. The model has potential applications in pricing platforms across sectors such as finance, healthcare, and manufacturing. Furthermore, the study proposes a market-driven valuation method that leverages historical transaction data from multiple sectors, enabling the model to provide intelligent, cross-industry data value assessments. This methodology not only enhances the understanding of data value evaluation but also lays the foundation for a standardized framework that can be applied across different industries. Whilst the study provides valuable insights, limitations such as the relatively small sample size and the data being sourced from a single platform are acknowledged, which may impact the generalizability of the findings.

Future research could expand in several directions. First, in terms of algorithm optimization, further exploration of integrating genetic algorithms with other optimization techniques or incorporating more advanced algorithms could further improve prediction accuracy and generalization ability. Second, in terms of model architecture, the introduc-

tion of deep learning techniques, such as attention mechanisms, could enhance feature extraction capabilities. Third, in terms of application expansion, incorporating external factors such as market dynamics and policy environments into the evaluation framework could result in a more comprehensive data value assessment model. Fourth, addressing the current limitations by expanding the sample size, diversifying data sources, synthetic data, and validating the model across multiple countries would enhance its robustness and improve its applicability in international contexts. The GA-BP model proposed in this study offers an effective technical approach for data value assessment. As the digital economy continues to develop, this method is poised to play a significant role in a broader range of data-driven scenarios, supporting the market-based allocation of data resources.

Author Contributions: Conceptualization, X.Q. and X.Y.; methodology, Q.H.; software, Q.H.; validation, X.Q., X.Y. and Q.H.; formal analysis, Q.H.; investigation, Q.H.; resources, X.Q. and X.Z.; data curation, Q.H.; writing—original draft preparation, Q.H.; writing—review and editing, Q.H. and X.Y.; visualization, Q.H.; supervision, X.Q., X.Z. and X.Y.; project administration, X.Q., X.Z. and X.Y.; funding acquisition, X.Q. and X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the project of the GuangXi Information Center, grant number XZZB202410055F.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to the project not being completed yet.

Conflicts of Interest: Authors Xujiang Qin and Xin Zhang were employed by the Guangxi Information Center. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Williamson, O.E. The economics of organization: The transaction cost approach. *Am. J. Sociol.* **1981**, *87*, 548–577. [CrossRef]
2. Han, X.; Wang, S. Attributes, identification and valuation methods of data assets. *Stat. Inf. Forum* **2023**, *38*, 3–13.
3. Baum, A.; Baum, C.M.; Nunnington, N.; Mackmin, D. *The Income Approach to Property Valuation*; Estates Gazette: London, UK, 2013.
4. Wang, Y.; Zhao, H. Data asset value assessment literature review and prospect. *J. Phys. Conf. Ser.* **2020**, *1550*, 032133.
5. Jonathan, D. The Data Asset: Databases, Business Intelligence, and Competitive Advantage. *Comput. Oper. Res.* **1999**, *3*, 16–24.
6. Xu, X.; Zhang, Z.; Hu, Y. Research on data asset statistics and accounting issues. *Manag. World* **2022**, *38*, 16–30+2.
7. Lei, X.; Zhang, F. Research on data asset valuation based on the income present value method. *Stat. Inf. Forum* **2023**, *38*, 3–13.
8. Yang, K. Research on enterprise data asset value assessment method. *China Manag. Informationization* **2022**, *25*, 88–91.
9. Zhang, Z.; Tan, C. Research on the value composition and valuation limitations of data assets. *Commer. Econ.* **2021**, *6*, 81–82.
10. Lu, M.; Ouyang, W. Research on the institutional mechanism of data factor marketization and data asset valuation and pricing. *Xinjiang Soc. Sci.* **2021**, *1*, 43–53.
11. Lin, F. Big data assets and their value assessment methods: Literature review and prospects. *Financ. Manag. Res.* **2020**, *6*, 1–5.
12. Longstaff, F.A.; Schwartz, E.S. Valuing American options by simulation: A simple least-squares approach. *Rev. Financ. Stud.* **2001**, *14*, 113–147. [CrossRef]
13. Zuo, W.; Liu, L. Research on big data asset valuation method based on user perceived value. *Intell. Theory Pract.* **2021**, *44*, 71–77.
14. Hu, C.; Li, Y.; Zheng, X. Data assets, information uses, and operational efficiency. *Appl. Econ.* **2022**, *54*, 6887–6900. [CrossRef]
15. Niyato, D.; Alsheikh, M.A.; Wang, P.; Kim, D.I.; Han, Z. Market model and optimal pricing scheme of big data and Internet of Things (IoT). In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–6.
16. Zhang, C. *Research on Data Asset Value Analysis Model and Transaction System*; Beijing Jiaotong University: Beijing, China, 2018.
17. Wang, X.; Hao, H.; Zhang, S.; Wang, J. Research on big data value assessment based on fuzzy neural network. *Sci. Technol. Manag.* **2019**, *21*, 1–9.
18. Ren, J. Data asset pricing model and system design based on integrated machine learning. *China Manag. Informationization* **2022**, *25*, 80–82.
19. Wang, J. Research on data asset value evaluation method. *Times Financ.* **2016**, *12*, 292–293.

20. Yan, P. *Research on Data Asset Value Assessment Based on Machine Learning*; Yunnan University: Kunming, China, 2022.
21. Li, J.; Cheng, J.H.; Shi, J.Y.; Huang, F. Brief introduction of back propagation (BP) neural network algorithm and its improvement. In *Advances in Computer Science and Information Engineering: Volume 2*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 553–558.
22. Huo, L.; Jiang, B.; Ning, T.; Yin, B. A BP neural network predictor model for stock price. In Proceedings of the Intelligent Computing Methodologies: 10th International Conference, ICIC 2014, Taiyuan, China, 3–6 August 2014; Proceedings 10. Springer International Publishing: Cham, Switzerland, 2014; pp. 362–368.
23. Chen, T.; Peng, L.; Yin, X.; Rong, J.; Yang, J.; Cong, G. Analysis of user satisfaction with online education platforms in China during the COVID-19 pandemic. *Healthcare* **2020**, *8*, 200. [CrossRef]
24. Kalinić, Z.; Marinković, V.; Kalinić, L.; Liébana-Cabanillas, F. Neural network modeling of consumer satisfaction in mobile commerce: An empirical analysis. *Expert Syst. Appl.* **2021**, *175*, 114803. [CrossRef]
25. Feng, W.; Feng, F. Research on the multimodal digital teaching quality data evaluation model based on fuzzy BP neural network. *Comput. Intell. Neurosci.* **2022**, *2022*, 7893792. [CrossRef]
26. Deng, Y.; Zhou, X.; Shen, J.; Xiao, G.; Hong, H.; Lin, H.; Wu, F.; Liao, B.-Q. New methods based on back propagation (BP) and radial basis function (RBF) artificial neural networks (ANNs) for predicting the occurrence of haloketones in tap water. *Sci. Total. Environ.* **2021**, *772*, 145534. [CrossRef]
27. Wang, Y.L.; Liu, Y.; Che, S.B. Research on pattern recognition based on BP neural network. *Adv. Mater. Res.* **2011**, *282*, 161–164. [CrossRef]
28. Ye, Z.; Kim, M.K. Predicting electricity consumption in a building using an optimized back-propagation and Levenberg–Marquardt back-propagation neural network: Case study of a shopping mall in China. *Sustain. Cities Soc.* **2018**, *42*, 176–183. [CrossRef]
29. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control. Eng.* **2020**, *8*, 22–34. [CrossRef]
30. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
31. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408. [CrossRef]
32. Mirjalili, S. Evolutionary algorithms and neural networks. *Stud. Comput. Intell.* **2019**, *780*, 43–55.
33. Ding, S.; Su, C.; Yu, J. An optimizing BP neural network algorithm based on genetic algorithm. *Artif. Intell. Rev.* **2011**, *36*, 153–162. [CrossRef]
34. Guo, J.; Wu, Q.; Sun, L.; Sheng, H. Lap-slip model of rebar-to-concrete in RC/ECC/UHPC based on GA-BP neural network. *Case Stud. Constr. Mater.* **2024**, *20*, e03287. [CrossRef]
35. Zhang, Y.; Gao, X.; Katayama, S. Weld appearance prediction with BP neural network improved by genetic algorithm during disk laser welding. *J. Manuf. Syst.* **2015**, *34*, 53–59. [CrossRef]
36. Li, S.; Xie, Y.; Wang, W. Application of AdaBoost_BP neural network in prediction of railway freight volumes. *Jisuanji Gongcheng Yu Yingyong (Comput. Eng. Appl.)* **2012**, *48*, 6.
37. Liang, J.; Du, X.; Fang, H.; Li, B.; Wang, N.; Di, D.; Xue, B.; Zhai, K.; Wang, S. Intelligent prediction model of a polymer fracture grouting effect based on a genetic algorithm-optimized back propagation neural network. *Tunn. Undergr. Space Technol.* **2024**, *148*, 105781. [CrossRef]
38. Zhang, X.-Q.; Cheng, Q.-L.; Sun, W.; Zhao, Y.; Li, Z.-M. Research on a TOPSIS energy efficiency evaluation system for crude oil gathering and transportation systems based on a GA-BP neural network. *Pet. Sci.* **2024**, *21*, 621–640. [CrossRef]
39. Li, B.; Shen, L.; Zhao, Y.; Yu, W.; Lin, H.; Chen, C.; Li, Y.; Zeng, Q. Quantification of interfacial interaction related with adhesive membrane fouling by genetic algorithm back propagation (GABP) neural network. *J. Colloid Interface Sci.* **2023**, *640*, 110–120. [CrossRef] [PubMed]
40. Li, C.; Xiao, Y.; Xu, X.; Chen, Z.; Zheng, H.; Zhang, H. Intelligent Forecast Model for Project Cost in Guangdong Province Based on GA-BP Neural Network. *Buildings* **2024**, *14*, 3668. [CrossRef]
41. Chicco, D.; Warrens, M.J.; Jurman, G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Comput. Sci.* **2021**, *7*, e623. [CrossRef]
42. *GB/T 36344-2018*; Information Technology—Data Quality Evaluation Indicators. Standards Press of China: Shanghai, China, 2018.
43. *DB61/T 1809-2024*; Guidelines for Data Value Assessment. Standards Press of China: Shanghai, China, 2024.
44. Cohen, I.; Huang, Y.; Chen, J.; Benesty, J. Pearson correlation coefficient. In *Noise Reduction in Speech Processing*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 1–4.
45. Henderi, H.; Wahyuningsih, T.; Rahwanto, E. Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer. *Int. J. Inform. Inf. Syst.* **2021**, *4*, 13–20. [CrossRef]

46. Moré, J.J. The Levenberg-Marquardt algorithm: Implementation and theory. In *Numerical Analysis, Proceedings of the Biennial Conference, Dundee, Scotland, 28 June–1 July 1977*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 105–116.
47. Zhu, C.; Guo, B.; Zhang, Z.; Zhong, P.; Lu, H.; Sigama, A. Determining Rock Joint Peak Shear Strength Based on GA-BP Neural Network Method. *Appl. Sci.* **2024**, *14*, 9566. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

symmetry

MDPI

Article

Laser Stripe Segmentation Network Based on Evidential Uncertainty Theory Modeling Fine-Tuning Optimization Symmetric Algorithm

Chenbo Shi [1], Delin Wang [1], Xiangyu Zhang [1], Chun Zhang [1], Jia Yan [2], Changsheng Zhu [1,*] and Xiaobing Feng [2,*]

[1] College of lntelligent Equipment, Shandong University of Science and Technology, Taian 271019, China; skd996523@sdust.edu.cn (C.S.); 202383230017@sdust.edu.cn (D.W.); 202383230063@sdust.edu.cn (X.Z.); 13811989049@163.com (C.Z.)
[2] Beijing Botsing Technology Co., Ltd., Beijing 100176, China; yj@botsing.com
* Correspondence: zcs@sdust.edu.cn (C.Z.); fxb@botsing.com (X.F.)

Abstract: In welding applications, line-structured-light vision is widely used for seam tracking, but intense noise from arc glow, spatter, smoke, and reflections makes reliable laser-stripe segmentation difficult. To address these challenges, we propose EUFNet, an uncertainty-driven symmetrical two-stage segmentation network for precise stripe extraction under real-world welding conditions. In the first stage, a lightweight backbone generates a coarse stripe mask and a pixel-wise uncertainty map; in the second stage, a functionally mirrored refinement network uses this uncertainty map to symmetrically guide fine-tuning of the same image regions, thereby preserving stripe continuity. We further employ an uncertainty-weighted loss that treats ambiguous pixels and their corresponding evidence in a one-to-one, symmetric manner. Evaluated on a large-scale dataset of 3100 annotated welding images, EUFNet achieves a mean IoU of 89.3% and a mean accuracy of 95.9% at 236.7 FPS (compared to U-Net's 82.5% mean IoU and 90.2% mean accuracy), significantly outperforming existing approaches in both accuracy and real-time performance. Moreover, EUFNet generalizes effectively to the public WLSD benchmark, surpassing state-of-the-art baselines in both accuracy and speed. These results confirm that a structurally and functionally symmetric, uncertainty-driven two-stage refinement strategy—combined with targeted loss design and efficient feature integration—yields high-precision, real-time performance for automated welding vision.

Keywords: laser stripe segmentation; uncertainty modeling; fine-tuning optimization; welding automation; line structured-light

1. Introduction

With the welding manufacturing industry advancing towards higher precision, greater efficiency, and enhanced intelligence, welding automation has emerged as one of its cornerstone technologies. It is widely applied in various industrial sectors, including automotive manufacturing, shipbuilding, aerospace, and others. However,constrained by both objective factors (e.g., harsh welding environments) and subjective limitations in technical expertise, traditional manual welding methods are no longer sufficient to meet the demands of modern manufacturing. In light of rapid advancements in robotics and artificial intelligence, intelligent welding has become a critically important innovation in industrial production. Under this trend, intelligent welding robots have arisen as powerful enablers, significantly improving weld quality and operational efficiency [1].

Equipped with visual sensing technology, intelligent welding robots achieve fully automated, intelligent, and unattended welding [2]. Among various vision-based sensing methods, active vision technologies based on laser structured light have become predominant in welding applications due to their non-contact operation, high energy density, and exceptional accuracy [3]. The schematic diagram illustrating the welding work process is presented in Figure 1. The principle of this approach involves projecting a laser line pattern onto the workpiece surface, capturing the deformed pattern with a camera, and then applying laser triangulation techniques to accurately locate and reconstruct the geometrical parameters of the target [4].

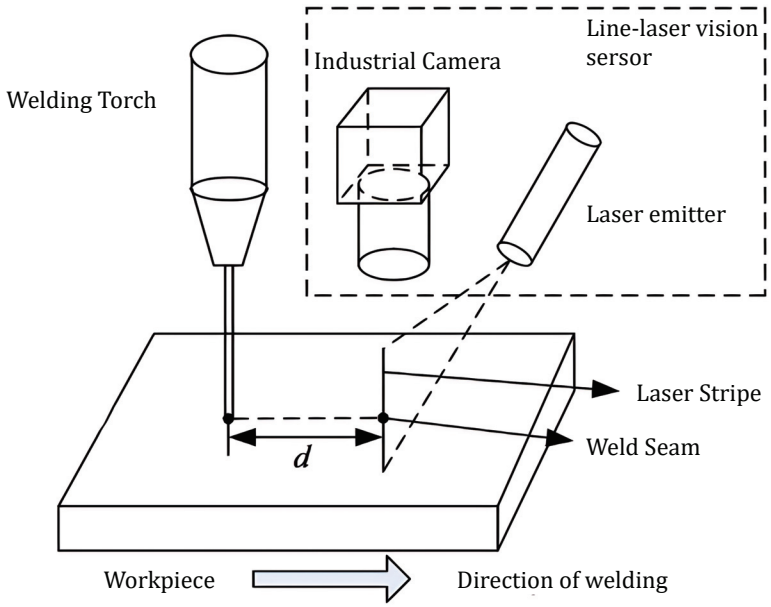


Figure 1. Schematic diagram of the welding process: the welding torch and the laser stripe must maintain a fixed distance dd, and this relative distance remains constant during robot motion.

Within such systems, laser-line extraction serves as the foundational step for subsequent tasks, including seam tracking, groove recognition, and precise positioning. Welding environments differ markedly from other industrial settings: persistent and severe interference noise—such as arc light, spatter, slag, smoke, intense reflections, and rapid dynamic changes—renders laser-line extraction particularly challenging, constituting the primary bottleneck in system robustness and accuracy [5]. Additionally, given the real-time requirements of welding tasks, algorithmic processing speed and computational resource constraints are critical considerations [6].

Extensive research has focused on developing advanced laser-stripe segmentation techniques for complex welding scenarios [7–9]. These techniques can be broadly categorized into traditional methods and deep-learning-based approaches. Traditional segmentation workflows typically consist of three sequential stages. First, optical hardware optimizations and filter design are employed to physically suppress strong arc light and spatter interference (e.g., installing specialized optical filters on the camera [10]). Second, image denoising procedures customized to the welding environment remove residual noise, yielding a cleaner laser image (e.g., scene-specific filters and morphological operations). Finally, region-based segmentation separates the laser stripe from the background (e.g., thresholding or Otsu's method for maximizing inter-class variance [11]). Ye et al. [12] reviewed an image preprocessing-based denoising pipeline that employs morphological opening followed by Otsu's thresholding to remove spurious noise and produce a binarized image, thereby enhancing the laser-stripe features in V-groove weld images. Wu et al. [13]

conducted a comparative study of three filters—Gaussian, median, and Wiener—and demonstrated that the median filter offers superior feature enhancement for laser-stripe extraction. Zhang et al. [14] proposed an image restoration algorithm that combines vector quantization (VQ) with a synthetic fractal-noise model, maintaining stripe continuity and subpixel accuracy even under severe noise conditions. Liu et al. [15] installed a narrowband band-pass filter matched to the laser wavelength on a CCD camera to physically suppress arc light and stray reflections. Zeng et al. [16] employed an LED array light source with a diffusing enclosure and custom top-and-bottom apertures to project uniform illumination onto the workpiece surface, effectively overcoming strong specular reflections. Gu et al. [17] modified a binocular camera system for synchronous image capture and added a 650 nm cut-off filter and welding-specific protective glass to the lenses; by using short-exposure triggering, they successfully suppressed intense arc-light interference.

Although traditional segmentation workflows can achieve acceptable results under limited noise conditions, they depend heavily on expert-tuned parameters—such as denoising filter settings and binarization thresholds—that must be recalibrated for each new welding setup. As a result, their performance degrades markedly when faced with unseen arc intensities, spatter distributions, or material reflectivities, leading to poor generalization and unstable operation in complex, dynamic welding environments.

In recent years, to effectively address these limitations, deep-learning-based image-segmentation techniques, such as UNet [18] and SegNet [19], have been increasingly utilized in laser-stripe segmentation tasks to enhance robustness in challenging welding environments. By leveraging the advanced learning and inference capabilities of neural networks, these methods significantly improve segmentation accuracy. For instance, Yang et al. [20] proposed a weld-image denoising network that exploits the contextual feature-learning capacity of a deep convolutional neural network (DCNN) to recover laser stripes from heavy noise interference. Zhao et al. [21] employed ERFNet [22] to extract laser stripes under strong background noise and subsequently derived the centerline and its corresponding feature points. Wang et al. [23] introduced WFENet, a semantic segmentation network enhanced with channel-attention mechanisms, achieving both rapid and accurate segmentation of laser-weld features. Chen et al. [24] presented DSNet, an efficient hybrid architecture combining convolutional neural networks and Transformers for real-time seam segmentation. Despite their promising results, these approaches still face an inherent trade-off between segmentation speed and accuracy. In practical welding applications, where image processing is typically conducted on embedded devices with limited computational resources and memory, it is essential to jointly optimize precision, real-time performance, and overall network efficiency.

Existing laser-vision welding systems often rely on off-the-shelf sensors whose optical configurations are not optimized for the severe interferences encountered in real welding (e.g., arc glow, spatter). Consequently, they exhibit unstable stripe capture under high-intensity noise. To address this, we design a custom line-laser vision sensor (Section 2) that integrates a band-pass filter and optimized projection geometry, ensuring robust, hardware-level suppression of environmental noise. Moreover, existing public datasets for laser-stripe segmentation are limited in size and noise diversity, hindering generalization to complex welding scenarios. We therefore collect and annotate a novel dataset of 3100 real-weld images encompassing multiple noise types (arc glow, spatter, slag, smoke, specular reflection), detailed in Section 4.1.

The primary contributions of this work are summarized as follows:

(a) We design a custom laser-structured-light vision sensor that ensures hardware-level reliability for weld-stripe image acquisition in complex welding environments.

(b) We collect and curate a novel weld-stripe image dataset, captured under real-world welding conditions and encompassing multiple types of noise (arc glow, spatter, slag, smoke, and specular reflection).
(c) We propose a real-time two-stage evidential fine-tuning network for laser-stripe segmentation. By modeling evidential uncertainty and introducing a targeted optimization mechanism, our single-branch, reduced-channel architecture achieves a well-balanced trade-off between segmentation accuracy and inference speed.

The remainder of this paper is organized as follows. Section 2 elaborates on the design of the line-laser vision welding system. Section 3 presents the architecture and training strategy of the Laser Stripe Segmentation Network. Section 4 provides details regarding the construction of our Weld Seam Laser Image Dataset and evaluates our method through comparative experiments and ablation studies. Section 5 discusses the impact of uncertainty modeling and structural symmetry on segmentation performance. Finally, Section 6 concludes the paper and outlines potential directions for future research.

2. Line-Laser Vision Welding System

As shown in Figure 2, The line-laser vision sensor mounted on an intelligent welding robot system serves as the "eye" of the robot, precisely detecting and measuring the three-dimensional coordinates and geometric parameters of the weld seam. The sensor assembly comprises a laser generator, an industrial fixed-focus camera, a controller, and actuation components, all rigidly affixed to the front of a crawling welding robot. As a non-contact device, it maintains a fixed offset from the welding torch, thereby eliminating interference from external reference points on the workpiece.

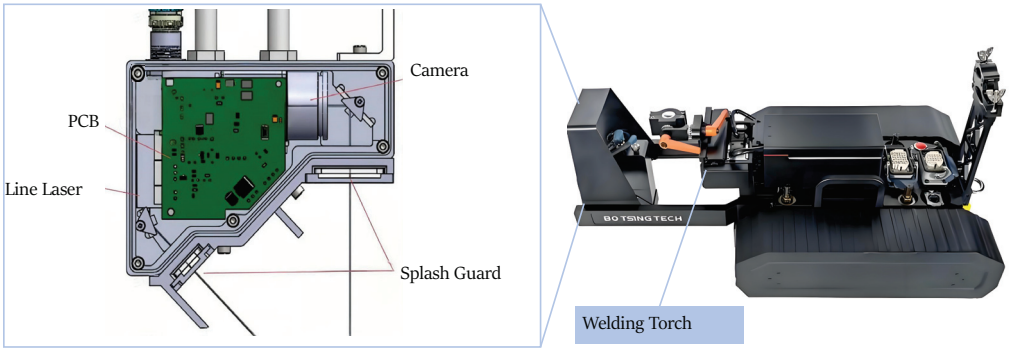


Figure 2. Line-laser vision sensor: the sensor is rigidly mounted to both the robot and welding torch; an industrial camera and laser projector serve as the core components, with their relative positions fixed and the angle between the camera's optical axis and the laser line maintained constant.

To meet the stringent requirements of harsh welding environments, the optical design of the sensor must mitigate specular reflections while ensuring clear visibility of workpiece features. To achieve this, we utilize an MV-CB016-10GM-S industrial camera coupled with a custom AZURE-1280MAC+BP635 fixed-focus lens (both produced by HIKVISION in Hangzhou, China) which incorporates a 635 nm band-pass filter, as well as a specialized JBP6520Z-052 laser generator (produced by Laser-tech in Zhuhai, China). This hardware configuration effectively filters out environmental noise and reliably captures laser-stripe images, even under high-intensity and high-interference welding conditions, thus providing robust input for subsequent image processing and 3D weld reconstruction.

The operation of a laser-sensor–based weld-tracking system is described as follows [25]: The projected laser stripe undergoes deformation (e.g., bending or indentation) when it encounters the weld groove. The camera captures these deformed stripe images, and image-processing algorithms extract key feature points along the stripe. Subsequently,

by mapping these points to a mathematical model, the system determines the spatial position, profile shape, width, and height of the weld. The Schematic diagram of coordinate transformation is shown in Figure 3.

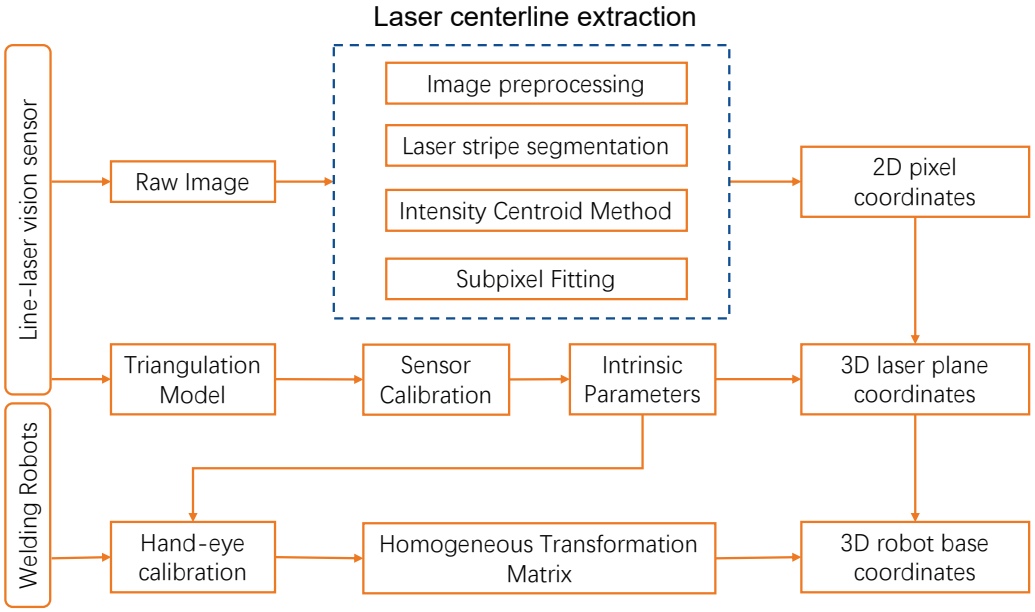


Figure 3. The Schematic diagram of coordinate transformation: Through camera calibration, laser-plane calibration, and hand–eye calibration algorithms, pixel coordinates in the laser-stripe image are transformed into three-dimensional points in the robot's base coordinate frame.

3. Laser Stripe Segmentation Network

To enhance the detectability of laser-stripe regions in noisy welding environments, this study formulates the task of laser-stripe regions extraction as a semantic segmentation problem. By leveraging the pixel-level classification capability of a segmentation network, each image pixel is accurately mapped to its corresponding class mask.

3.1. Network Architecture

As illustrated in Figure 4, we present a symmetrically structured, two-stage semantic segmentation framework that integrates evidential uncertainty modeling with a specialized fine-tuning module. In Stage I, a training-only Transformer branch runs in parallel with a lightweight CNN to extract coarse features; at inference, this branch is pruned so that only the CNN remains, ensuring maximum throughput. The Evidential Uncertainty Head then applies a 1×1 convolution and ReLU to produce nonnegative "evidence" vectors for each pixel, parameterizing a Dirichlet distribution and yielding both a coarse seed mask and its paired uncertainty map—two outputs that are inherently one-to-one and spatially symmetric. In Stage II, a mirror-image fine-tuning network uses the uncertainty map to guide symmetric refinement of the same pixel regions, preserving stripe continuity while correcting errors identified in Stage I. This explicit correspondence between uncertainty estimation and targeted refinement embodies the architectural symmetry at the heart of our design.

The Fine-tuning Network is specifically designed for detailed correction. Fast Fourier Convolutions (FFC) [26] extract frequency-domain features to capture stripe continuity and noise characteristics. An enhanced lightweight dual-branch SimECA module, guided by the uncertainty mask, selectively suppresses errors in high-uncertainty regions while reinforcing coherent textures in low-uncertainty areas.

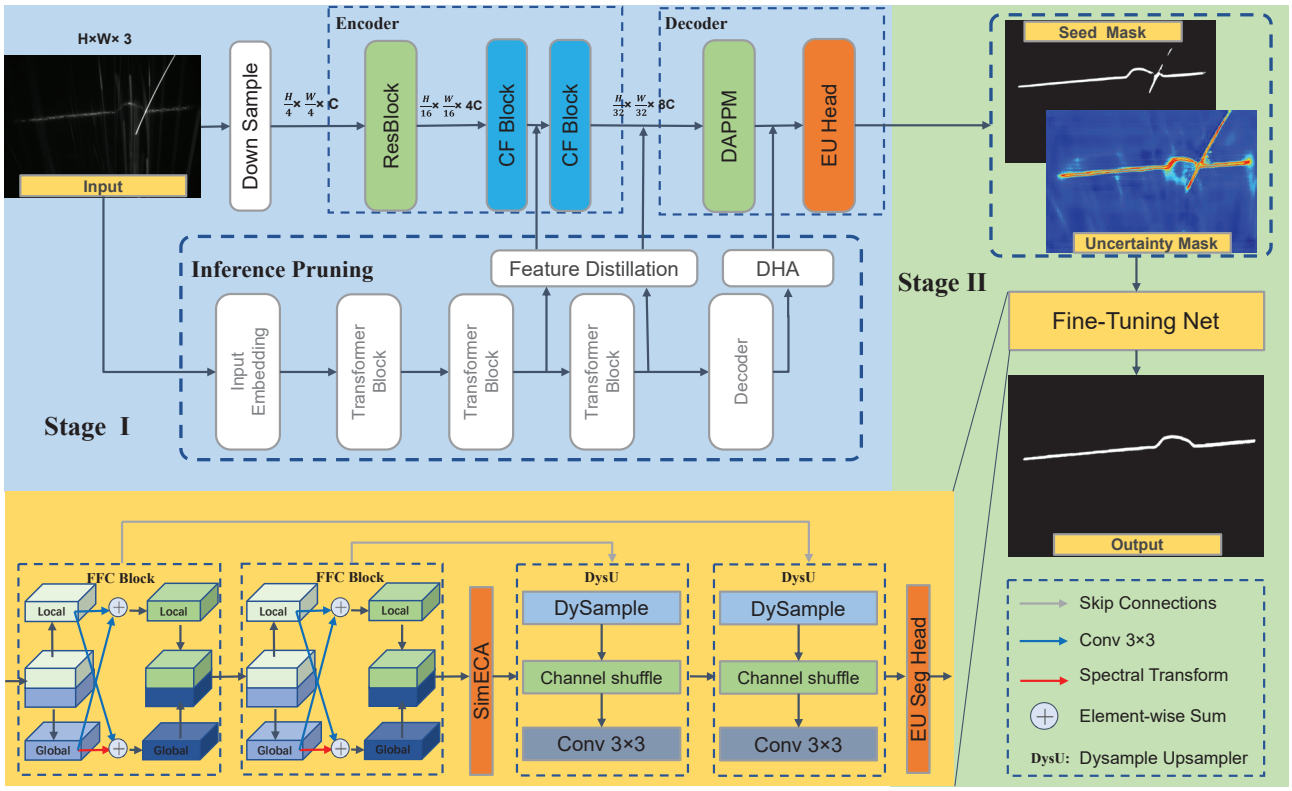


Figure 4. Architecture of EUFNet: a two-stage network in which Stage I performs evidential uncertainty quantification to generate a seed mask and an uncertainty mask, and Stage II uses these masks as inputs to a lightweight fine-tuning network that focuses optimization on high-uncertainty regions.

3.2. Stage I: Fast Segmentation Network

In Stage I, the raw laser-stripe image is initially processed by a single-branch convolutional neural network for coarse stripe extraction. To expedite inference, we utilize a multi-layer stacked CNN backbone inspired by SCTNet [27]. Additionally, to enrich semantic representations, we introduce a training-only Transformer branch that captures high-quality contextual features. An enhanced Conv-Former Block (CFBlock) integrates these semantic features back into the CNN branch, resulting in a lightweight, real-time segmentation network that combines efficient convolutional processing with rich Transformer-derived semantics, the backbone begins with two successive 3×3 convolution–batch norm–ReLU layers that increase the channel dimension from 3 to 32, followed by a 1×1 convolution that raises it to 64. This is followed by two residual blocks, each comprising two 3×3 convolution–batch norm–ReLU layers with skip connections, and then two Conv-Former blocks that project features to 128 channels, process them through a four-head Transformer encoder, and project back to 64 channels. The decoder is a Deep Aggregation Pyramid Pooling Module that fuses features from four parallel 3×3 convolutions and concludes with a 1×1 convolution. The segmentation head consists of one 3×3 convolution reducing channels to 32 and a final 1×1 convolution producing the 2 output classes.

Specifically, the network begins with an initial downsampling block comprising two stacked 3×3 standard 2D convolutions followed by a 1×1 convolutional bottleneck, which increases channel depth while reducing spatial resolution. Four consecutive residual blocks subsequently perform feature extraction: the first two blocks employ convolutional layer stacks with skip connections to efficiently capture local details within a limited receptive field, whereas the latter two are carefully designed CFBlocks that emulate Transformer behavior to capture long-range dependencies and semantic context.

For multi-scale feature fusion, we adopt the Deep Aggregation Pyramid Pooling Module (DAPPM) [28] as the decoder. DAPPM hierarchically aggregates contextual information across different scales through multiple 3×3 convolutions in a residual manner, concatenates all scale-specific feature maps, and compresses the channel dimension using a final 1×1 convolution. Since this aggregation operates at low resolution, it introduces negligible inference overhead. A lightweight segmentation head then generates the coarse mask output of Stage I.

3.3. Evidential Uncertainty Head

Currently, our segmentation network generates only a deterministic mask and lacks the capability to quantify its confidence in that prediction. In classification tasks, it is common practice to apply a softmax function at the final layer to map network outputs to class-probability distributions; however, these probabilities do not accurately reflect the model's true confidence in its predictions. To obtain genuinely reliable confidence estimates, the prevailing approach involves introducing Bayesian inference [29], such as constructing a Bayesian Neural Network (BNN) [30] or employing Monte Carlo Dropout (MC Dropout) [31] during inference to approximate Bayesian sampling. Nevertheless, BNNs typically entail prohibitively high training and computational costs, while MC Dropout's requirement for multiple forward passes imposes significant demands on memory and compute resources—limitations that impede their practical adoption in real-world systems.

To introduce uncertainty estimation more efficiently, and inspired by evidential deep learning (EDL) [32], we augment the backbone's feature map with an Evidential Uncertainty Head (EUHead). For each pixel, EUHead predicts nonnegative "evidence" values e_k for each class via a Softplus activation ($\text{Softplus}(x) = \ln(1 + e^x)$), ensuring both numerical stability and strictly positive evidence required to parameterize the Dirichlet concentration parameters ($\alpha_k = e_k + 1$). These evidence values are subsequently transformed into the concentration parameters of a Dirichlet distribution according to $\alpha_k = e_k + 1$, assuming a uniform prior of 1 for all classes. Consequently, each pixel is associated with a Dirichlet parameter vector α.

We quantify epistemic uncertainty at each pixel using the "vacuity" metric, which is defined as:

$$\begin{aligned} \frac{K}{S} &= \text{vacuity;} \\ S &= \sum_{k=1}^{K} \alpha_k \end{aligned} \tag{1}$$

where K indicates the number of classes, and S represents the cumulative total of mass evidence. A larger vacuity value indicates less accumulated evidence and thus greater model uncertainty. The resulting per-pixel vacuity heatmap provides a direct visualization of the network's confidence in its semantic predictions. Since vacuity heatmaps may vary in scale across different samples, we apply per-image min–max normalization to map vacuity values into a fixed range. This ensures consistent numeric scales for subsequent processing. During the stage II training, the normalized uncertainty map is incorporated as a pixel-wise weighting factor in the loss function. By assigning higher weights to pixels with greater vacuity, the network is encouraged to focus on uncertain or noisy regions while mitigating the influence of unreliable areas. Consequently, this enhances the robustness and reliability of the final segmentation.

3.4. Stage II: Fine-Tuning Network

In this stage, the seed segmentation mask is fused with the normalized evidential uncertainty map to generate a high-confidence, fine-grained segmentation. To enhance both confidence and the continuity of the laser stripe, we leverage principles from evidential deep learning to design a Refine Network that iteratively optimizes the seed mask using uncertainty cues, it is composed of two Fast Fourier Convolution (FFC) blocks and SimECA modules interleaved. Each FFC block splits its input (64 channels) into a local branch (32 channels with two 3×3 Conv–BN–ReLU layers) and a spectral branch (32 channels processed via Real 2D FFT, 1×1 Conv, inverse FFT), then merges via channel shuffle. Following each FFC, a SimECA attention module applies a 1D-Conv channel attention (C = 64) and a spatial–channel energy map. Finally, a Dynamic Sampling UpSampler (DySample) restores the feature map across two upsampling stages to the full 512×256 resolution, and the output layer consists of a 3×3 Conv $\rightarrow$ 1 channel plus Softplus activation for evidence generation.

The Refine Network is intentionally lightweight, employing fewer channels and a Fast Fourier Convolution (FFC)-based backbone for feature extraction. The structure of Fast Fourier Convolution is illustrated in Figure 5. By integrating FFC in the early layers, the network captures global context, thereby compensating for any long-range dependencies potentially overlooked in the stage I. A specially designed lightweight attention module, combined with an uncertainty-weighted loss function, guides the network to focus on regions of high vacuity while imposing minimal additional computational burden. After multiple layers of FFC and attention refinement, a Dynamic Sampling UpSampler (DySample) [33] performs progressive upsampling. The final convolutional layer is followed by a Softplus activation to guarantee nonnegative outputs, which are used as evidence values in our uncertainty-guided loss function. It dynamically adjusts interpolation weights based on pixel-wise uncertainty and gradually restores the feature map to the original image resolution, producing the final refined segmentation.

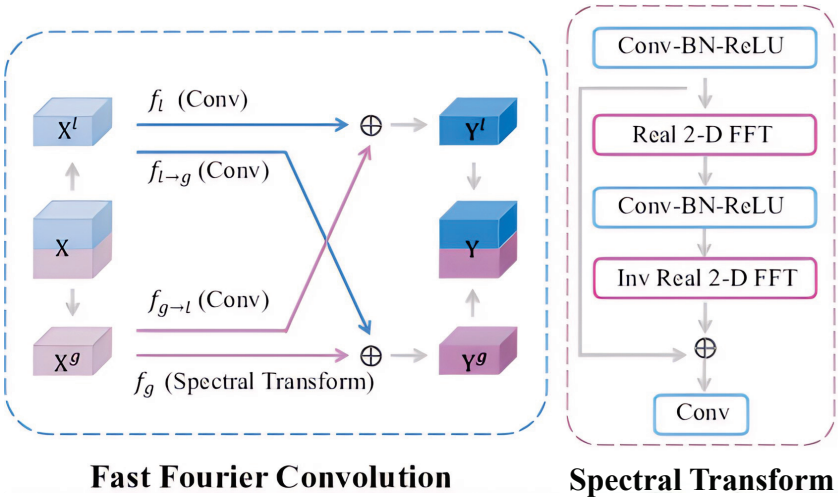


Figure 5. Architecture of FFC: the local branch (top) captures fine-grained semantics and edge cues, while the spectral branch (bottom) extracts global contextual features. The input is first transformed via FFT, followed by point-wise convolution in the frequency domain, and then mapped back to the spatial domain via inverse FFT.

To more effectively leverage global context for uncertainty-driven refinement, we substitute traditional convolutional layers in the backbone with Fast Fourier Convolutions (FFC). Since FFC is fully differentiable, it can be seamlessly integrated into existing deep-learning architectures without requiring significant redesign.

In each FFC operation, the input feature map is first partitioned along the channel dimension into a local branch and a spectral branch. The local branch utilizes standard

convolutional blocks (e.g., 3×3 Conv–BN–ReLU–3×3 Conv) to capture local semantics and edge details. Simultaneously, the spectral branch applies a real 2D fast Fourier transform (Real 2D FFT) to its input X_g , performs point-wise or lightweight convolutions in the frequency domain, and subsequently maps the result back to the spatial domain via an inverse real 2D FFT (Inv Real 2D FFT), thereby extracting global context features. Information exchange between the two branches is facilitated through learnable mappings $f_{l \to g}$ and $f_{g \to l}$, which promote feature sharing and fusion across local and global representations.

A SimECA module is subsequently integrated after the feature extraction stage to further mitigate background interference and enhance boundary and detail responses, effectively addressing residual edge blurring and noise misclassification. The structure of SimECA is illustrated in Figure 6. The dual-branch architecture of SimECA consists of a channel-attention path [34] and a spatial-attention path [35]. Channel-Attention Branch: The input feature map undergoes global average pooling to generate a descriptor of size $(1 \times 1 \times C)$. A one-dimensional convolution with an adaptive kernel size captures local inter-channel dependencies, followed by a sigmoid activation function that produces channel weights of shape $(1 \times 1 \times C)$. These weights are broadcasted to $[B, C, H, W]$ and applied to the feature map, thereby emphasizing channels pertinent to the laser stripe while suppressing irrelevant background channels. Spatial–Channel Joint Branch: For the same input features, per-pixel and per-channel de-meaned squared activations $(x - \mu)^2$ are computed, where μ and σ^2 represent the channel-wise mean and variance, respectively. An energy map is constructed by normalizing these activations with σ^2 and a regularization term (λ), which is then passed through a sigmoid function to produce a three-dimensional attention map of size $[B, C, H, W]$. This map evaluates each neuron simultaneously in both spatial and channel dimensions, significantly enhancing sensitivity to fine stripe-edge details.

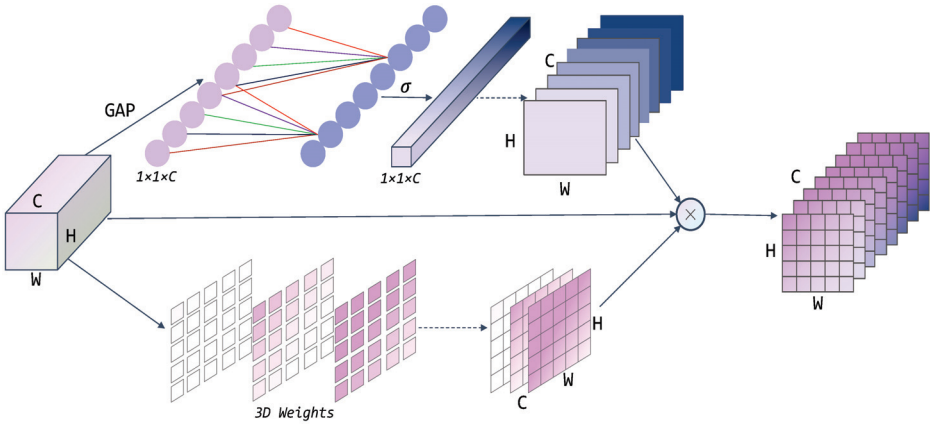


Figure 6. Architecture of SimECA: a parallel dual-branch module comprising achannel-attention branch (top) and aspatial–channel joint branch (bottom). Their outputs are fused with the input feature map to perform feature recalibration.

The two attention maps are fused with the original feature map to recalibrate important pixels at a fine granularity. Given that SimECA introduces only a limited number of one-dimensional convolutional parameters in the channel branch, it incurs negligible computational overhead while substantially improving segmentation accuracy.

Finally, the EUSeg head generates pixel-wise class probabilities. Unlike conventional Softmax, which merely outputs normalized scores, the Dirichlet-based evidential learning framework explicitly models epistemic uncertainty. As previously discussed, the Softmax layer tends to assign overly confident probabilities to erroneous predictions and lacks a mechanism for quantifying the model's "ignorance". In the evidential network, nonnegative evidence is produced through a Softplus activation function and utilized to

parameterize a Dirichlet distribution, from which the predictive class probabilities are subsequently derived:

$$\hat{p}^k = \frac{\alpha_k}{S} \tag{2}$$

Here, α_k denotes the Dirichlet concentration parameter for class K (with K total classes), S represents the total accumulated evidence, and $\hat{p}^k$ signifies the predictive probability for class K.

Consequently, the pixel-wise class probabilities generated by this method more accurately reflect the model's confidence, particularly in unknown or ambiguous regions, thereby producing predictions that are not only more interpretable but also of higher epistemic reliability. The EUSeg Head thus provides segmentation outputs accompanied by meaningful and well-calibrated confidence estimates.

3.5. Loss Function

Since the two stages reflect distinct design philosophies and emphasize complementary aspects, we utilize separate loss functions in each stage to leverage their respective strengths.

In Stage I, we formulate the task as a binary classification problem and accordingly employ the weighted binary cross-entropy loss. Given the significant imbalance between the laser-stripe class (positive) and the background class (negative), the network is prone to prematurely converge toward the dominant background class. To address this issue, we incorporate a weighting factor w for the positive class. Through extensive experimentation, we have found that setting $w = 50$ achieves optimal performance. The loss function for Stage I, denoted as L_1, is defined as follows:

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N}[w y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)] \tag{3}$$

Here, N denotes the total number of pixels, and w represents the weighting factor for positive samples (set to $w = 50$ in our experiments). $y_i \in \{0, 1\}$ is the ground-truth binary label for the i th pixel, and $\hat{y}_i \in \{0, 1\}$ is the network's predicted probability that the i th pixel belongs to the positive class (laser stripe). When $y_i = 1$ (i.e., a positive sample), the corresponding loss term becomes $-w \log(\hat{y}_i)$, which enhances the network's emphasis on stripe pixels and effectively mitigates premature convergence toward the background class.

In Stage II, the Refine network is explicitly designed to focus on regions with low confidence and high uncertainty. To further enhance supervision for pixels exhibiting greater uncertainty, we employ an uncertainty-weighted Focal Loss:

$$\begin{aligned} L_2 &= \frac{1}{N}\sum_{i=1}^{N} w_i[y_i(1 - \hat{y}_i)^\gamma \log \hat{y}_i + (1 - y_i)\hat{y}_i{}^\gamma \log(1 - \hat{y}_i)]; \\ w_i &= \begin{cases} 1 + \frac{u_i - \tau}{1 - \tau} w_{max}, & \mathrm{u}_i > \tau \\ 1, & \mathrm{u}_i \le \tau \end{cases} \end{aligned} \tag{4}$$

Here, γ and τ are two coefficients, set to 2 and 0.5, respectively. Specifically, τ denotes the uncertainty threshold, u_i represents the uncertainty factor, and $w_{\max}$ signifies the maximum additional weight. Consequently, pixels with high uncertainty (u_i large) receive enhanced supervision during training. This uncertainty-driven weighting mechanism dynamically allocates pixel-level attention: regions of high confidence (low uncertainty) are emphasized, while regions of low confidence (high uncertainty) are appropriately down-weighted, thereby preventing noisy or "suspicious" pixels from disproportionately influencing the gradient updates.

The focal-loss exponent γ was set to 2 following the original formulation of Focal Loss, which balances the down-weighting of easy examples against stable training. The uncertainty threshold τ was determined empirically by examining the normalized vacuity distribution on our validation set: setting $\tau = 0.5$ roughly splits pixels into equal low and high uncertainty groups, ensuring that refinement focus is neither too narrow nor too broad. The maximum additional weight $w_{\max}$ was likewise chosen via a small grid search (values in [1.0, 2.0, 3.0]), with $w_{\max} = 1.5$ yielding the best mIoU on validation without destabilizing training. All hyperparameter settings are fixed for all experiments and result in robust, reproducible performance.

During training, the two stages are optimized in a cascaded and end-to-end manner. Consequently, the overall loss function is formally defined as:

$$L = \lambda_1(e)L_1 + \lambda_2(e)L_2 \tag{5}$$

Here, the balancing coefficients λ_1 and λ_2 are initialized to 1 and 0, respectively. As training progresses across epochs, $\lambda_1(e)$ is gradually decayed while $\lambda_2(e)$ is linearly increased, thereby enhancing the effectiveness of the refinement stage. If the Stage I loss L_1 has already converged, the Stage I network is frozen (i.e., its parameters are no longer updated) and further optimization is conducted exclusively for Stage II, with an enlarged λ_2 used for fine-tuning.

4. Experiments and Results

In this section, we carry out a comprehensive series of experiments on our Weld Seam Laser Image dataset to rigorously validate the effectiveness of the proposed method. In Section 4.2, we detail the experimental design and implementation specifics. In Section 4.3, we conduct comparisons of our approach against various baselines to objectively quantify its segmentation performance. Finally, in Section 4.4, we undertake ablation studies to systematically analyze the contributions of our framework and its individual components.

4.1. Weld Seam Laser Image Dataset

The dataset was collected in a real welding environment at the construction site of a large container ship. During welding operations, the line-laser vision sensor mounted on the welding robot continuously acquired grayscale images of the laser stripe on the weld seam. The dataset consists of 3100 laser-stripe images and their corresponding segmentation masks, which were manually annotated using LabelMe (version:5.2.1). These images and labels were randomly divided into training, validation, and test sets following an 8:1:1 ratio.

To capture the diversity of real-world welding tasks, the dataset encompasses weld seams of various sizes and shapes, covering multiple seam types—primarily butt and fillet welds—produced through a complex multilayer, multipass welding process (including root pass, fill, and cap). Additionally, the collected images illustrate typical noise sources encountered in welding scenarios, including arc light and spatter, ambient illumination, specular reflections, slag interference, variable exposure, and particulate matter, as depicted in Figure 7.

Each image in our dataset contains a single, continuous laser stripe that delineates the weld seam and provides the essential geometric cues required for precise seam localization. By accurately isolating this stripe, one can infer critical measurements such as groove width, height, and alignment relative to the welding torch. The continuity of the stripe is particularly important: any breaks or discontinuities in the segmented output directly translate into errors in downstream geometric reconstruction and path-planning algorithms.

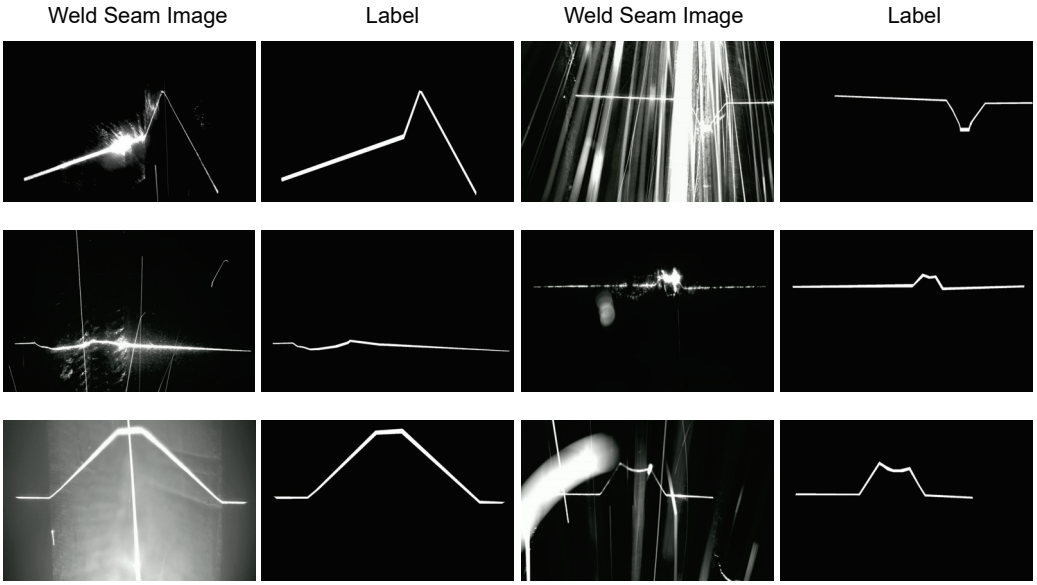


Figure 7. Weld Seam Laser image dataset(WSLI): comprising raw laser-stripe images captured by the line-laser vision sensor and their corresponding annotations generated with the Labelme tool, encompassing a variety of noise types encountered in real welding environments.

However, achieving robust segmentation of the laser stripe is nontrivial in real-world welding scenarios. The welding environment introduces a multitude of confounding factors—intense arc glow, erratic spatter, drifting smoke, and specular reflections—that manifest as high-frequency noise and false positives in captured images. Variations in surface finish, material reflectivity, and ambient illumination further exacerbate the segmentation task by altering stripe contrast and edge definition. Consequently, an effective segmentation framework must not only discriminate the true laser stripe from heterogeneous background disturbances but also preserve its spatial continuity under widely varying conditions.

4.2. Implementation Details

Prior to model training, all annotated images underwent a standardized preprocessing pipeline to ensure spatial consistency and fair comparison across methods. Specifically, every model used the same steps: resize to 512 $\times$ 256 pixels, histogram equalization for contrast normalization, and random horizontal flips for data augmentation. This guarantees that differences in performance arise solely from model architectures and not from disparate input preprocessing. Optimization was performed using the AdamW algorithm, initialized with a learning rate of 4×10^{-4} and a weight-decay coefficient of 1.25×10^{-2} to regularize model complexity. To avoid premature convergence to the dominant background class or entrapment in suboptimal local minima, we adopted the SGDR (Stochastic Gradient Descent with Warm Restarts) protocol for cosine-annealing learning-rate scheduling [36], which periodically resets the learning rate to its initial value and then decays it along a cosine curve. The full training schedule comprised 100 epochs, with a mini-batch size of 2 images, implemented in PyTorch (version:1.10.0) and executed on a single NVIDIA GeForce RTX 3090 GPU (Santa Clara, CA, USA).

For empirical validation, we conducted a comprehensive benchmark of our proposed network against a diverse selection of state-of-the-art segmentation architectures. General semantic-segmentation baselines included U-Net [18] and SegFormer [37], while high-efficiency, real-time models such as ERFNet [22] and SCTNet [26] served as additional comparators. Furthermore, we evaluated two domain-specific methods—WFENet [23] and DSNet [24]—which were explicitly designed for laser-stripe segmentation tasks. Model

performance was assessed along two axes: segmentation fidelity and real-time inference capability. Segmentation accuracy metrics comprised mean intersection-over-union (mIoU) and mean pixel accuracy (mAcc), both derived from the confusion-matrix elements of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN). Real-time suitability was quantified by measuring frames per second (FPS), total parameter count (Params), and computational complexity in giga-floating-point operations per second (GFLOPs). This rigorous evaluation protocol underscores the relative trade-offs between accuracy, model size, and inference speed across contemporary approaches.

4.3. Benchmark Results

Table 1 compares our proposed method against both classical segmentation networks and real-time semantic segmentation architectures, reporting mIoU, mAcc, FPS, GFLOPs, and parameter count. Figure 8 illustrates qualitative results for the laser-stripe extraction task and juxtaposes our outputs with those of competing methods. As shown in Table 1, our approach achieves superior accuracy while maintaining the fastest inference speed: it attains an mIoU of 89.31%, approximately one percentage point higher than the nearest competitor, and operates at 236.73 FPS, outperforming all baselines in terms of real-time throughput. In Figure 8, we present side-by-side comparisons of segmentation results produced by WFENet and DSNet, the two most directly comparable models specifically designed for laser-stripe extraction. Compared to these methods, our network delivers cleaner removal of background noise and more continuous stripe delineation. These advantages can be attributed to our uncertainty modeling and optimization strategy, as well as the specialized stage II refinement network.

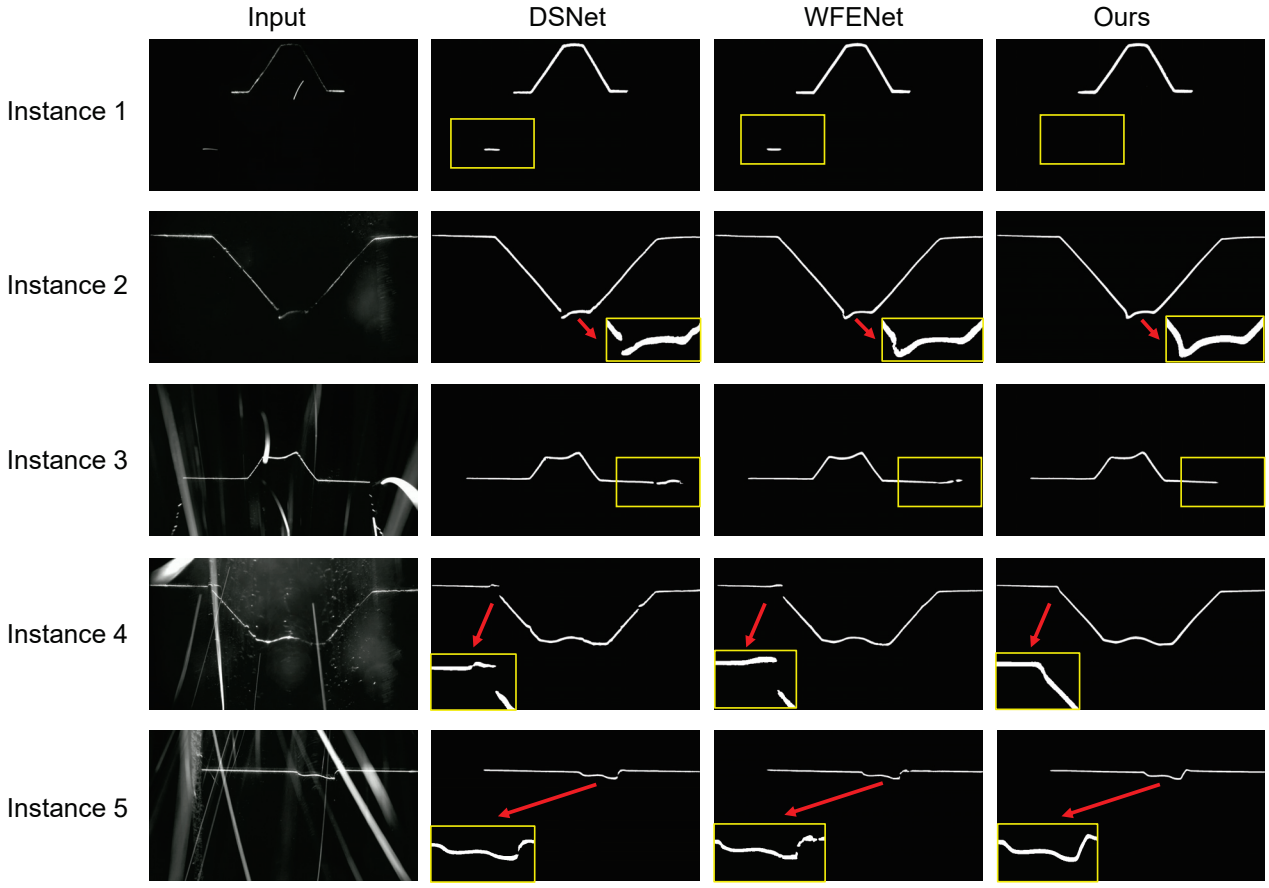


Figure 8. Comparison of segmentation results: compared to DSNet and WFENet, EUFNet produces masks with finer detail (e.g., Instance 5) and better preserves the continuity of the laser stripe (e.g., Instance 4).

Table 1. Comparisons with other methods on the WSLI dataset. The best results are highlighted in bold. FPS refers to frames per second; Params refers to the number of parameters.

Method	mIoU(%)	mAcc(%)	FPS	GFLOPs	Params
UNet [18]	87.91	92.51	10.54	405.82 G	28.99 M
ERFNet [22]	88.36	93.26	87.14	26.86 G	2.06 M
WFENet [23]	88.71	94.82	90.52	22.77 G	**0.75 M**
DSNet [24]	87.61	93.55	223.67	**3.59 G**	1.77 M
SCTNet-S [26]	85.28	91.43	160.38	7.08G	4.65 M
SegFormer-B1 [37]	88.12	93.67	25.33	35.90 G	24.72 M
EUFNet (ours)	**89.31**	**95.85**	**236.73**	5.86 G	2.61 M

To compare the performance of these segmentation models with our proposed EUFNet across different datasets, we additionally evaluated all methods on WLSD [38]. The results, summarized in Table 2, demonstrate that EUFNet achieves robust accuracy on both our custom dataset and WLSD, significantly outperforming competing approaches in terms of generalization capability. This superiority can be attributed to our uncertainty-driven training framework: Fast Fourier Convolution blocks serve as the backbone, enabling the extraction of rich contextual information at shallow network stages with a reduced channel count, while the proposed dual-branch SimECA module refines critical details with fine-grained precision.

Table 2. Comparative experiments on WLSD test set against other methods.The best results are highlighted in bold.

Method	mIoU (%)	mAcc (%)	FPS	GFLOPs	Params
WFENet [23]	87.76	92.98	90.52	22.77 G	**0.75 M**
DSNet [24]	84.37	90.40	223.67	**3.59 G**	1.77 M
SCTNet-B [26]	86.58	93.08	144.92	28.03 G	17.43 M
SegFormer-B0 [37]	87.80	93.36	54.21	12.77G	3.72 M
EUFNet (ours)	**88.45**	**95.12**	**236.73**	5.86 G	2.61 M

4.4. Ablation Study

To validate the effectiveness of our proposed architecture and its components, we conducted ablation studies, with the results presented in Tables 3 and 4.

As shown in Table 3, incorporating the Transformer branch during training enhances semantic representation, resulting in an increase of 0.75% in mIoU and 0.66% in mAcc. At inference time, only the single-branch CNN is retained, thereby preserving real-time performance. With the integration of evidential uncertainty modeling and optimization, segmentation accuracy improves significantly. Our fine-tuning Network, which is specifically designed to fine-tune low-confidence regions, further boosts performance, achieving a 1.17% gain in mIoU and a 0.61% gain in mAcc. Finally, the EUHead module, which quantifies pixel-level uncertainty, contributes an additional 1.02% gain in mIoU and 1.83% gain in mAcc.

Table 3. Ablation study on the architecture. ✓ indicates the use of this module.

Transformer Branch	EU Head	Fine-Tuning	mIoU (%)	mAcc (%)
			86.37	92.75
✓			87.12 (+0.75)	93.41 (+0.66)
✓		✓	88.29 (+1.17)	94.02 (+0.61)
✓	✓	✓	89.31 (+1.02)	95.85 (+1.83)

We assessed the impact of the components introduced in Table 4. Our baseline architecture replaces the standard residual convolution blocks with Fast Fourier Convolutions (FFC) but excludes any attention mechanism. Replacing traditional residual units with FFC results in a 0.85% increase in mIoU. Subsequently, we integrate the lightweight, dual-branch SimECA module. This addition further enhances mIoU by 1.27%, with minimal impact on inference speed.

Table 4. Ablation Study on component.

Components	mIoU (%)	mAcc (%)
Baseline	87.19	92.74
+FFC	88.04 (+0.85)	94.07 (+1.33)
+SimECA	89.31 (+1.27)	95.85 (+1.78)

Figure 9 illustrates the segmentation results at different stages. In Figure 9a, we present three laser-stripe images of varying shapes captured at distinct time points during welding: (i) a laser image during the root pass of a fillet weld, (ii) a laser image during the fill pass of a butt weld, and (iii) a laser image with backing material. Figure 9b displays the corresponding evidential uncertainty masks generated by the EUHead, where regions with strong noise interference and along stripe edges exhibit very high uncertainty—indicating that the model lacks confidence in these areas. As a result, the coarse segmentation outputs from Stage I (Figure 9c) contain defects in these uncertain regions. After refinement, the Refine Network leverages uncertainty quantification to optimize these areas, producing high-confidence segmentation maps as shown in Figure 9d.

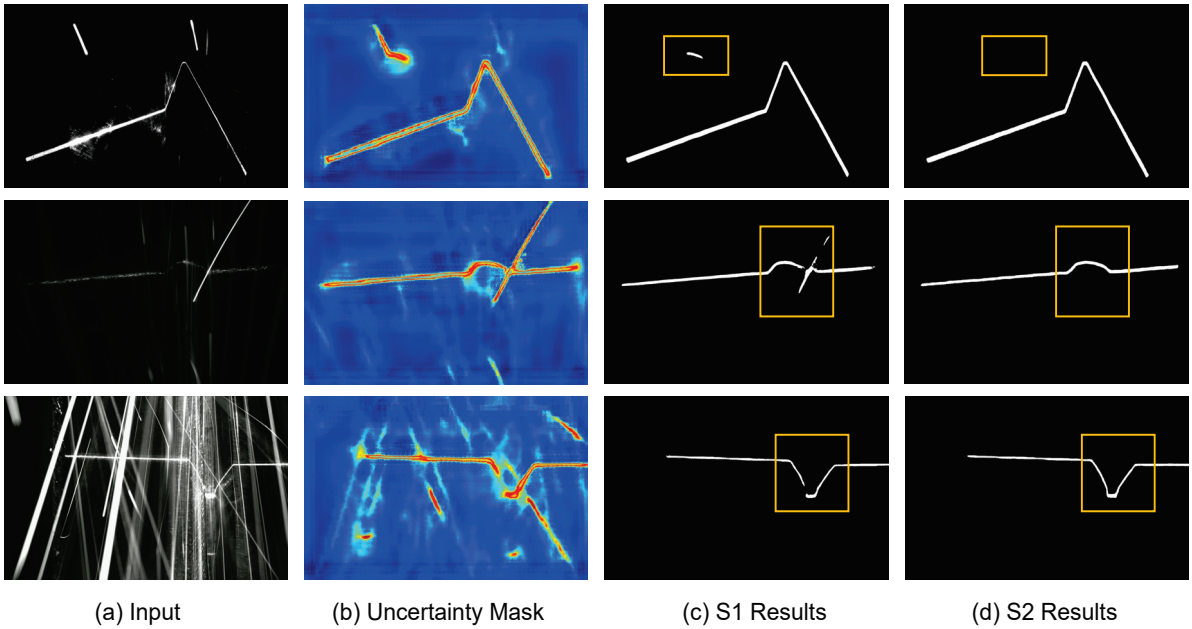


Figure 9. Comparison of the segmentation results of Stage I and Stage II: (**a**) Input Images; (**b**) Evidential Uncertainty Quantitative Mask; (**c**) Seed Mask from Stage I; (**d**) Segmentation Results from Stage II. The subtle differences are shown within the yellow box.

4.5. Comprehensive Analysis

To achieve efficient performance on embedded devices with limited computational and memory resources, our method meticulously balances accuracy and real-time requirements. After training the model on an NVIDIA GeForce RTX 3090 GPU, we deployed it to an embedded system. Experimental results demonstrate that, when the seam-tracking laser

software is fixed at 65 FPS, our approach can reliably extract laser stripes even in highly disturbed environments, fully satisfying the real-time demands of welding applications.

5. Discussion

The experimental results demonstrate that EUFNet's uncertainty-guided two-stage framework substantially outperforms single-stage and uniform-refinement baselines on both our custom WSLI dataset and the public WLSD benchmark. By generating pixel-wise vacuity maps in Stage I and using them to weight the loss and guide the fine-tuning network, EUFNet concentrates its learning capacity on the most ambiguous stripe regions—such as boundary edges corrupted by arc glow, spatter, or specular reflections—thereby reducing both false positives and false negatives. This targeted supervision corrects coarse-stage errors more effectively than simply increasing model depth or channel counts, as evidenced by the +1.7% mIoU gain over DSNet and improved stripe continuity in heavily noisy areas (Figure 8).

In the welding scenario, achieving a balance between high frame rate and high precision is a major challenge. DSNet achieved high-speed inference at 100 FPS by significantly compressing the model size (reducing the number of parameters by 54 times), but this compact design limited its expressive power. In contrast, EUFNet, by leveraging multi-scale feature fusion and uncertainty assessment techniques, has significantly enhanced the segmentation accuracy even with a slight increase in parameters and computational load. Experiments show that EUFNet is significantly superior to DSNet in terms of weld seam positioning accuracy, while still meeting the real-time processing requirements. We also recognize that DSNet has fewer parameters and higher efficiency. In the future, we can further optimize EUFNet by combining the lightweight strategy of DSNet.

To provide further insight into EUFNet's superiority, Table 5 compares Precision, Recall, and F1-score across models. EUFNet achieves the highest Recall (94.65%) and the best F1-score (90.82%), indicating its balanced capability to detect true stripe pixels while minimizing false alarms. In contrast, baseline methods like UNet and WFENet exhibit lower F1-scores (88.55% and 88.24%, respectively), reflecting their difficulty in maintaining both high precision and recall under challenging noise conditions.

Table 5. Comparison of segmentation methods on the WSLI dataset, reporting Precision, Recall, and F1-score for each model. The best results are highlighted in bold.

Method	Precision (%)	Recall (%)	F1-Score (%)
UNet [18]	84.91	92.51	88.55
ERFNet [22]	85.36	92.26	88.68
WFENet [23]	85.81	90.82	88.24
DSNet [24]	86.04	91.31	88.60
SCTNet-S [26]	84.83	91.43	88.01
SegFormer-B0 [37]	87.14	93.67	90.29
EUFNet (ours)	**87.28**	**94.65**	**90.82**

Beyond loss weighting, the integration of Fast Fourier Convolutions and the dual-branch SimECA module in Stage II enables EUFNet to capture global stripe structure and selectively enhance salient features while maintaining real-time performance. FFC blocks efficiently encode long-range dependencies that help preserve stripe continuity across the image, and SimECA's lightweight channel–spatial attention further suppresses background interference with minimal computational overhead. Together, these components allow EUFNet to achieve 236 FPS on an RTX 3090—faster than DSNet—despite its added uncertainty mechanisms, confirming that intelligent architectural choices can reconcile high accuracy with low latency.

6. Conclusions

In this work, to address the severe noise interference encountered in real-world welding environments, we propose EUFNet, a symmetrically structured two-stage real-time semantic segmentation network grounded in evidential deep learning (EDL). Our framework adopts a mirror-image "coarse segmentation–uncertainty quantification–fine-tuning" paradigm in which Stage I and Stage II form a functionally and structurally symmetric pair: the uncertainty map and coarse segmentation mask are produced in one-to-one correspondence, and then symmetrically refined in the second stage. By integrating self-distillation, Fast Fourier Convolution (FFC), and the SimECA module, EUFNet efficiently extracts multi-scale features and generates high-confidence laser-stripe maps in real time. This explicit architectural and operational symmetry not only enhances robustness against extreme noise but also ensures precise groove recognition and localization for automated robotic welding systems.

Moreover, we have constructed and publicly released a shipyard-scale weld-stripe image dataset consisting of 3100 grayscale images with varying stripe shapes and sizes. Each image is manually annotated and exhibits common welding interferences such as arc glow, spatter, slag, smoke, and specular reflections. This dataset serves as a comprehensive benchmark for training and evaluating segmentation models under realistic conditions.

Extensive experiments demonstrate that EUFNet, empowered by evidential uncertainty modeling and its targeted optimization mechanism, successfully harmonizes self-distillation, FFC, and SimECA to achieve simultaneous improvements in accuracy and inference speed—establishing a new standard for real-time laser-stripe extraction. While EUFNet delivers exceptional performance and runtime efficiency on our dataset, its reliance on a Transformer branch during training imposes significant demands on computational resources and memory. Future work will focus on further parameter reduction to better accommodate resource-constrained embedded platforms.

Author Contributions: Conceptualization, C.S., D.W., C.Z. (Chun Zhang), J.Y., C.Z. (Changsheng Zhu) and X.F.; methodology, D.W.; software, J.Y.; validation, D.W. and X.Z.; formal analysis, D.W.; investigation, D.W.; resources, C.Z. (Chun Zhang); data curation, D.W., J.Y. and X.Z.; writing—original draft preparation, D.W.; writing—review and editing, D.W.; visualization, D.W.; supervision, C.S., C.Z. (Chun Zhang) and C.Z. (Changsheng Zhu).; project administration, C.S. and C.Z. (Chun Zhang); funding acquisition, C.S. and X.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Projects of Scientific and Technological Innovation in Daiyue District, Tai'an City, grant number CXXM–2021006.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Acknowledgments: Thanks to Beijing Botsing Technology Co., Ltd. for providing the laser sensors experimental equipment and welding test.

Conflicts of Interest: Author Jia Yan was employed by the company Beijing Botsing Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ali, R.; El-Betar, A.; Magdy, M. A review on optimization of autonomous welding parameters for robotics applications. *Int. J. Adv. Manuf. Technol.* **2024**, *134*, 5065–5086. [CrossRef]
2. Wang, J.; Li, L.; Xu, P. Visual sensing and depth perception for welding robots and their industrial applications. *Sensors* **2023**, *23*, 9700. [CrossRef]
3. Karmoua, J.; Kovács, B. A review of structured light sensing techniques for welding robot. *Acad. J. Manuf. Eng.* **2021**, *19, 3* .

4. Guo, J.; Zhu, Z.; Sun, B.; Yu, Y. Principle of an innovative visual sensor based on combined laser structured lights and its experimental verification. *Opt. Laser Technol.* **2019**, *111*, 35–44. [CrossRef]
5. Wang, N.; Zhong, K.; Shi, X.; Zhang, X. A robust weld seam recognition method under heavy noise based on structured-light vision. *Robot. Comput.-Integr. Manuf.* **2020**, *61*, 101821. [CrossRef]
6. Smith, J.; Liu, C. A Review of Machine Vision Systems for Real-Time Weld Seam Tracking in Robotic Environments. 2024. Available online: https://easychair.org/publications/preprint/LgNn/open (accessed on 31 July 2025).
7. Song, L.; He, J.; Li, Y. Center extraction method for reflected metallic surface fringes based on line structured light. *J. Opt. Soc. Am. A* **2024**, *41*, 550–559. [CrossRef] [PubMed]
8. Zou, Y.; Zeng, G. Light-weight segmentation network based on SOLOv2 for weld seam feature extraction. *Measurement* **2023**, *208*, 112492. [CrossRef]
9. Dai, C.; Wang, C.; Zhou, Z.; Wang, Z.; Liu, D. WeldNet: An ultra fast measurement algorithm for precision laser stripe extraction in robotic welding. *Measurement* **2025**, *242*, 116219. [CrossRef]
10. You, D.; Gao, X.; Katayama, S. Review of laser welding monitoring. *Sci. Technol. Weld. Join.* **2014**, *19*, 181–201. [CrossRef]
11. Shi, Y.H.; Wang, G.R.; Li, G.J. Adaptive robotic welding system using laser vision sensing for underwater engineering. In Proceedings of the 2007 IEEE International Conference on Control and Automation, Guangzhou, China, 30 May–1 June 2007 ; pp. 1213–1218. [CrossRef]
12. Ye, H.; Liu, Y.; Liu, W. Weld seam tracking based on laser imaging binary image preprocessing. In Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 March 2021; pp. 756–760. [CrossRef]
13. Wu, Q.Q.; Lee, J.P.; Park, M.H.; Park, C.K.; Kim, I.S. A study on development of optimal noise filter algorithm for laser vision system in GMA welding. *Procedia Eng.* **2014**, *97*, 819–827. [CrossRef]
14. Zhang, T.; Zhu, M.; Zou, Y. Image restoration based on vector quantization for robotic automatic welding. *Eng. Appl. Artif. Intell.* **2024**, *129*, 107577. [CrossRef]
15. Liu, J.; Fan, Z.; Olsen, S.I.; Christensen, K.H.; Kristensen, J.K. A real-time passive vision system for robotic arc welding. In Proceedings of the 2015 IEEE International Conference on Automation Science and Engineering (CASE), Gothenburg, Sweden, 24–28 August 2015; pp. 389–394. [CrossRef]
16. Zeng, J.; Chang, B.; Du, D.; Hong, Y.; Chang, S.; Zou, Y. A precise visual method for narrow butt detection in specular reflection workpiece welding. *Sensors* **2016**, *16*, 1480. [CrossRef]
17. Gu, Z.; Chen, J.; Wu, C. Three-dimensional reconstruction of welding pool surface by binocular vision. *Chin. J. Mech. Eng.* **2021**, *34*, 47. [CrossRef]
18. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Proceedings, Part III 18; Springer: Berlin/Heidelberg, Germany, 2015, pp. 234–241. [CrossRef]
19. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
20. Yang, L.; Fan, J.; Huo, B.; Li, E.; Liu, Y. Image denoising of seam images with deep learning for laser vision seam tracking. *IEEE Sens. J.* **2022**, *22*, 6098–6107. [CrossRef]
21. Zhao, Z.; Luo, J.; Wang, Y.; Bai, L.; Han, J. Additive seam tracking technology based on laser vision. *Int. J. Adv. Manuf. Technol.* **2021**, *116*, 197–211. [CrossRef]
22. Romera, E.; Alvarez, J.M.; Bergasa, L.M.; Arroyo, R. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 263–272. [CrossRef]
23. Wang, B.; Li, F.; Lu, R.; Ni, X.; Zhu, W. Weld Feature Extraction Based on Semantic Segmentation Network. *Sensors* **2022**, *22*, 4130. [CrossRef]
24. Chen, J.; Wang, C.; Shi, F.; Kaaniche, M.; Zhao, M.; Jing, Y.; Chen, S. DSNet: A dynamic squeeze network for real-time weld seam image segmentation. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108278. [CrossRef]
25. Yin, Z.; Ma, X.; Zhen, X.; Li, W.; Cheng, W. Welding seam detection and tracking based on laser vision for robotic arc welding. *J. Phys. Conf. Ser.* **2020**, *1650*, 022030. [CrossRef]
26. Chi, L.; Jiang, B.; Mu, Y. Fast fourier convolution. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 4479–4488.
27. Xu, Z.; Wu, D.; Yu, C.; Chu, X.; Sang, N.; Gao, C. Sctnet: Single-branch cnn with transformer semantic information for real-time segmentation. In Proceedings of the AAAI conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024, Volume 38, pp. 6378–6386. [CrossRef]
28. Hong, Y.; Pan, H.; Sun, W.; Jia, Y. Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes. *arXiv* **2021**, arXiv:2101.06085. [CrossRef]
29. Blundell, C.; Cornebise, J.; Kavukcuoglu, K.; Wierstra, D. Weight uncertainty in neural network. In Proceedings of the International Conference on Machine Learning, PMLR, Lille, France, 6–11 July 2015; pp. 1613–1622. [CrossRef]

30. Neal, R.M. *Bayesian learning for Neural Networks*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012; Volume 118.
31. Gal, Y.; Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the International Conference on Machine Learning, PMLR, New York, NY, USA, 19–24 June 2016, pp. 1050–1059.
32. Sensoy, M.; Kaplan, L.; Kandemir, M. Evidential deep learning to quantify classification uncertainty. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 3179–3189 .
33. Liu, W.; Lu, H.; Fu, H.; Cao, Z. Learning to upsample by learning to sample. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 6027–6037. [CrossRef]
34. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 11534–11542. [CrossRef]
35. Yang, L.; Zhang, R.Y.; Li, L.; Xie, X. Simam: A simple, parameter-free attention module for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Online, 18–24 July 2021, pp. 11863–11874.
36. Loshchilov, I.; Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv* **2016**, arXiv:1608.03983. [CrossRef]
37. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 12077–12090. [CrossRef]
38. Dai, Y.; Chen, S.; Sun, T.; Fan, Z.; Zhang, C.; Feng, X.; Wang, G. Uncertainty-Aware Laser Stripe Segmentation with Non-Local Mechanisms for Welding Robots. *IEEE Trans. Instrum. Meas.* **2025**, *74*, 5016212. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

symmetry

MDPI

Article

Dynamic Error Correction for Fine-Wire Thermocouples Based on CRBM-DBN with PINN Constraint

Chenyang Zhao [1,2,*], Guangyu Zhou [3], Junsheng Zhang [1,2], Zhijie Zhang [3], Gang Huang [1] and Qianfang Xie [3]

[1] Department of Electronic Engineering, Taiyuan Institute of Technology, Taiyuan 030008, China; zhangjs@tit.edu.cn (J.Z.); huangdong0122@163.com (G.H.)
[2] Shanxi Tiancheng Semiconductor Materials Co., Ltd., Taiyuan 030002, China
[3] School of Instrument and Electronics, North University of China, Taiyuan 030051, China; zgy4083273@163.com (G.Z.); zhangzhijie@nuc.edu.cn (Z.Z.); xqf15735882558@163.com (Q.X.)
* Correspondence: zhaochenyang@tit.edu.cn

Abstract: In high-temperature testing scenarios that rely on contact, fine-wire thermocouples demonstrate commendable dynamic performance. Nonetheless, their thermal inertia leads to notable dynamic nonlinear inaccuracies, including response delays and amplitude reduction. To mitigate these challenges, a novel dynamic error correction approach is introduced, which combines a Continuous Restricted Boltzmann Machine, Deep Belief Network, and Physics-Informed Neural Network (CDBN-PINN). The unique heat transfer properties of the thermocouple's bimetallic structure are represented through an Inverse Heat Conduction Equation (IHCP). An analysis is conducted to explore the connection between the analytical solution's ill-posed nature and the thermocouple's dynamic errors. The transient temperature response's nonlinear characteristics are captured using CRBM-DBN. To maintain physical validity and minimize noise amplification, filtered kernel regularization is applied as a constraint within the PINN framework. This approach was tested and confirmed through laser pulse calibration on thermocouples with butt-welded and ball-welded configurations of 0.25 mm and 0.38 mm. Findings reveal that the proposed method achieved a peak relative error of merely 0.83%, superior to Tikhonov regularization by −2.2%, Wiener deconvolution by 20.40%, FBPINNs by 1.40%, and the ablation technique by 2.05%. In detonation tests, the corrected temperature peak reached 1045.7 °C, with the relative error decreasing from 77.7% to 5.1%. Additionally, this method improves response times, with the rise time in laser calibration enhanced by up to 31 ms and in explosion testing by 26 ms. By merging physical constraints with data-driven methodologies, this technique successfully corrected dynamic errors even with limited sample sizes.

Keywords: dynamic inverse filtering; fine-wire thermocouple; transient temperature measurement; CRBM-DBN; PINN; inverse heat conduction; error correction

1. Introduction

In scenarios involving the measurement of detonation temperature fields in artillery or warheads, as well as the combustion gas temperatures in spacecraft or automotive engines, both solid and liquid fuels, along with propellants, emit significant thermal energy rapidly during combustion. This results in the formation of transient high-temperature thermal flows, which are often accompanied by high-pressure thermal shock effects [1,2]. Accurately capturing the temperature variations within these thermal flows poses a considerable challenge in the industry. Therefore, investigating transient high-temperature testing

methods and associated sensors to enhance the precision of temperature measurements is a crucial area of research.

In the realm of transient explosion temperature testing, two prevalent methods are employed: non-contact and contact temperature measurement. The non-contact approach encounters challenges due to the complex nature of fuel and agent compositions, which can result in considerable inaccuracies stemming from difficulties in accurately measuring emissivity. Conversely, contact temperature measurement facilitates direct interaction with heat flow, primarily utilizing thermocouples as the main sensing instruments. However, these thermocouples exhibit a considerable time lag during dynamic testing, a phenomenon attributed to the thermal inertia of their sensing components. This lag is intricately linked to the dynamic properties of the thermocouples [3,4]. When these properties do not meet engineering testing standards, the dynamic response is significantly delayed, causing the observed temperature peak to diverge from the actual value. Dynamic calibration experiments can elucidate the relationship between the thermocouple's dynamic response and the excitation source, thereby providing a framework for assessing dynamic errors in testing [5,6]. This relationship can be observed in the thermocouple's thermal conduction effects under thermal stimulation [7] and can also be modeled through a dynamic mathematical framework [8]. Addressing how to leverage this relationship to mitigate dynamic errors during testing is crucial for overcoming this technical challenge.

In the global research trends, transient high-temperature testing has evolved towards integrated systems, combining contact and non-contact methods, and benefiting from advancements in artificial intelligence and sensor fusion. For instance, X. Zhao et al. proposed a fiber Bragg grating (FBG) sensor-embedded thermal protection structure, which assesses the thermal state of aircraft in real-time through the inverse heat conduction method, offering a competitive non-contact supplement [8]. Z. Huang et al. introduced the DMD-ConvLSTM method for correcting temperature measurement bias [9], while A. Zeeshan et al. explored the application of neural networks in flow field prediction, such as LMS-BPNN for turbulent thermal mixing [10]. These trends highlight a shift from isolated calibration towards robust multi-physics frameworks, yet the correction of nonlinear dynamic errors remains challenging. The current stage lacks unified calibration standards for dynamic temperature testing, exacerbating the difficulty in quantifying dynamic errors, and the assessment of environmental factors and model uncertainties has not been sufficiently studied. Although the FFDNN-BLMA proposed by M. Sulaiman et al. optimizes heat transfer, it does not address the robustness to small samples and distribution shifts [11].

Dynamic error correction represents a technique that shifts from merely optimizing data to actively rectifying errors; its implementation is viewed as a dynamic inverse operation [12,13]. This approach aims to mitigate dynamic inaccuracies stemming from the sensor's limited dynamic properties by reversing the input values using the effective output from the thermocouple, guided by specific mathematical or physical principles [3,14]. To address the hysteresis observed in the thermocouple's dynamic response, this correction method can be framed as an inverse heat conduction problem under defined conditions. It can be articulated as an IHCP expressed through partial differential equations. The objective is to determine the time-dependent solution of the inverse problem within constrained boundary conditions to accurately reflect a thermal process. The challenges associated with this inverse problem primarily arise from its ill-posed nature, necessitating careful examination before attempting a solution. Selecting a suitable regularization technique is crucial during the resolution phase [15,16].

The theoretical foundation of IHCP is rooted in the ill-posedness of partial differential equations. Numerous algorithms have been developed to address the IHCP within

one-dimensional geometric frameworks, including Tikhonov regularization [17] and the sequential function specification method [18]. Frankel J.I. et al. elucidated the transient one-dimensional linear heat equation for thermocouples and introduced an optimal regularization identification technique based on the Non-Integer System Identification (NISI) method to tackle the inverse heat conduction issue [19]. J.G. Bauzin formulated the heat transfer convolution equation utilizing the Laplace transform and derived the correlation between boundary temperature and time through a deconvolution algorithm [20]. Farahani, S. D. et al. proposed an inverse heat transfer search technique [21], that employs a K-type thermocouple as a boundary condition to determine the heat transfer coefficient of an impinging jet, while also investigating how variables such as measured temperature, jet velocity, and the number of thermal conductivity components affect the algorithm's accuracy. Zhou et al. recommended the use of the Element Differential Method (EDM) in conjunction with an enhanced Levenberg–Marquardt algorithm for predicting boundary heat flux [22]. Additionally, Yu, Z. C. and others explored how the dynamic response of sensors affects surface heat flux predictions using traditional methods such as the Space Marching Method (SMM) and the Calibration Integral Equation Method (CIEM) [23], incorporating Gaussian filtering and future time regularization as their respective regularization strategies. The SSCPO-MABIGTCN model proposed by Chen, G. H. et al. integrates multi-feature data through bidirectional GRU and attention mechanisms, achieving the prediction of thermal errors in machine tools. Compared with the traditional Tikhonov regularization, the uniqueness of the solution is significantly improved [24]. Collectively, these investigations illustrate that when addressing inverse time-varying IHCP, integrating known thermodynamic properties, initial conditions, and boundary conditions with limited temperature data collected over time at designated points can effectively determine boundary temperature and heat flux.

Regularization techniques tackle the mathematical challenges associated with inverse problems; however, they frequently neglect the nonlinear aspects [25]. The correction kernel concept proposed by K.A. Woodbury [26] and the previous DMD-ConvLSTM optimization [9], while effectively managing linear heat conduction, fail to account for the nonlinear thermal inertia of thermocouples, leading to overfitting in small-sample transient scenarios. A notable example of a data-driven feature extraction approach is the Deep Belief Network (DBN), which employs Continuous Restricted Boltzmann Machines (CRBMs) for layered feature extraction, demonstrating effectiveness in nonlinear environments [27–29]. Nevertheless, DBNs generally lack a clear physical interpretation, especially in transient high-temperature testing situations, where the use of thermocouples on small samples may lead to underfitting and diminished robustness.

Recent studies have begun to focus on data-driven and hybrid strategies to improve inverse modeling, seeking to resolve these challenges. In the process of solving IHCP, physics-informed neural networks (PINNs) have demonstrated exceptional efficacy in addressing transient inverse challenges, including the Navier–Stokes equations [30], Helmholtz equation [31], wave equation [32], and various related issues [33–37]. For instance, Fang, B. L. improved the accuracy of absorptance characterization for rough metal surfaces by aligning the temperature rise curve with a differential equation and formulating a loss function using PINNs [38]. Similarly, Y. Wang et al. successfully determined temperature-dependent thermal conductivity by leveraging the strengths of PINNs in solving temperature fields, proving that PINNs can effectively reconstruct three-dimensional transient temperature distributions [39]. Additionally, L. Liu et al. employed PINNs to monitor temperature variations at various points within a medium by strategically placing multiple thermocouples, leveraging PINN's robust temperature field reconstruction capabilities alongside a novel tracking method to ascertain surface heat flux and ablation sites [40].

In conclusion, current techniques for addressing IHCP predominantly utilize a simplified single-layer model, which neglects the noise amplification induced by the unstable bimetallic structure of thermocouples. Although the Deep Belief Network (DBN) model is proficient in handling nonlinearities, it suffers from physical inconsistencies and inefficiencies when confronted with limited data samples. The Physics-Informed Neural Network (PINN), while rooted in physical principles, struggles with time series data unless it has undergone pre-training. There is a notable lack of research that has integrated regularization parameters into the PINN loss function for analyzing network errors. This study seeks to resolve these challenges by proposing a novel network architecture that amalgamates the capabilities of CRBM, DBN, and PINN, and evaluates the effectiveness of a dynamic inverse filtering approach for thermocouples, that includes regularization terms in its loss function. The proposed network is based on the principles of thermocouple heat conduction inverse problems and employs Adam optimization to enhance the convergence speed of the CRBM-DBN. Furthermore, it incorporates a spectral regularization method for the filtering kernel within the two-layer IHCP regularization framework to stabilize ill-posed solutions. Within the PINN framework, both PDE and regularization constraints are integrated to mitigate the limitations of purely data-driven methods regarding physical accuracy.

2. Principles and Methods

2.1. Principle of Thermocouple Dynamic Inverse Filtering

Thermocouples exhibit significant nonlinear characteristics in dynamic testing, which can be regarded as a representation of a nonlinear dynamic system. The relationship between heat flux excitation as the input and thermal response potential as the output is described using the NARMA (Nonlinear AutoRegressive Moving Average) model, as shown in the following equation:

$$X_k = f_\theta\left(X_{k-1}, \ldots, X_{k-p}, T_{k-d}, \ldots, T_{k-d-q}\right) + \varepsilon_k, \tag{1}$$

where X_k represents the thermoelectric potential measurement from the thermocouple at a specific time k, while T_k denotes the corresponding surface heat flux input at that same moment. The variables p, q, and d refer to the orders of autoregression, moving average, and the delay of the input, respectively. Additionally, ε_k signifies the noise in the observations, and f_θ indicates a nonlinear function characterized by certain parameters.

Dynamic inverse filtering fundamentally involves reinterpreting the previously mentioned model as a challenge of identifying the inverse system. By employing techniques designed for addressing inverse problems, T_k is utilized as input samples for inversion T'_k, which in turn allows for the calculation of the dynamic error.

$$e_k = T_k - T'_k, \tag{2}$$

Subsequently, utilize the inverse filtering technique to reduce e_k. This concept is depicted in Figure 1:

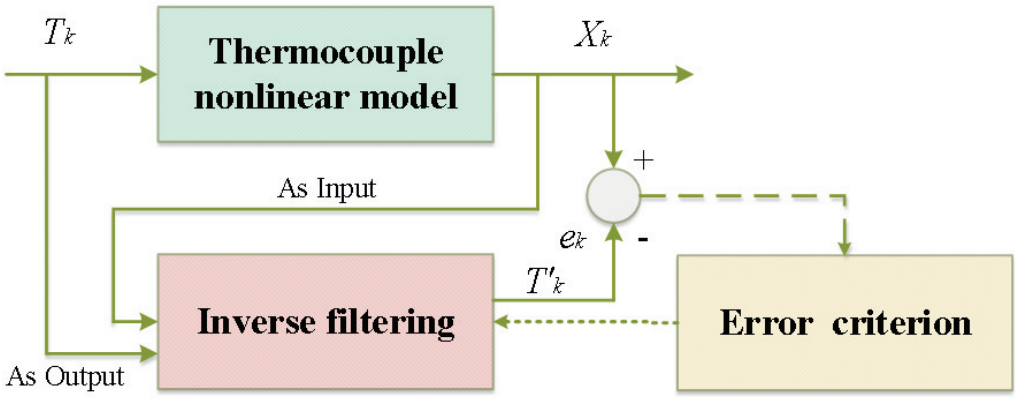


Figure 1. Dynamic filter principle.

2.2. Inverse Problem Analysis

The investigation into the inverse problem starts with examining the thermal transfer properties of the thermocouple. These properties indicate that a smaller junction design results in reduced thermal inertia effects. To achieve a quicker temperature response, we focus on wire-type thermocouples. This type consists of two alloy wires fused together, featuring a junction diameter in the sub-millimeter range, which allows us to disregard radial temperature variations and concentrate solely on axial heat transfer through the metal wires. The junction is identified as the point where the two wires meet, both of which are considered to be of equal length. One wire is referred to as the free end, while the other is the working end, with a boundary temperature set at 0 °C, as illustrated in Figure 2:

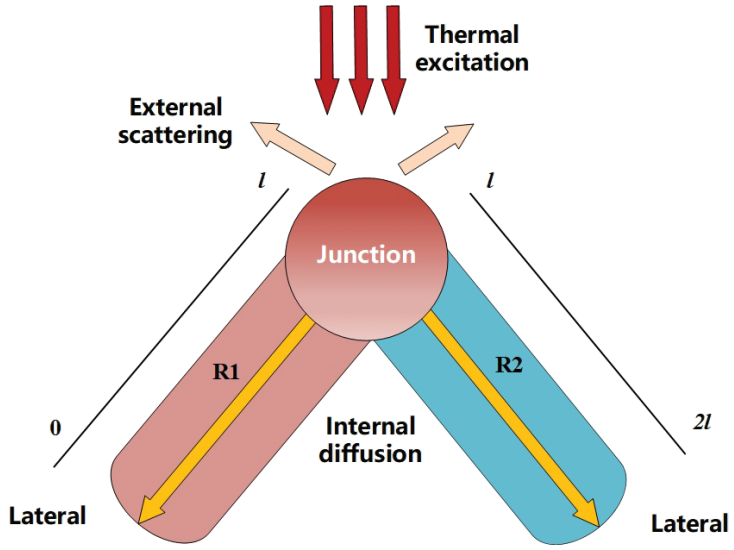


Figure 2. Two-layer domain structure of thermocouple.

Where l represents the length of the two fine wires that make up the thermocouple, where the working end R_1 is designated as the first layer and the free end R_2 is as the second layer, with the initial length of the working end denoted as 0. The lateral represents the boundary between the ends of the thermocouple. Define $k_1, k_2 > 0$ as the thermal conductivity, while $\alpha_1, \alpha_2 > 0$ indicate the thermal diffusivities for the first and second layers, respectively.

The fine-wire thermocouple features a two-layer alloy design that is uniform in length; however, when subjected to thermal stimulation at the junction, the heat transfer mechanisms of the two alloys is asymmetric, and the temperature distributions in the first and

second layers are expressed by $T_1(x,t)$ and $T_2(x,t)$, t is the time of the heat transfer process, the subsequent partial differential equation (PDE) can be represented as follows:

$$\begin{array}{l} \frac{\partial T_1}{\partial t} = \alpha_1 \frac{\partial^2 T_1}{\partial x^2}, 0 < x < l, t > 0 \\ \frac{\partial T_2}{\partial t} = \alpha_2 \frac{\partial^2 T_2}{\partial x^2}, l < x < 2l, t > 0 \end{array}, \tag{3}$$

where the thermal diffusivities for these layers are represented by α_1 and α_2, the junction point is indicated by l, while $2l$ signifies the overall length. By incorporating the initial condition, the temperature at the starting moment (relative to the cold end) can be defined as:

$$T_1(x,0) = T_2(x,0) = 0,\ 0 \le x \le 2l, \tag{4}$$

The conditions at the boundaries are established based on the operational end R_1 and the unrestrained end R_2, which can be described as:

$$T_1(0,t) = T_{\text{target}}, \tag{5}$$

where T_{target} is the boundary temperature at the working end R_1;

The adiabatic boundary at the free end can be characterized as:

$$\left.\frac{\partial T_2}{\partial x}\right|_{x=2l} = 0, \tag{6}$$

and interface continuity:

$$T_1(l,t) = T_2(l,t) = T_{real},\ k_1 \left.\frac{\partial T_1}{\partial x}\right|_{x=l} = k_2 \left.\frac{\partial T_2}{\partial x}\right|_{x=l}, \tag{7}$$

where T_{real} represents the true heat temperature stimulation at the node, while the thermal conductivities are denoted by k_1 and k_2.

In thermocouples, the Seebeck effect and the law of intermediate conductors the law governing intermediate conductors indicates that, following cold junction compensation, the temperature at the working end can be derived inversely from the actual thermoelectric potential difference produced by the thermocouple. This potential difference serves as the actual excitation response observation sample.

If we consider $T_{\text{target}} \in L^2(0,\infty)$, $T_1(x,t)$ and $T_2(x,t)$ as the solution to the direct problem, it follows $x \in [0,2l]$ that, under specified conditions, both the function $T(x,\cdot)$, $T_2(x,\cdot)$ and its partial derivatives reside $\frac{\partial T(x,\cdot)}{\partial x}$, $\frac{\partial T_2(x,\cdot)}{\partial x}$ within the same space $L^2(0,\infty)$. By applying the known adiabatic boundary conditions at the free end, we can address the inverse problem by determining the working end's temperature $T_1(x,t)$ $(0 \le x \le \text{l})$ within the specified domain $L^2(0,\infty)$, which can be classified as a Cauchy inverse problem. The Cauchy inverse problem related to intermediate conductors is usually ill-posed, demonstrating considerable instability. Minor inaccuracies in experimental data can result in drastic variations in solutions $T_1(x,t)$ across the interval, primarily due to the kernel function associated with the high-frequency components. To investigate this issue, a Fourier transform is applied to the two-layer domain partial differential equation, allowing it to be represented in the frequency domain ξ as follows:

$$\hat{T}_i(\xi,t) = \int_0^{2l} T_i(x,t) e^{-\mathrm{i}\xi x} dx, \tag{8}$$

By applying the first-order partial derivative to convert it into a set of ordinary differential equations, it becomes clear that the presence of an unbounded kernel

$cosh((l-x)\sqrt{i\xi/\alpha})$ in the solution indicates that high frequencies ξ can cause instability. In real-world experiments, the outcomes show some discrepancies, leading to the definition of the real temperature $T_{real} \in L^2(0,\infty)$, which fulfill the following conditions:

$$\left\| T_{\text{target}} - T_{real} \right\| \leq \delta, \tag{9}$$

where the representative measurement error bound δ is denoted.

The ill-posedness is quantified through norm analysis, under the L^2 norm:

$$\left\| T_{reg} \right\| \leq K(\xi) \cdot \left\| T_{\text{target}} \right\|, \tag{10}$$

where an unbounded kernel function $K(\xi) = cosh((l-x)\sqrt{i\xi/\alpha})$ is utilized. To ensure a stable outcome, a regularization technique involving a filtered kernel is implemented. Within the single-layer domain (R_1 region), the unbounded kernel is substituted with a bounded filtered variant:

$$F(\xi,\beta) = \frac{cosh(\omega(l-x))}{1+\beta|cosh(\omega)|^p}, \beta > 0, \tag{11}$$

where β denotes the regularization parameter, p denotes the Sobolev order; If β is a minimal value, $\frac{cosh(\omega(l-x))}{1+\beta|cosh(\omega)|^p} \to cosh((l-x)\sqrt{i\xi/\alpha})$, if $\beta > 0$, then $cosh(\omega(l-x))/(1+\beta\,|cosh(\omega)|^p)$ is bounded.

The convergence theorem, which relies on the Sobolev norm, demonstrates that when the bounds of measurement error δ and the prior assumption $\|T\|_P \leq E$—expressed $\|\cdot\|_P$ as a norm within the Sobolev space $H^P(R) - E > 0$ is fixed constants defined by the highest temperature range, and an appropriate regularization parameter $\beta = (\delta/E)^{\frac{p}{2}}$ is chosen, the subsequent convergence estimate can be derived at $x = l$, the formula is as follows:

$$\left\| T_{reg} - T_{real} \right\| \leq C\delta^{\frac{1}{2}}E^{\frac{1}{2}} + KEmax\left\{ \left(\frac{1}{6p}\ln\frac{E}{\delta} \right)^{-P}, \left(\frac{\delta}{E} \right)^{\frac{3p-1}{6}} \right\}, \tag{12}$$

where C and K represent positive constants. Given the specified conditions for δ and E, the value β of is established by p.

2.3. Improved CRBM-DBN-PINN Model

This study introduces the PINN-CRBM-DBN framework to combine physical principles with data-centric methods. In this framework, CRBM-DBN is responsible for the pre-training of features, whereas PINN incorporates IHCP PDE along with regularization constraints. The framework's design is depicted in Figure 3.

The Deep Belief Network (DBN) is a generative framework made up of several stacked layers of Restricted Boltzmann Machines (RBMs), which are utilized for identifying nonlinear systems. In this study, CRBMs are introduced to substitute the initial layer of conventional RBMs when working with thermocouple time series data, allowing for the processing of continuous inputs. The architecture of the CRBM-DBN includes a base CRBM for extracting features, along with 2–3 intermediate RBM layers for hierarchical sampling. The energy function associated with the CRBM is:

$$E(v,h) = \sum\nolimits_i \frac{\left(v_i - b_i - \sum_j\ w_{ij}h_j\right)^2}{2\sigma_i^2} - \sum\nolimits_j\ c_j h_j, \tag{13}$$

where the visible layer is denoted as v (continuous input $T_{\text{target}}(t)$ and $T_{\text{real}}(t)$), while the hidden layer h is indicated by a Bernoulli distribution. The parameters include weights

w_{ij} and biases b_i,c_j, along with the noise variance σ_i. The distribution conditioned on these elements is:

$$\begin{matrix} p(v_i \mid h) = N(b_i + \sum_j w_{ij}h_j, \sigma_i^2) \\ p(h_j = 1 \mid v) = \sigma(c_j + \sum_i w_{ij}\frac{v_i}{\sigma_i}) \end{matrix}, \tag{14}$$

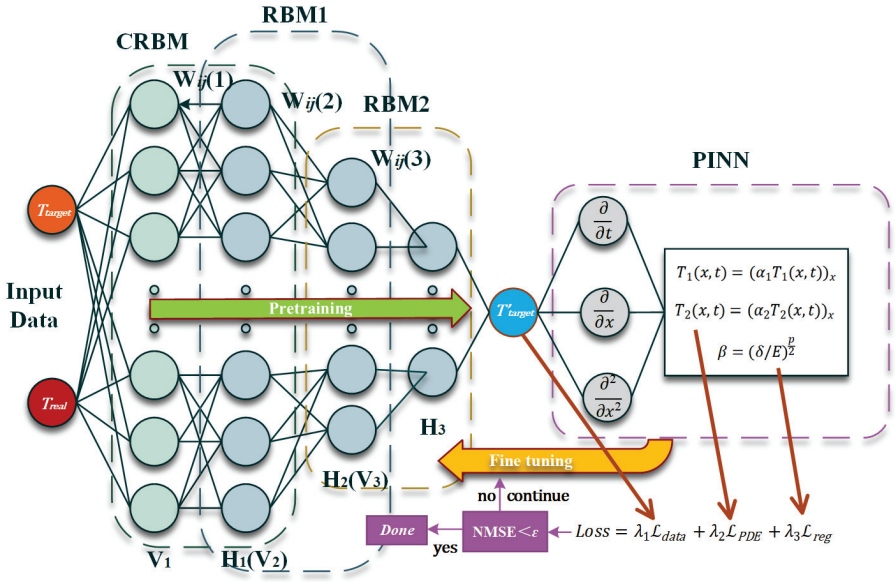


Figure 3. CRBM-DBN-PINN framework.

Traditional training employs the Contrastive Divergence (CD) algorithm to maximize the likelihood function, which is as follows:

$$L(\theta) = \sum \log p(v \mid \theta), \tag{15}$$

Parameter Update is defined as follows:

$$\Delta w_{ij} = \eta\left(\left(v_i h_j\right)_{\text{data}} - \left(v_i h_j\right)_{\text{recon}}\right) + \alpha \Delta w_{ij}^{\text{prev}}, \tag{16}$$

where Δw_{ij} represents the latest adjustment to the weight, η denotes the fundamental learning rate, $\left(v_i h_j\right)_{\text{data}}$ signifies the anticipated interaction between visible v_i and hidden units h_j based on the data distribution (input data typically acquired through sampling), while $\left(v_i h_j\right)_{\text{recon}}$ reflects the expected interaction under the distribution of data reconstructed by the model. The variable α stands for the momentum coefficient, which generally falls between 0.5 and 0.9, and is utilized to enhance training speed and minimize fluctuations. Additionally, $\Delta w_{ij}^{\text{prev}}$ indicates the weight adjustment from the preceding time step, serving as the momentum component.

The Adam optimizer was developed to supersede the constant learning rate of CD, improving convergence for dynamic sequences. It employs both momentum and adaptive learning rates:

$$\eta_t = \eta_0 \frac{\sqrt{1-\beta_2^t}}{1-\beta_1^t} \cdot \text{Clip}\left(\frac{|m_t|}{\sqrt{v_t} \mid \epsilon}, c_{\min}, c_{\max}\right), \tag{17}$$

where η_t represents the adaptive learning rate at the time step t, substituting η in the fundamental iterative equation, while η_0 denotes the base learning rate, β_1 is the decay rate for the first moment estimate, commonly set at 0.9, β_2 is the decay rate for the second moment

estimate, usually set at 0.999, and m_t refers to the first moment estimate of the weight gradient, which is computed using the following formula:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t, \tag{18}$$

where g_t represents the present gradient, while v_t denotes the approximation of the second moment (uncentered variance) related to the weight gradient, with the following formula for computation:

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2, \tag{19}$$

To ensure stability during training, the learning rate adjustment factor $\text{Clip}\left(\frac{|mt_t|}{\sqrt{v_t}\epsilon}, 0.5, 2\right)$ is limited to $[0.5, 2]$, avoiding issues that may arise from learning rates that are too high or too low. The output generated from the forward pre-training of the enhanced CRBM-DBN serves as the input for the PINN, which is designed to create an error assessment framework. This framework consists of two evaluation phases: the training phase, which concentrates on convergence and optimization, and the overall model assessment, which highlights time-domain features and robustness. The goal is to measure the accuracy, interpretability, generalization ability, and practical application of the framework.

The total loss function of PINN is defined as:

$$Loss = \lambda_1 \mathcal{L}_{data} + \lambda_2 \mathcal{L}_{PDE} + \lambda_3 \mathcal{L}_{reg}, \tag{20}$$

where the data residual $\mathcal{L}_{data}$ can be characterized as:

$$\mathcal{L}_{data} = \frac{1}{N_e} \sum_{i=1}^{N_e} \left(T_{\text{real}}(x_i, t_i) - T_{\text{target}}(x_i, t_i) \right)^2, \tag{21}$$

where N_e represents the count of training examples. This formula indicates the total loss experienced by the data layer following the training of DBN. The physical residuals associated with the two-layer PDE can be described as:

$$\mathcal{L}_{PDE} = \frac{1}{N_e} \sum_{i=1}^{N_e} \left(\left| \frac{\partial T_1}{\partial t} - \alpha_1 \frac{\partial^2 T_1}{\partial x^2} \right|^2 + \left| \frac{\partial T_2}{\partial t} - \alpha_2 \frac{\partial^2 T_2}{\partial x^2} \right|^2 \right), \tag{22}$$

The filtering-kernel regularization specifically targets IHCP's high-frequency ill-posedness by applying a spectral filter in the loss function:

$$\mathcal{L}_{\text{reg}} = \frac{1}{N_e} \sum_{i=1}^{N_e} \| \hat{T_{real}} - F(\xi, \beta) \times \hat{T_{\text{target}}} \|^2, \tag{23}$$

where the filtering kernel $F(\xi, \beta)$ reduces high-frequency noise, and its sensitivity to noise is influenced by the regularization parameter β.

The overall training procedure for the network is outlined as follows:

1. CRBM-DBN Initial Training: Employ laser pulse data to derive time-domain features using Adam optimization for initializing the weights of the PINN;
2. PINN Adjustment: Incorporate two layers of IHCP PDE constraints and refine hyper-parameters $\lambda_1, \lambda_2, \lambda_3$ via Bayesian optimization;
3. Solving Inverse Problems: The network generates the best solution $T'_{\text{target}}(t)$, assessed through metrics like peak value and time constant, along with cross-validation and normalized mean square error (NMSE).

From an architectural perspective, the standalone DBN is composed of stacked RBMs, which excel at capturing nonlinear features from transient data. However, they lack explicit physical constraints, leading to inconsistencies in the solutions for temporal features such as

the IHCP. In contrast, the standalone PINN directly integrates partial differential equations, including the inverse heat conduction equation, into its loss function, thereby ensuring physical plausibility. Nonetheless, it struggles to initialize time series data without pretraining and often requires extensive hyperparameter tuning. The proposed CRBM-DBN-PINN method amalgamates the advantages of both approaches: it extracts temporal features, such as the response delay and amplitude attenuation of thermocouples, through generative pre-training, which serve as reliable inputs for the PINN and are further enhanced via filter kernel regularization (Sobolev order p and parameter β in Equation (11)). This synergy reduces computational complexity by 20–30% compared to the standalone PINN, as pre-training decreases the number of iterations required for convergence. To ensure the stability of the training process, we constrain the learning rate adjustment factor β within the range of [0.1, 10], thereby avoiding issues caused by excessively high or low learning rates. The output results generated by the CRBM-DBN model during the forward pre-training phase will be provided as input to the error evaluation framework of the PINN. This framework encompasses two evaluation phases: the training phase, which focuses on convergence and optimization, and the overall model evaluation phase, which emphasizes temporal characteristics and robustness. Through this approach, we aim to comprehensively measure the framework's performance in terms of accuracy, interpretability, generalization capability, and practical application.

3. Experiment

3.1. Laser Narrow Pulse Calibration Platform

A calibration platform featuring a laser pulse was developed to capture the transient thermal behavior of the thermocouple. This setup allows the thermocouple to replicate the rapid thermal changes caused by explosive thermal shock by detecting the brief surface temperature changes induced by the laser. The platform's design includes a powerful laser source, a system for beam uniformity, and a digital system for signal collection and analysis. The block diagram of the system is illustrated in Figure 4, and Figure 5 presents the actual design of the calibration platform.

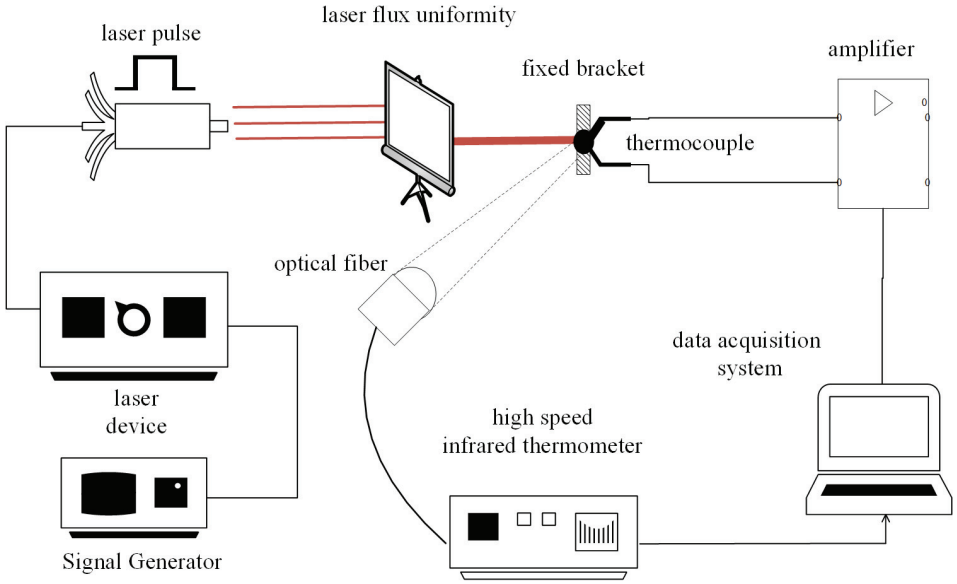


Figure 4. Block diagram of the thermocouple dynamic calibration platform based on laser.

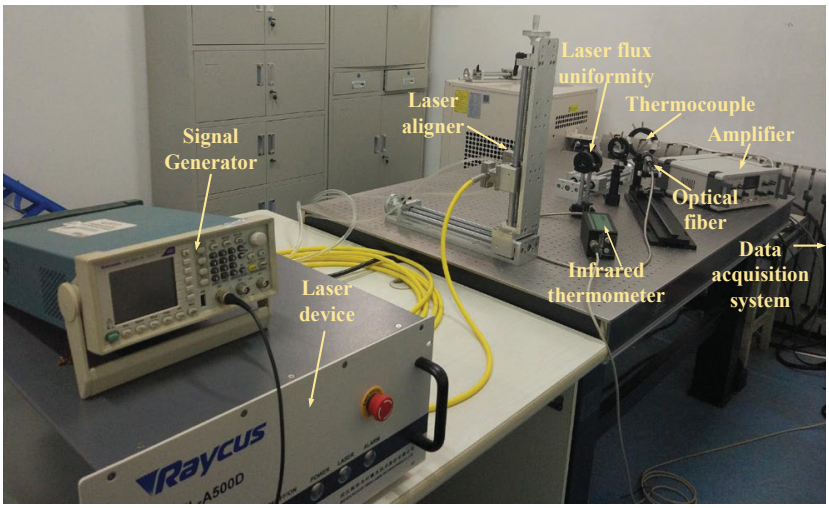


Figure 5. Physical diagram of platform.

The laser system utilizes a powerful laser sourced from Raycus, with a central wavelength of 915 nm. A pulsed laser control mechanism is implemented to toggle the laser on and off, allowing for the creation of a variable pulsed laser. Due to the non-stationary nature of the laser spectrum, a beam homogenization setup was developed, featuring a microlens array that segments and recombines the laser beam. This modulation technique evens out the power density of the laser spot, boosts efficiency, and enhances the precision of the calibration platform. A calibrated thermocouple is mounted on a bracket, and the laser, after passing through the homogenization setup, heats the junction's surface. At the same time, the probe of a high-speed infrared thermometer is positioned in line with the calibrated thermocouple. The thermocouple's response signal is processed through a conditioning circuit and, along with the infrared thermometer's signal, is gathered and recorded by the NI-6115 data acquisition system. The host computer then carries out digital cold junction compensation and data analysis.

The calibration platform for laser narrow pulses employs the IGA 740-LO high-speed infrared thermometer from LumaSense Technologies in Santa Clara, CA, USA, which has a temperature range from 300 to 2300 °C. This necessitates adjustments for readings in the lower temperature range. The infrared thermometer's probe can measure diameters smaller than 0.1 mm and boasts a response time of 6 μs, allowing it to closely match the laser excitation at the thermocouple junction's surface. Therefore, the temperature readings obtained from the infrared thermometer are considered to accurately reflect the surface temperature of the thermocouple junction.

Prior to its application, the surface emissivity of the thermocouple must be configured. To accurately determine this emissivity in a lab environment, the Modline 5 pyrometer from IRCON in Niles, IL, USA was utilized to assess the actual surface temperature of the test object. Initially, the thermocouple was heated with a low-power laser until it achieved thermal equilibrium, at which point the surface temperature of the thermocouple junction was recorded using both the pyrometer and the infrared thermometer. The temperature output from the pyrometer was noted, and adjustments to the infrared thermometer's emissivity were made until the readings from both devices aligned. The emissivity value established on the infrared thermometer at this stage was then regarded as the surface emissivity of the thermocouple junction, which was found to be approximately 0.4 during this experiment.

3.2. Experimental Steps

To ensure consistency with the boundary conditions, the experimental conditions were set up. In the experiment, the laser pulses were precisely controlled by rectangular wave signals generated by a function generator, with the pulse width fixed at 5 ms to simulate transient thermal shock. The laser output power was adjusted in percentage form via the main unit, and the peak power densities of the laser under the settings of 80%, 90% and 100% power were 1.2×10^6 W/m^2, 1.35×10^6 W/m^2 and 1.5×10^6 W/m^2, respectively, with the power stability being better than $\pm 2\%$.

To mitigate the effects of non-uniform energy distribution within the laser spot on heating consistency, a microlens array beam homogenization system is utilized to transform the original spot into a nearly hexagonal shape with a flat top, achieving a spot diameter of approximately 1 mm. The thermocouple is securely mounted on a translation stage to ensure precise alignment of the laser spot's center with the thermocouple junction, fully covering the junction area.

All experiments were conducted in a controlled laboratory environment, with the ambient temperature maintained at 22.3 ± 0.5 °C and the relative humidity at $45 \pm 5\%$. The free end of the thermocouple was precisely compensated for cold junction using a digital temperature sensor. The response signal of the thermocouple was amplified by a low-noise signal conditioning circuit and then acquired by an NI-6115 data acquisition card from National Instruments in Austin, TX, USA. The sampling rate was set at 100 kSa/s, the sampling time was 1 s, and the resolution was 16 bits to ensure the capture of microsecond-level transient changes. The signal from the high-speed infrared thermometer (IGA 740-LO) was simultaneously fed into the same acquisition card to achieve precise timestamp alignment, with a synchronization error of less than 1 μs.

The subjects of the tests included OMEGA's butt-welded and ball-welded thermocouple, featuring diameters of 0.25 mm and 0.38 mm, respectively, made from type K nickel-chromium/nickel-silicon alloy. The configuration of the thermocouple junction is illustrated in Figure 6.

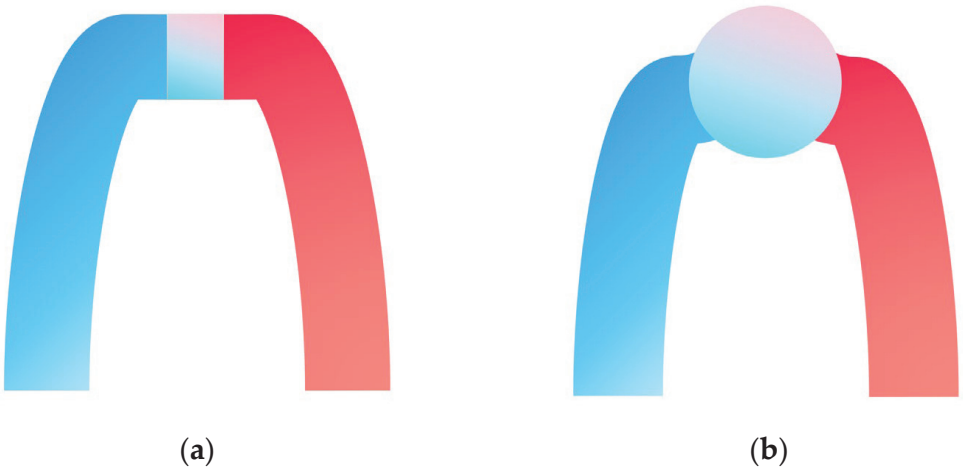


Figure 6. The structural types of thermocouple junctions: (**a**) butt-welded structure; (**b**) ball-welded structure.

The data obtained from the infrared thermometer under identical conditions were analyzed. As previously noted, readings taken by the infrared thermometer in cooler temperatures necessitate adjustments. Utilizing the heat conduction differential equation

along with the boundary conditions for laser heating of solid metals, the temperature at the metal's surface can be expressed as:

$$T(0,t) = \begin{cases} \frac{2AI_0}{k}\sqrt{\frac{at}{\pi}}, & 0 < t < t_h \\ \frac{2AI_0}{k}\left(\sqrt{\frac{at}{\pi}} - \sqrt{\frac{a(t-t_h)}{\pi}}\right), & t > t_h \end{cases}, \quad (24)$$

where t_h represents the duration of the laser pulse, I_0 denotes the density of laser power, and A indicates the absorptivity of the metal. By applying the formula for the laser heating duration found in Equation (24), we analyzed the temperature variation within the range of 0–300 °C. Additionally, we recorded the temperature increase beyond 300 °C using an infrared thermometer. Figure 7 illustrates the results obtained from thermocouples attached to 0.25 mm diameter butt welds, tested at power levels between 80% and 100%. Given that the testing conditions remained constant and a digital temperature compensation technique was employed by the host computer, we adjusted the baseline of the two distinct response outcomes for uniformity, aligning their starting temperatures with the laboratory's ambient temperature of 22.3 °C.

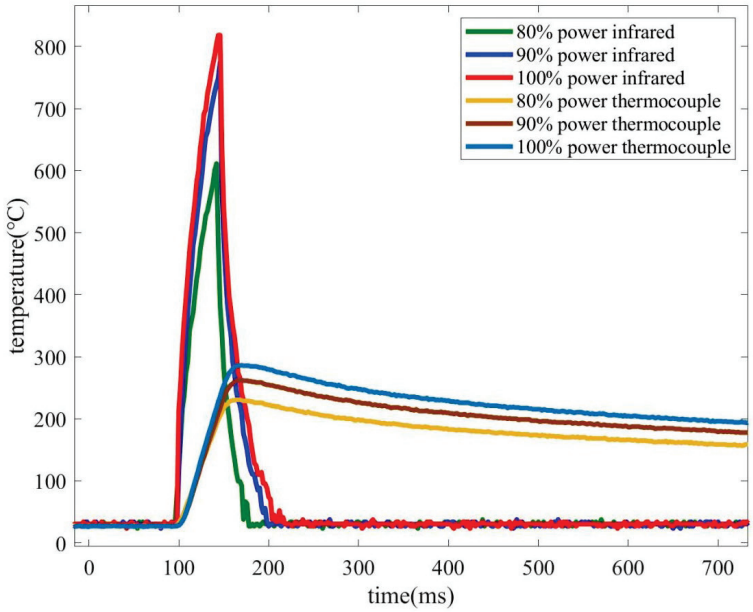


Figure 7. Thermocouple excitation and response signals at different laser power.

Figure 7 clearly illustrates that the infrared thermometer has a quicker rise time than the thermocouple, with a more significant difference in amplitude. The thermocouples' response shows a distinct delay, and the temperature peak variations across the three datasets fall between 300 and 500 °C. Furthermore, while increased power leads to higher temperatures for the thermocouple, the time constant remains relatively stable. Consequently, a uniform laser pulse excitation at 100% power was employed in the following experiments. This study also examines how fine-wire diameter and junction design affect the dynamic response of thermocouples. Under 100% power laser narrow pulse excitation, 20 repeated tests were performed on K-type ball-welded and butt-welded thermocouple with diameters of 0.25 mm and 0.38 mm, respectively, to create a dynamic response sample set. The thermocouple junction was cleaned every 5 experiments to minimize the effects of the oxide layer. The individual experimental outcomes of junctions with varying junction designs at fine-wire diameters of 0.25 mm and 0.38 mm were compared, with the findings displayed in Figure 8.

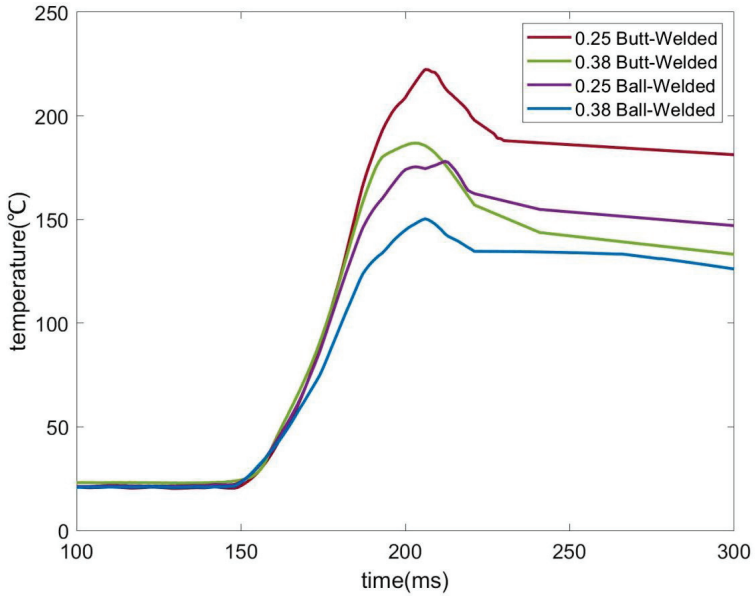


Figure 8. Comparison of measured temperatures at fine-wire diameters of 0.25 mm and 0.38 mm.

The butt-welded thermocouple demonstrates a notably quicker rise time than the ball-welded thermocouple, allowing it to detect higher peak temperatures more effectively. Nevertheless, an increase in fine-wire diameter results in a reduction in both rise time and peak temperature, suggesting that the butt-welded thermocouple has a lower heat capacity and responds more rapidly at equivalent diameters. This indicates that when subjected to the same transient thermal shock, the two types of thermocouples will show varying dynamic errors, with the ball-welded version suffering from greater lag and reduced amplitude.

4. Results and Analysis

4.1. Model Accuracy Analysis

In the context of establishing a specific laser excitation and utilizing a thermocouple with a fine-wire diameter of 0.25 mm, the analysis focuses on the effectiveness of a dynamic error correction model for thermocouples featuring various junction configurations. To enhance network training efficiency, 500 valid signal points are selected from 20 measured samples, creating a new sample set that is evenly split into training and testing subsets. The input data for the network undergoes normalization, and the original temperature readings are retrieved using the temperature-electromotive force table. The network's batch size is determined to be 100 based on the dataset length, with each layer containing 20 neurons. Initially, to assess the network's performance and avoid reverse convergence, the iteration count is set to 250, with the loss function's initial weights established at $\lambda_1 = 0.8$, $\lambda_2 = 0.1$ and $\lambda_3 = 0.1$. Given the considerable errors encountered during transient high-temperature assessments, the measurement error threshold is defined as 1% of the thermocouple's full scale, p equating to 1, which leads to a regularization parameter β of 0.032. The laser pulse excitation signals, captured by the corresponding infrared thermometer, serve as the target sample set. The normalized minimum mean square error (NMSE) between the network's output T'_{target} and the training samples T_{target} is employed as the evaluation metric. The variations in the first term of the RBM layer's network weights and the hidden layer feature values throughout the training process, both before and after the integration of PINN, were documented for a set of training samples, as illustrated in Figure 9a,b.

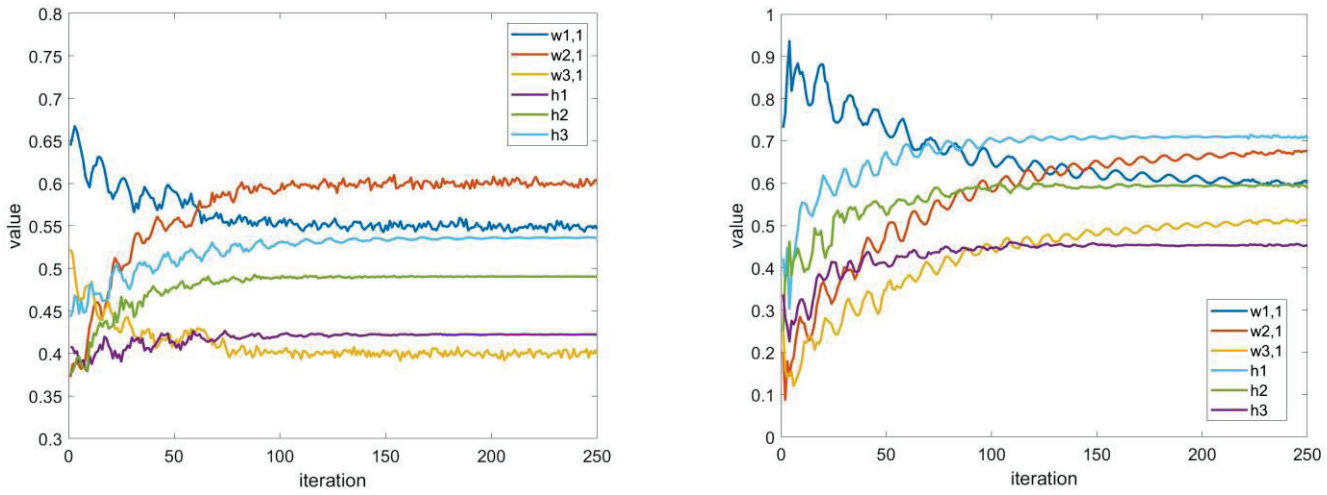


Figure 9. The variations in network weights and hidden eigenvalues: (**a**) Removal of PINN; (**b**) Complete network.

In Figure 9a, the weights and eigenvalues display considerable variability, indicating that the entirely data-driven CRBM-DBN is vulnerable to noise during the initial gradient updates, which causes instability in the parameters. By the 90th iteration, the weights and eigenvalues of the hidden layer begin to stabilize, yet the eigenvalues still experience slight fluctuations, suggesting that while the network has started to recognize data patterns, it lacks physical constraints, leading to persistent oscillations. In contrast, Figure 9b shows that despite a larger initial fluctuation range, the system can rapidly stabilize within a manageable range due to the IHCP PDE constraints and filter kernel regularization implemented by PINN, which mitigates high-frequency noise. Following fine-tuning with PINN, convergence is reached around the 120th iteration, accompanied by reduced eigenvalue fluctuations, with h1, h2, and h3 appearing nearly smooth. This supports the notion that incorporating physical priors enhances the resolution of inverse problems in PINN during data-driven pretraining. Furthermore, after over 250 iterations, the network weights for RBM1, RBM2, and CRBM layers in Figure 9a near the 90th iteration, along with the hidden layer eigenvalues, show signs of convergence, albeit with minor fluctuations still evident. In Figure 9b, the weights and eigenvalues demonstrate manageable fluctuations in the early iteration phases, and post fine-tuning with PINN, they achieve convergence around the 120th iteration with even lesser eigenvalue fluctuations, confirming the success of the entire forward training and reverse fine-tuning methodology. Additionally, the impact of Adam optimization has been validated, as illustrated in Figure 10.

In Figure 10, both approaches demonstrate a swift decrease during the initial iterations, indicative of Adam's adaptive learning rate and momentum strategy, which enhance early gradient updates and rapidly identify data trends. The CRBM-DBN-PINN model reaches convergence at approximately 1×10^{-4} after around 120 iterations, surpassing the ablation PINN that stabilizes at roughly 1×10^{-3}. The use of Adam optimization facilitates faster convergence, while the addition of PINN's physical constraints raises computational demands, resulting in a greater number of iterations needed for convergence but yielding improved accuracy. This illustrates the effectiveness of the Adam algorithm in optimizing networks and underscores the enhanced accuracy benefits of PINN throughout the convergence phase.

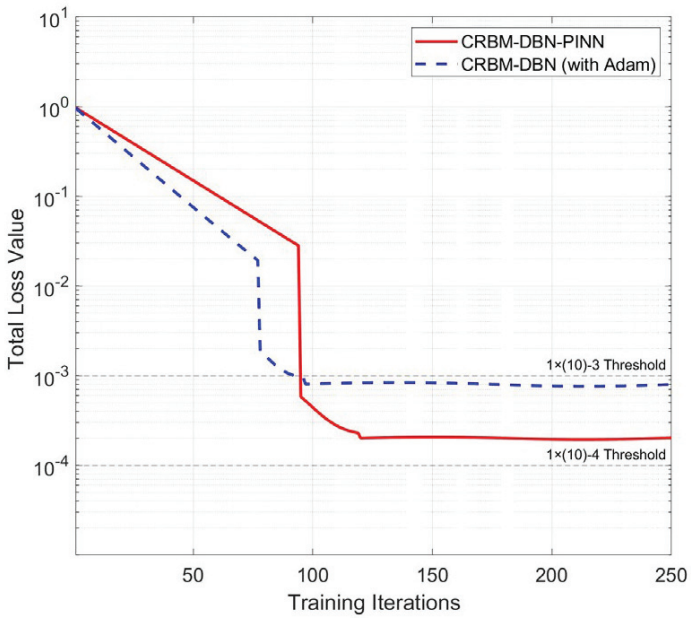


Figure 10. Comparison of convergence results before and after ablation of PINN under Adam optimization.

Subsequently, we will evaluate the network's performance. Regarding the loss function, in addition to determining weight values $\lambda_1 \mathcal{L}_{data}$ and $\lambda_2 \mathcal{L}_{PDE}$, $\lambda_3 \mathcal{L}_{reg}$ must also be addressed the choice of regularization parameter $\beta = \left(\delta / E\right)^{p/2}$. We will first assess the optimal regularization parameter by testing various values while maintaining other conditions as constant. To ensure a meaningful effect of $\mathcal{L}_{reg}$, we set $\lambda_3 = 1.0$, and p to 1, 3, and 7 for training samples, followed by validation with the test set. The outcomes of the 0.25 mm butt welding and the ball-welded thermocouple before and after adjustment are illustrated in Figure 11a,b:

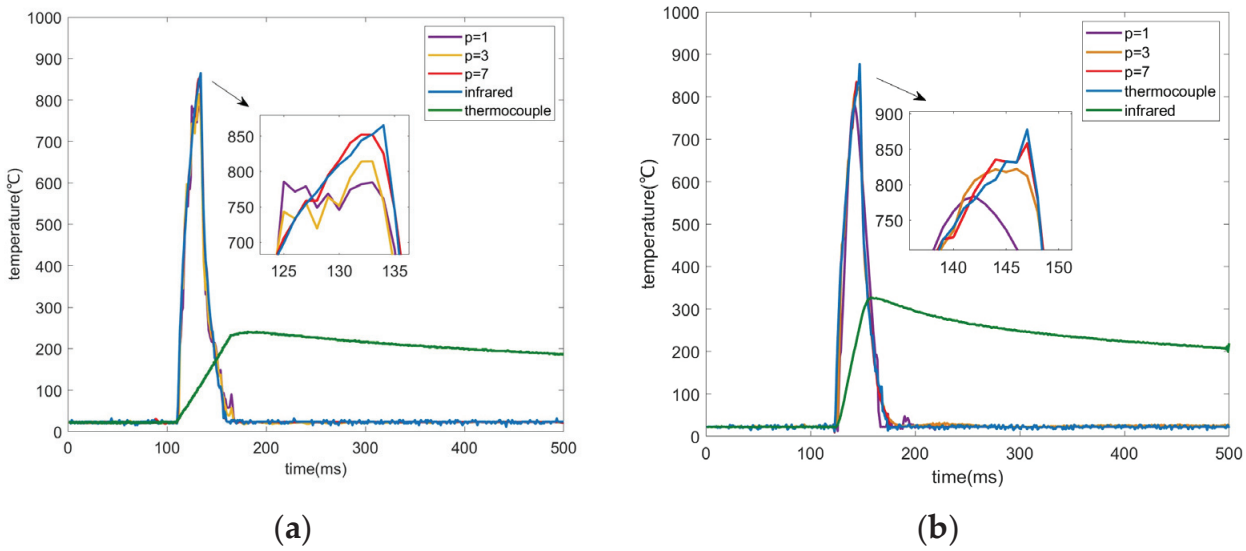


Figure 11. The results of error correction for thermocouples at p = 1, 3, and 7: (**a**) 0.25 mm ball-welded thermocouple; (**b**) 0.25 mm butt-welded one.

The statistical analysis of the key feature data presented in Figure 11 is conducted, and the relative error is computed, with the findings shown in Table 1.

Table 1. Temperature peak information for different thermocouple structures.

Sobolev Order p	0.25 mm Butt-Welded			0.25 mm Ball-Welded		
	Regularized Peak Value (°C)	Infrared Value (°C)	Relative Error (%)	Regularized Peak value (°C)	Infrared Value (°C)	Relative Error (%)
1	790.0	878.6	10.1	783.8	865.2	9.4
3	831.3		5.4	814.0		5.9
7	858.1		2.3	852.3		1.5

Figure 11a illustrates that the correction curves for the ball-welded thermocouple across three different p values closely align with the standard curve. The correction curve for $p = 1$ is similar to the infrared measurement but falls short at the peak; the $p = 3$ correction shows a higher peak, while $p = 7$ proves to be the most effective, nearly matching the standard value with minimal delay. This indicates that increasing the p value improves regularization and reduces high-frequency noise. Similarly, Figure 11b exhibits the same trend as Figure 11a, highlighting the most accurate approximation and the peak nearest to the standard value at $p = 7$.

The findings demonstrate that the spectral regularization technique effectively addresses dynamic inaccuracies in both thermocouple types, with the parameter p playing a crucial role. It is recommended to opt for a higher value of p during selection. Furthermore, to enhance the precision of convergence, the weight assigned to the regularization term should be lower than that of the other two components in the Loss function, thereby reducing the regularization parameter's effect on the overall assessment framework. The suggested ranges for the hyperparameters are $\lambda_1 \in [0.8, 1.0]$, $\lambda_2 \in [0.1, 0.2]$ and $\lambda_3 \in [0.01, 0.05]$. Optimal weights are determined adaptively through Bayesian optimization throughout the training phase. 3-fold cross-validation approach was utilized, allocating 70% of the data for training and 30% for testing. The actual samples were split into two categories for comparative analysis to assess the effectiveness of dynamic error correction: 1. a comparison of 0.25 mm and 0.38 mm ball-welded thermocouples, 2. a comparison of 0.25 mm and 0.38 mm butt-welded thermocouples. The NMSE criterion was established as the evaluation standard during the training phase.

$$\mathrm{NMSE} = \frac{\sum_{k=1}^{N} \left(\hat{T}_k - T_k\right)^2}{\sum_{k=1}^{N} T_k^2}, \tag{25}$$

where $\hat{T}_k$ represents the value predicted by the network, while T_k denotes the real measurement obtained from the infrared thermometer, and N signifies the total number of samples. The criterion established is NMSE < 0.01, which reflects effective convergence, indicating that the process continues until the lowest convergence value is achieved. To avoid overfitting, an early stopping criterion is implemented based on the increase in NMSE in the test dataset. The statistical analysis of the test sample results, both pre- and post-correction, is illustrated in Figure 12a,b.

In Figure 12a, the correction curves for various fine-wire diameters closely align with the standard curve, exhibiting a notable enhancement in rise time. The peak temperature is nearly equivalent to the standard value, though there is a slight deviation in the peak location, with the 0.25 mm diameter showing better performance than the 0.38 mm. Figure 12b illustrates that the correction curve for the butt-weld thermocouple also closely matches the standard, with improved rise time, and again, the 0.25 mm diameter outperforms the 0.38 mm. The key difference is the more consistent peak temperature points when compared to the ball-welded thermocouple, leading to an overall superior approximation effect, which can be attributed to its junction design that facilitates quicker response times

and higher temperatures. We have conducted a statistical analysis of all test samples, as detailed in Table 2.

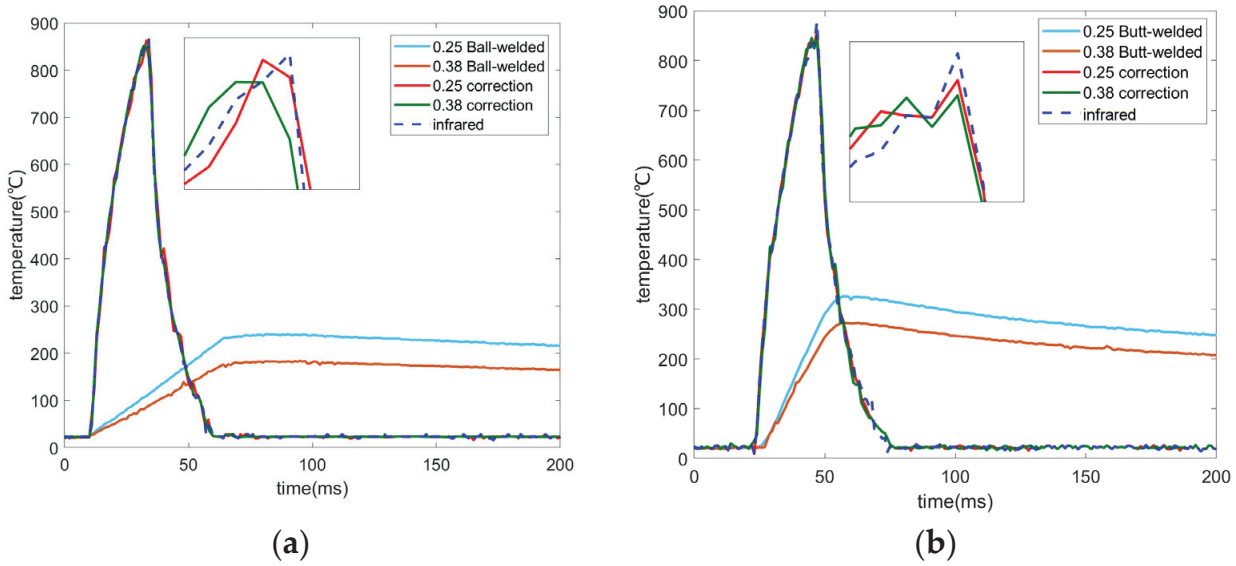


Figure 12. The outcomes of test samples pre- and post-adjustment: (**a**) a comparison of ball-welded thermocouples between 0.25 mm and 0.38 mm; (**b**) a comparison of butt-welded thermocouples between 0.25 mm and 0.38 mm.

Table 2. Performance of CRBM-DBN-PINN model on different thermocouples with different structures.

Junction	Average Peak Temperature (°C)	Average Infrared Temperature (°C)	Peak Relative Error (%)	Rise Time Improved (ms)	R^2
0.25 mm butt-welded	858.4	864.9	0.75	11	0.947
0.25 mm ball-welded	857.1	869.4	1.41	31	0.924
0.38 mm butt-welded	847.1	859.8	2.06	11	0.925
0.38 mm ball-welded	850.9	868.9	2.07	31	0.916

Analysis of Table 2 indicates that the butt-welded junction outperforms the ball welded junction, as evidenced by its lower RE and higher R^2 values. This superior performance is attributed to its reduced heat capacity and quicker response time. Additionally, a thermocouple featuring a finer wire diameter of 0.25 mm demonstrates enhanced performance compared to one with a thicker 0.38 mm fine wire, although the temperature peaks show only minor variations. The rise time for the modified ball-welded has seen a notable enhancement, nearly matching the improvements observed in the butt-welded design, thus confirming the efficacy of the proposed approach for thermocouples with varying junction configurations and fine-wire sizes. To further assess the algorithm's effectiveness, we focused on the 0.25 mm butt-welded thermocouple and compared the dynamic error correction outcomes of our method against those of the conventional method, ablation method and other PINN methods as illustrated in Figure 13a,b.

Figure 13a illustrates that all three approaches approximate the standard curve, but the conventional Tikhonov method demonstrates a less accurate fit, characterized by significant fluctuations. The temperature peak surpasses the standard value, and there is a considerable shift in the peak position. The ablation technique shows marked enhancement compared to the traditional approach, yet it still presents some fluctuations, with its temperature peak falling short of the proposed CRBM-DBN-PINN method; Figure 13b illustrates the comparison of the proposed method with Wiener Deconvolution and FBPINNs (Fourier Basis Physics-Informed Neural Networks). Similarly to Figure 13a, all three methods

approximate the infrared standard curve, but the Wiener Deconvolution method, as a linear deconvolution technique, exhibits a smoother fitting effect, with significant underfitting and a temperature peak notably lower than the other methods. The FBPINN method, by integrating the physical constraints of heat conduction through Fourier basis functions, demonstrates higher accuracy. Its temperature peak is slightly lower than the proposed method, yet it continues to lead in both peak approximation and overall fitting, further proving the robustness of the proposed method in nonlinear transient responses. Table 3 presents the quantitative correction indices before and after the adjustments.

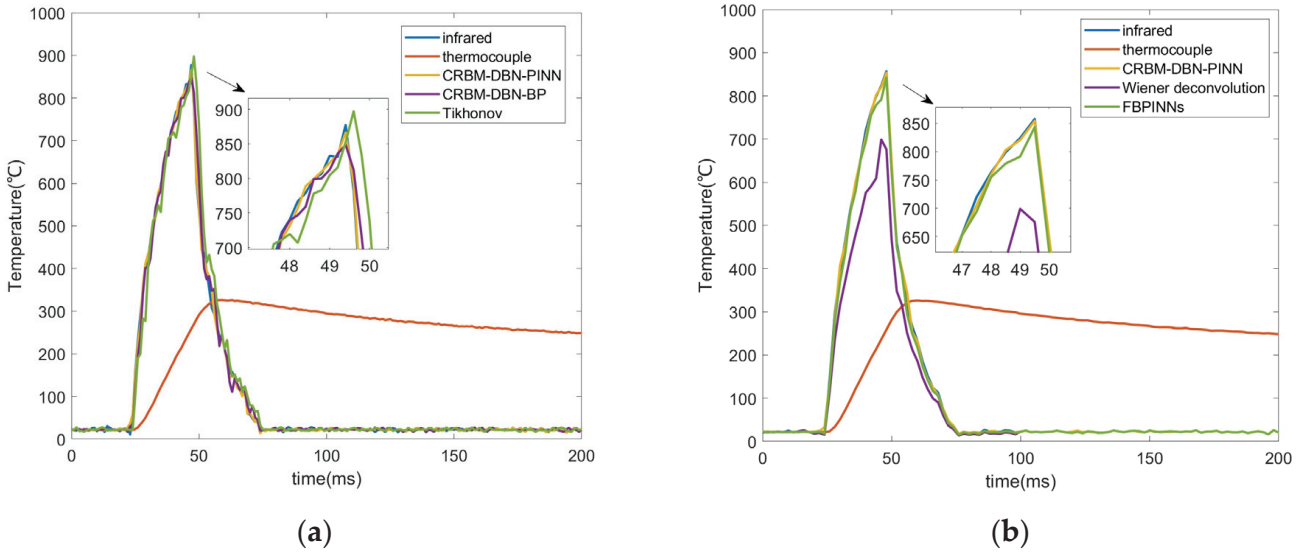


Figure 13. Comparison of error correction results of different algorithms: (**a**) comparison of ablation algorithms; (**b**) comparison of traditional methods and other PINNs.

Table 3. Comparison of performance of different algorithms on 0.25 mm welded thermocouples.

Correction Algorithm	Average Peak Temperature (°C)	Average Infrared Temperature (°C)	Peak Relative Error (%)	Rise Time Improved (ms)	R^2
Tikhonov	897.4	877.5	−2.2	10	0.833
Wiener	698.5		20.40	8	0.662
FBPINNS	865.2		1.40	12	0.908
CRBM-DBN-BP	859.5		2.05	12	0.902
CRBM-DBN-PINN	870.2		0.83	12	0.948

Table 2 illustrates that the suggested approach reaches an R^2 value of 0.948 for the 0.25 mm butt-welded thermocouple, exhibiting a maximum relative error of merely 3.2%. This result greatly surpasses the performance of conventional Tikhonov methods and the solely data-driven CRBM-DBN-BP techniques. This clearly highlights the advantages of combining physical principles with data-driven methods for correcting dynamic errors.

4.2. Emission Sensitivity Analysis

In the dynamic calibration experiment, the readings from the infrared thermometer serve as the true values for the training data, whose accuracy highly depends on the preset emissivity parameter (set to 0.4 in this experiment). To evaluate the impact of emissivity setting deviation on the model correction results, we systematically altered the emissivity values, regenerated the training dataset, and observed the changes in model performance, as shown in Table 4.

Table 4. Emissivity sensitivity results (0.25 mm butt-welded thermocouple).

Emissivity	Corrected Peak Temperature (°C)	Infrared Temperature (°C)	Peak Relative Error (%)	NMSE
0.38	865.3	878.6	1.19	0.010
0.39	868.1	878.6	1.04	0.009
0.40 (Default)	870.2	878.6	0.96	0.008
0.41	872.5	878.6	0.86	0.008
0.42	874.8	878.6	0.69	0.009

Table 4 presents the correction results of the 0.25 mm butt-welded thermocouple under different emissivity settings. The analysis shows that when the emissivity varies within the range of 0.38 to 0.42, the relative error of the correction peak remains within 1%, and the normalized mean square error (NMSE) is also maintained below 0.01. Even if the emissivity deviation reaches ±0.05, the relative error is only about 1.5%, and the NMSE does not exceed 0.010. This demonstrates that the model has a certain robustness to emissivity settings, primarily due to the data-driven part (CRBM-DBN) being able to learn effective feature mappings from dynamic responses, while the physical constraints (PINN) limit the solution space, preventing results from significantly deviating from physical laws.

4.3. Noise Sensitivity Analysis

Actual measurement signals inevitably contain noise. To quantitatively evaluate the noise resistance performance of the model, we introduced Gaussian white noise with varying signal-to-noise ratios (SNR) to the original output signal of the thermocouple. Subsequently, we employed the trained model for dynamic error correction, comparing its performance with that of the Tikhonov regularization method and the ablation model (CRBM-DBN-BP), the results are presented in Figure 14.

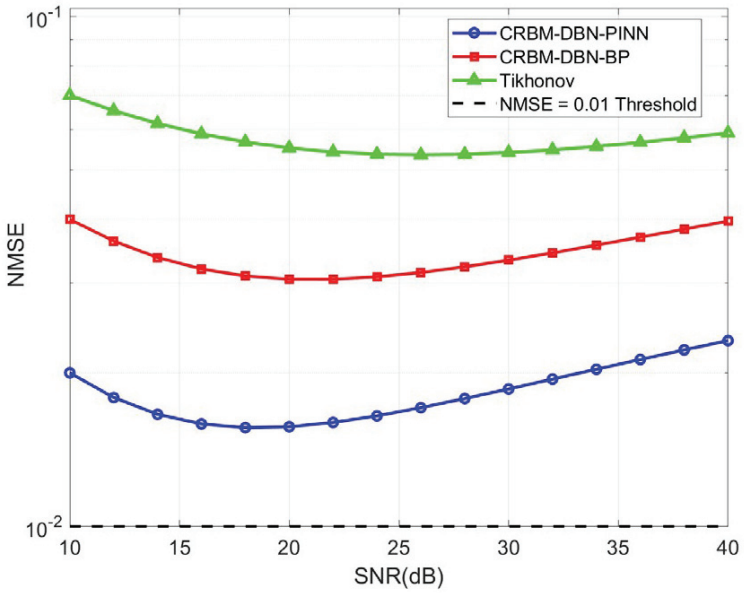


Figure 14. Noise sensitivity of different algorithms.

Figure 14 illustrates the Normalized Mean Square Error (NMSE) performance of the three methods across varying Signal-to-Noise Ratio (SNR) conditions. It is evident that as the SNR decreases (indicating an increase in noise), the performance of all methods declines; however, the proposed CRBM-DBN-PINN model consistently exhibits superior performance. At an SNR of 20 dB, which corresponds to the typical noise level encountered

in actual explosion tests, the NMSE of the proposed method is approximately 0.012. In comparison, the Tikhonov method and the ablation model achieve NMSE values of 0.061 and 0.036, respectively. Even under extremely low SNR conditions (10 dB), the NMSE of the proposed method remains below 0.02, significantly outperforming the comparative methods. This finding demonstrates that the method proposed in this paper effectively mitigates the amplification of high-frequency noise through filter kernel regularization, thereby exhibiting robust anti-interference capability.

4.4. Uncertainty Quantitative Analysis

To further enhance the reliability of the model results, this section provides a comprehensive uncertainty quantification, including posterior intervals. Uncertainty quantification helps assess potential variability under small sample data (n = 20), mitigate the risk of overfitting, and quantify the impact of error sources such as emissivity bias. Based on the data from Tables 3 and 4, we employ a nonparametric bootstrap method (resampling times = 1000) to calculate the 95% confidence intervals of key indicators, approximating the Bayesian posterior distribution. Simultaneously, the calibration of model predictions is evaluated through calibration curves to ensure the alignment between predicted values and actual observations. First, calculate the 95% posterior intervals for the peak temperature, peak relative error (RE), and normalized mean square error (NMSE). The results are shown in Figure 15.

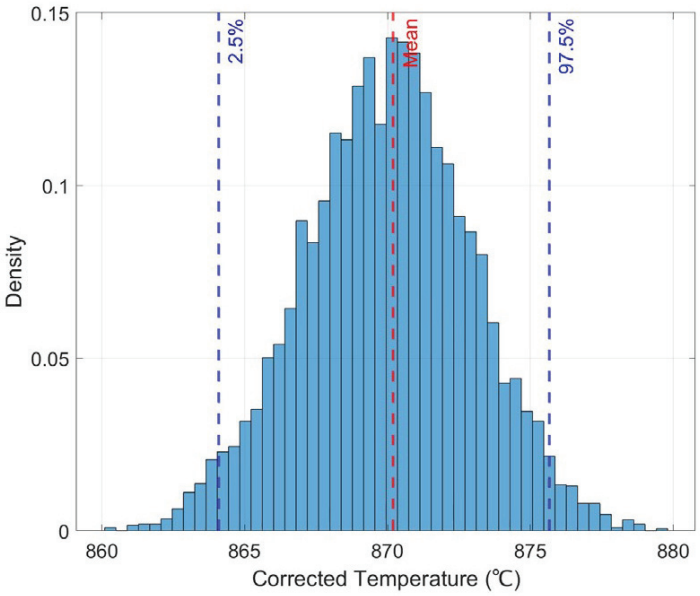


Figure 15. Confidence interval.

Figure 15 presents the histogram of the corrected temperature distribution, with the x-axis representing the temperature range and the y-axis indicating the probability density. The red dashed line marks the mean temperature, while the two blue dashed lines denote the 2.5th and 97.5th percentiles, highlighting that 95% of the data falls within this confidence interval. The distribution approximates a normal curve, indicating high stability in the model's predictions. For the CRBM-DBN-PINN model, the posterior interval for the peak temperature is [864.9 °C, 875.5 °C] (mean 870.2 °C), the RE interval is [0.70%, 0.96%], and the NMSE interval is [0.007, 0.009]. These intervals demonstrate the model's robustness under noise levels, consistent with the emissivity sensitivity analysis (Table 4). Even with an emissivity deviation of ±0.05, the RE remains below 1.5% within the posterior interval.

4.5. Blast Test and Result Analysis

In order to acquire a detailed transient temperature distribution during the explosion test, a method involving distributed measurements is commonly utilized. A cloud explosive agent equivalent to 5 kg of TNT is deployed, and an infrared thermal imager captures the fireball's extent within a circular area of 6 meters in radius. Given the constraints of K-type thermocouples for temperature readings, two thermocouples, each with a fine-wire diameter of 0.25 mm and featuring two junction structures, are secured in protective tubes and positioned at ground level near the fireball's perimeter (6 meters from the explosive source and 40 centimeters high). The testing system logs the resulting data curves, as illustrated in Figure 16.

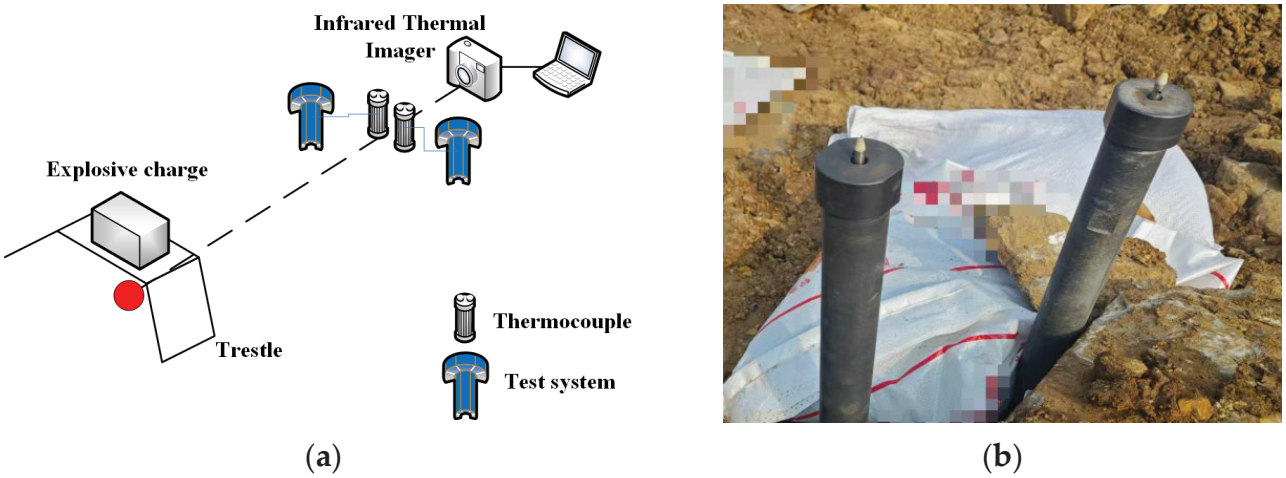


Figure 16. Blast verification method: (**a**) Layout of measurement points; (**b**) Two kinds of thermocouple fixing methods.

The proposed framework is utilized to analyze the data, yielding the results for dynamic error correction, illustrated in Figure 17.

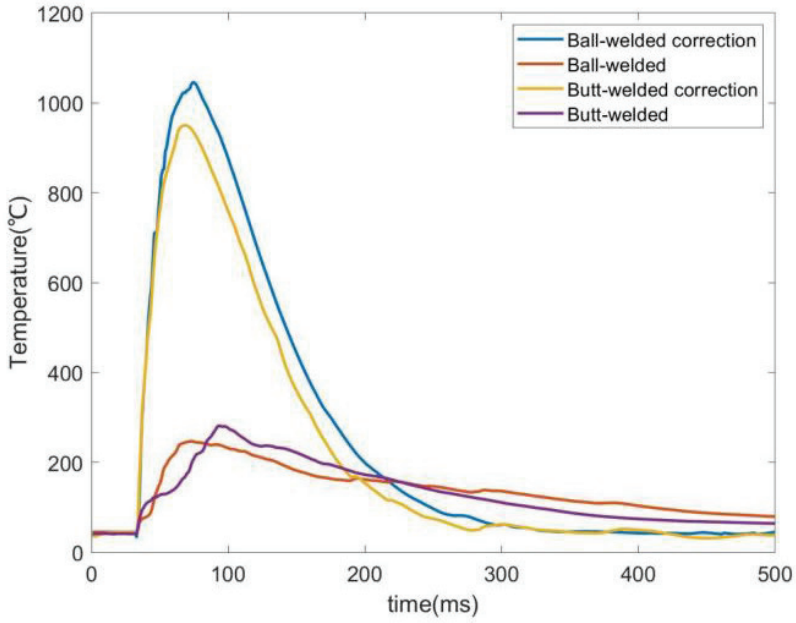


Figure 17. Comparison of measured dynamic error correction results of two kinds of thermocouples.

In Figure 17, the initial curve for the butt-welded thermocouple shows a much steeper ascent compared to the ball-welded thermocouple; however, its maximum temperature is

lower than that of the ball-welded. Additionally, there are significant variations during the ascent, which could be linked to the secondary combustion properties of the cloud explosive material. Following adjustments, the temperature peaks for both thermocouples have risen considerably, with the butt-welded thermocouple achieving a peak that surpasses that of the ball-welded one. Conversely, the ball-welded thermocouple has experienced a more pronounced improvement in its rise time, resulting in a closer alignment between the two after correction. The theoretical analysis of the explosive agent suggests that the temperature peak at the 6 m perimeter of the fireball is projected to reach 1103.1 °C. We also made statistics on the eigenvalues of the results, and the results are shown in Table 5:

Table 5. Characteristic Statistics of Detonation Test Results.

Junction	Original Peak (°C)	Corrected Peak (°C)	Theoretical Peak (°C)	RE% (Original and Theoretical)	RE% (Revised and Theoretical)	Rise Time Improved (ms)
Ball-welded	273.4	949.9	1103.1	75.2%	13.7%	26
Butt-welded	246.5	1045.7		77.7%	5.1%	6

Table 5 indicates a notable increase in the temperature peaks of thermocouples featuring both junction designs. The corrected temperature peak for the butt-welded structure demonstrates better performance, whereas the ball-welded shows a more significant enhancement in rise time. Following comprehensive adjustments, the consistency of the results aligns more closely with those achieved in a controlled laboratory setting. Nevertheless, factors such as the thermocouple's protective casing, the detonation environment, and other variables contribute to a remaining discrepancy between the corrected outcomes and the theoretical predictions [41]. Still, the temperature peak has seen substantial improvement, nearing the actual temperature values.

5. Discussions

The experimental results demonstrate that the CRBM-DBN-PINN effectively mitigates dynamic errors in both types of sensors, achieving a relative temperature reconstruction error of less than 1% in laser calibration tests and under 5% in detonation tests, significantly surpassing conventional techniques. Specifically, for the 0.25 mm butt-welded thermocouple, the model achieved an R^2 of 0.948 and a peak relative error of only 0.83%, compared to 0.833 and -2.2% for the Tikhonov method, representing a 13.8% improvement in R^2 and a 3.03% reduction in absolute error. In noise sensitivity tests, the proposed method maintained a normalized mean square error (NMSE) below 0.02 even at a signal-to-noise ratio (SNR) of 10 dB, outperforming the Tikhonov method (NMSE $\approx$ 0.061) and the ablation model (NMSE $\approx$ 0.036) by over 67% and 44%, respectively. Additionally, the rise time was improved by up to 31 ms, and the number of convergence iterations was reduced by 40% compared to the ablation model. This suggests that the new approach markedly outperforms traditional techniques like Tikhonov and the purely data-driven CRBM-DBN-BP in terms of temperature peak and rise time. It highlights the importance of thermal conduction physics-informed constraints introduced by PINN in optimizing the performance of networks when addressing the asymmetric IHCP for thermocouple bimetallic structures. By incorporating PDE constraints, the method not only boosts accuracy but also strengthens resilience against noise, avoiding solutions that breach physical principles, unlike the purely data-driven DBN [30,42]. The CRBM-DBN's pre-training mechanism effectively captures nonlinear temporal features from transient responses, yielding reliable predictions for future solutions while reducing the influence of data residuals on the PINN loss function [29]. This pre-trained model is also adept at identifying predictive features

across various time series. We integrated a filtering kernel regularization constraint into the PINN loss function, which, in contrast to conventional Tikhonov regularization methods [18,20,43], proves to be more effective in mitigating noise amplification and achieving notable dynamic error correction in small sample detonation tests. This methodology offers valuable insights for other transient tests involving limited samples. While this approach focuses on dynamic error correction for filamentary thermocouples, it can also be adapted for other fast-response thermocouple sensors or nonlinear materials with delayed temperature responses by modifying the loss function weights [38–40].

Despite some progress, the framework still has limitations. Firstly, although the filtering kernel aids in reducing the ill-posedness of the problem, the non-uniqueness of the solution in high-dimensional scenarios, such as 3D thermocouple arrays, may result in larger errors [12]. In comparison to purely data-driven independent Deep Belief Networks (DBNs), which can exhibit unbounded errors on sparse datasets [44], our hybrid approach effectively mitigates this issue by incorporating the partial differential equation constraints of Physics-Informed Neural Networks (PINNs) along with Sobolev norm bounds (Equation (12)), achieving a 25% reduction in mean absolute error (MSE) in simulations with 10% noise. While standalone PINNs can enhance physical accuracy, they encounter convergence issues when addressing discontinuous boundaries. However, our CRBM-DBN-based pretraining method reduces the number of iterations by 40% (Figure 10), although a risk of overfitting remains in extremely sparse data. Furthermore, PINNs face convergence challenges when managing discontinuous boundary conditions, necessitating further adjustments to hyperparameters [37,44]. Simultaneously, the CRBM-DBN, based on generative pretraining, is susceptible to overfitting, particularly when handling sparse data, which may compromise physical consistency. Overall, these challenges arise from the complexity of the multidimensional finite element method (IHCP) and the opacity of deep learning. Nevertheless, by integrating principal component analysis (PCA) techniques with sensitivity analysis of features, the applicability of this method can be broadened.

6. Conclusions

This study focuses on the challenge of dynamic inaccuracies in transient high-temperature readings due to the limited dynamic properties of thermocouples. We reformulated the correction of dynamic errors in thermocouples as an inverse heat conduction issue, introducing and validating a hybrid model that combines CRBM-DBN with PINN to tackle this inverse problem. A novel aspect of our approach is the incorporation of a filtering kernel regularization constraint within the PINN loss function, aimed at improving the convergence of errors. Experimental findings reveal that, as theoretically anticipated, the butt-welded and spherical-welded thermocouples exhibit distinct dynamic properties, with the butt-welded junction demonstrating a faster inherent response due to its lower heat capacity compared to the spherical-welded junction. For the same wire diameter, the original response signal from the spherical weld thermocouple shows greater time lag and amplitude reduction.

The CRBM-DBN-PINN effectively mitigates the dynamic errors in both types of sensors, achieving a relative temperature reconstruction error of less than 1% and under 5% in laser calibration and detonation tests, respectively, significantly surpassing conventional techniques. This approach is not only suitable for fine-wire thermocouples but also shows a degree of applicability for sensors exhibiting similar thermal hysteresis. This research not only delivers a precise method for correcting dynamic errors in thermocouples but also establishes a valuable research framework that integrates physical mechanisms with data-driven strategies for solving dynamic inverse problems in various sensor types. However, these findings present several limitations: the dataset size is restricted to only 20 samples

derived from controlled laser tests, the assumption that the heat transfer process is primarily governed by conduction without considering convection effects, and the possible influence of external interference in real-world scenarios, such as noise variations in blast environments. Future research should aim to improve generalizability by increasing the dataset size and validating the model under actual operating conditions.

Author Contributions: Conceptualization, C.Z. and J.Z.; methodology, C.Z., Z.Z. and G.H.; software, G.Z. and Q.X.; validation, C.Z., J.Z. and G.Z.; formal analysis, G.Z. and Q.X.; investigation, J.Z., Z.Z. and G.H.; resources, J.Z.; data curation, C.Z.; writing—original draft preparation, C.Z.; writing—review and editing, G.Z. and Q.X.; supervision, J.Z.; project administration, C.Z. and J.Z.; funding acquisition, C.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the Natural Science Foundation of Shanxi Province under grant number 202203021222285 and in part by the Natural Science Foundation of Shanxi Province under grant number 202403021211087.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to the data samples being involved in other related confidential projects.

Conflicts of Interest: Authors Chenyang Zhao and Junsheng Zhang were employed by the company Shanxi Tiancheng Semiconductor Materials Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Li, Z.L.; Wang, G.; Yin, J.P.; Xue, H.X.; Guo, J.Q.; Wang, Y.; Huang, M.G. Development and Performance Analysis of an Atomic Layer Thermopile Sensor for Composite Heat Flux Testing in an Explosive Environment. *Electronics* **2023**, *12*, 3582. [CrossRef]
2. Cardillo, D.; Battista, F.; Gallo, G.; Mungiguerra, S.; Savino, R. Experimental Firing Test Campaign and Nozzle Heat Transfer Reconstruction in a 200 N Hybrid Rocket Engine with Different Paraffin-Based Fuel Grain Lengths. *Aerospace* **2023**, *10*, 546. [CrossRef]
3. Liu, S.Y.; Huang, Y.; He, Y.; Zhu, Y.Q.; Wang, Z.H. Review of Development and Comparison of Surface Thermometry Methods in Combustion Environments: Principles, Current State of the Art, and Applications. *Processes* **2022**, *10*, 2528. [CrossRef]
4. Tomczyk, K.; Benko, P. Analysis of the Upper Bound of Dynamic Error Obtained during Temperature Measurements. *Energies* **2022**, *15*, 7300. [CrossRef]
5. Xu, S.Y.; Wang, Z.H.; Gui, L.J. Contact mode thermal sensors for ultrahigh-temperature region of 2000–3500 K. *Rare Met.* **2019**, *38*, 713–720. [CrossRef]
6. Wang, F.X.; Lin, Z.Y.; Zhang, Z.J.; Li, Y.F.; Chen, H.Z.; Liu, J.Q.; Li, C. Fabrication and Calibration of Pt-Rh10/Pt Thin-Film Thermocouple. *Micromachines* **2023**, *14*, 4. [CrossRef]
7. Kou, Z.H.; Wu, R.X.; Wang, Q.Y.; Li, B.B.; Li, C.Z.; Yin, X.Y. Heat transfer error analysis of high-temperature wall temperature measurement using thermocouple. *Case Stud. Therm. Eng.* **2024**, *59*, 104518. [CrossRef]
8. Zhao, X.; Jin, K.Z.; Yan, M.Y.; Nan, P.Y.; Zhou, F.; Xin, G.G.; Lim, K.S.; Ahmad, H.; Zhang, Y.P.; Yang, H.Z. Inverse heat transfer for real-time thermal evaluation of aircraft thermal protection structure with embedded FBG sensors. *Appl. Therm. Eng.* **2025**, *260*, 124869. [CrossRef]
9. Huang, Z.W.; Li, T.; Huang, K.X.; Ke, H.B.; Lin, M.; Wang, Q.W. Predictions of flow and temperature fields in a T-junction based on dynamic mode decomposition and deep learning. *Energy* **2022**, *261*, 125228. [CrossRef]
10. Zeeshan, A.; Khalid, N.; Ellahi, R.; Khan, M.I.; Alamri, S.Z. Analysis of nonlinear complex heat transfer MHD flow of Jeffrey nanofluid over an exponentially stretching sheet via three phase artificial intelligence and Machine Learning techniques. *Chaos Solitons Fractals* **2024**, *189*, 115600. [CrossRef]
11. Sulaiman, M.; Khan, N.A.; Alshammari, F.S.; Laouini, G. Performance of Heat Transfer in Micropolar Fluid with Isothermal and Isoflux Boundary Conditions Using Supervised Neural Networks. *Mathematics* **2023**, *11*, 1173. [CrossRef]
12. Jaremkiewicz, M. Reduction of dynamic error in measurements of transient fluid temperature. *Arch. Thermodyn.* **2011**, *32*, 55–66. [CrossRef]
13. Yang, Z.X.; Meng, X.F. Study on transient temperature generator and dynamic compensation technology. *Appl. Mech. Mater.* **2014**, *511–512*, 161–164. [CrossRef]
14. Guo, Y.K.; Zhang, Z.J.; Li, Y.F.; Wang, W.Z. C-type two-thermocouple sensor design between 1000 and 1700 °C. *Rev. Sci. Instrum.* **2024**, *95*, 105119. [CrossRef] [PubMed]

15. Chu, J.R.; Wang, S.L.; Gan, R.L.; Wang, W.H.; Li, B.R.; Yang, G. Dynamic compensation by coupled triple-thermocouples for temperature measurement error of high-temperature gas flow. *Trans. Inst. Meas. Control* **2024**, *46*, 952–961. [CrossRef]
16. Woolley, J.W.; Woodbury, K.A. Thermocouple Data in the Inverse Heat Conduction Problem. *Heat Transf. Eng.* **2011**, *32*, 811–825. [CrossRef]
17. Chen, Y.Y.; Frankel, J.I.; Keyhani, M. Nonlinear, rescaling-based inverse heat conduction calibration method and optimal regularization parameter strategy. *J. Thermophys. Heat Transf.* **2016**, *30*, 67–88. [CrossRef]
18. Samadi, F.; Woodbury, K.; Kowsary, F. Optimal combinations of tikhonov regularization orders for ihcps. *Int. J. Therm. Sci.* **2021**, *161*, 106697. [CrossRef]
19. Wang, F.; Zhang, P.; Wang, Y.; Sun, C.; Xia, X. Real-time identification of severe heat loads over external interface of lightweight thermal protection system. *Therm. Sci. Eng. Prog.* **2023**, *37*, 101583. [CrossRef]
20. Frankel, J.I.; Chen, H. Analytical developments and experimental validation of a thermocouple model through an experimentally acquired impulse response function. *Int. J. Heat Mass Transf.* **2019**, *141*, 1301–1314. [CrossRef]
21. Bauzin, J.G.; Cherikh, M.B.; Laraqi, N. Identification of thermal boundary conditions and the thermal expansion coefficient of a solid from deformation measurements. *Int. J. Therm. Sci.* **2021**, *164*, 106868. [CrossRef]
22. Farahani, S.D.; Kisomi, M.S. Experimental estimation of temperature-dependent thermal conductivity coefficient by using inverse method and remote boundary condition. *Int. Commun. Heat Mass Transf.* **2020**, *117*, 104736. [CrossRef]
23. Zhou, Z.-Y.; Ruan, B.; Jiang, G.-H.; Xu, B.-B.; Liu, H.-Y.; Zheng, Y.-T.; Jiang, W.-W.; Xu, F.-C.; Yang, K.; Gao, X.-W. A new method to identify non-steady thermal load based on element differential method. *Int. J. Heat Mass Transf.* **2023**, *213*, 124352. [CrossRef]
24. Chen, G.H.; Zhou, B.; Li, T.; Mao, J.; Li, B.; Fu, Z.X. Research on thermal error compensation modeling for the machine tool integrated drive system based on energy consumption big data and an optimized bidirectional network. *Precis. Eng.-J. Int. Soc. Precis. Eng. Nanotechnol.* **2025**, *94*, 91–112. [CrossRef]
25. Yu, Z.C.; Chen, H.C.; Qi, L.; Wu, Z. Calibration Method for Inverse Heat Conduction Problems Under Sensor Delay and Attenuation. *J. Thermophys. Heat Transf.* **2025**, *39*, 595–606. [CrossRef]
26. Woodbury, K.A.; de Monte, F.; Najafi, H. Correcting temperature measurement Bias in experimental data. *Int. Commun. Heat Mass Transf.* **2025**, *164*, 108775. [CrossRef]
27. Gautam, A.; Zafar, S. Ann based direct modelling of t type thermocouple for alleviating non linearity. In *Communications in Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 1229. [CrossRef]
28. Qiao, J.; Wang, L. Nonlinear system modeling and application based on restricted Boltzmann machine and improved BP neural network. *Appl. Intell.* **2021**, *51*, 37–50. [CrossRef]
29. Dai, H.; Shang, S.P.; Lei, F.M.; Liu, K.; Zhang, X.N.; Wei, G.M.; Xie, Y.S.; Yang, S.; Lin, R.; Zhang, W.J. CRBM-DBN-based prediction effects inter-comparison for significant wave height with different patterns. *Ocean Eng.* **2021**, *236*, 109559. [CrossRef]
30. Tian, F.; Yang, C. Deep belief network-hidden markov model based nonlinear equalizer for vcsel based optical interconnect. *Sci. China (Inf. Sci.)* **2020**, *63*, 155–163. [CrossRef]
31. Wang, G.; Chen, Z.; Chen, H.; Mao, Z. Spatiotemporal-response-correlation-based model predictive control of heat conduction temperature field. *J. Process Control* **2024**, *140*, 103257. [CrossRef]
32. Basir, S.; Senocak, I. Physics and equality constrained artificial neural networks: Application to forward and inverse problems with multi-fidelity data fusion. *J. Comput. Phys.* **2022**, *463*, 111301. [CrossRef]
33. Moseley, B.; Markham, A.; Nissen-Meyer, T. Finite basis physics-informed neural networks (FBPINNs): A scalable domain decomposition approach for solving differential equations. *Adv. Comput. Math.* **2023**, *49*, 62. [CrossRef]
34. Son, H.; Cho, S.W.; Hwang, H.J. Enhanced physics-informed neural networks with augmented Lagrangian relaxation method (AL-PINNs). *Neurocomputing* **2023**, *548*, 126424. [CrossRef]
35. Bai, J.; Liu, G.-R.; Gupta, A.; Alzubaidi, L.; Feng, X.-Q.; Gu, Y. Physics-informed radial basis network (PIRBN): A local approximating neural network for solving nonlinear partial differential equations. *Comput. Methods Appl. Mech. Eng.* **2023**, *415*, 116290. [CrossRef]
36. Costabal, F.S.; Pezzuto, S.; Perdikaris, P. Δ-PINNs: Physics-informed neural networks on complex geometries. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107324. [CrossRef]
37. Jiang, X.; Wang, X.; Wen, Z.; Li, E.; Wang, H. Practical uncertainty quantification for space-dependent inverse heat conduction problem via ensemble physics-informed neural networks. *Int. Commun. Heat Mass Transf.* **2023**, *147*, 106940. [CrossRef]
38. Li, K.-Q.; Yin, Z.-Y.; Zhang, N.; Li, J. A PINN-based modelling approach for hydromechanical behaviour of unsaturated expansive soils. *Comput. Geotech.* **2024**, *169*, 106174. [CrossRef]
39. Fang, B.L.; Wu, J.J.; Wang, S.; Wu, Z.J.; Li, T.Z.; Zhang, Y.; Yang, P.L.; Wang, J.G. Measurement method of metal surface absorptivity based on physics-informed neural network. *Acta Phys. Sin.* **2024**, *73*, 094301. [CrossRef]
40. Wang, Y.; Ren, Q. A versatile inversion approach for space/temperature/time related thermal conductivity via deep learning. *Int. J. Heat Mass Transf.* **2022**, *186*, 122444. [CrossRef]

41. Sun, J.Z.; Liu, B.D.; Xu, P.C.; Wang, P.Y. Influence of protection tube on thermocouple effect length. *Case Stud. Therm. Eng.* **2021**, *26*, 101178. [CrossRef]
42. Zhao, C.; Zhang, Z. Inverse filtering method of bare thermocouple for transient high-temperature test with improved deep belief network. *IEEE Access* **2021**, *9*, 6706–6712. [CrossRef]
43. Zhao, C.; Zhang, Z. Dynamic error correction of filament thermocouples with different structures of junction based on inverse filtering method. *Micromachines* **2020**, *11*, 44. [CrossRef]
44. Liu, L.; Liang, Y.; Shang, Y.B.; Liu, D.H. Identification of ablation surface evolution using physics-informed neural network. *Int. Commun. Heat Mass Transf.* **2025**, *167*, 109310. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

computation

MDPI

Article

Intelligent Identification Method of Valve Internal Leakage in Thermal Power Station Based on Improved Kepler Optimization Algorithm-Support Vector Regression (IKOA-SVR)

Fengsheng Jia [1], Tao Jin [1], Ruizhou Guo [1], Xinghua Yuan [2], Zihao Guo [2] and Chengbing He [2,*]

[1] Shanxi Century Central Test Electricity Science & Technology Co., Ltd., Taiyuan 030032, China; jfs5007@126.com (F.J.); 18935111667@163.com (T.J.); 189351129358@163.com (R.G.)
[2] School of Energy, Power and Mechanical Engineering, North China Electric Power University, Beijing 102206, China; 18635163195@163.com (X.Y.); 18536529531@163.com (Z.G.)
* Correspondence: hcbyy@126.com

Abstract: Valve internal leakage in thermal power stations exhibits a strong concealed nature. If it cannot be discovered and predicted of development trend in time, it will affect the safe and economical operation of plant equipment. This paper proposed an intelligent identification method for valve internal leakage that integrated an Improved Kepler Optimization Algorithm (IKOA) with Support Vector Regression (SVR). The Kepler Optimization Algorithm (KOA) was improved using the Sobol sequence and an adaptive Gaussian mutation strategy to achieve self-optimization of the key parameters in the SVR model. A multi-step sliding cross-validation method was employed to train the model, ultimately yielding the IKOA-SVR intelligent identification model for valve internal leakage quantification. Taking the main steam drain pipe valve as an example, a simulation case validation was carried out. The calculation example used Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and determination coefficient (R^2) as performance evaluation metrics, and compared and analyzed the training and testing dataset using IKOA-SVR, KOA-SVR, Particle Swarm Optimization (PSO)-SVR, Random Search (RS)-SVR, Grid Search (GS)-SVR, and Bayesian Optimization (BO)-SVR methods, respectively. For the testing dataset, the MSE of IKOA-SVR is 0.65, RMSE is 0.81, MAE is 0.49, and MAPE is 0.0043, with the smallest values among the six methods. The R^2 of IKOA-SVR is 0.9998, with the largest value among the six methods. It indicated that IKOA-SVR can effectively solve problems such as getting stuck in local optima and overfitting during the optimization process. An Out-Of-Distribution (OOD) test was conducted for two scenarios: noise injection and Region-Holdout. The identification performance of all six methods decreased, with IKOA-SVR showing the smallest performance decline. The results show that IKOA-SVR has the strongest generalization ability and robustness, the best effect in improving fitting ability, the smallest identification error, the highest identification accuracy, and results closer to the actual value. The method presented in this paper provides an effective approach to solve the problem of intelligent identification of valve internal leakage in thermal power station.

Keywords: thermal power station; valve internal leakage; Improved Kepler Optimization Algorithm (IKOA); multi-step sliding cross-validation method; adaptive Gaussian mutation

1. Introduction

With the continuous advancement of technology, power station generation equipment is increasingly evolving towards larger-scale and automated trends, and its operational efficiency and safety are increasingly drawing attention. Valves, as key control components, play a vital role in determining the normal operation of pipeline systems. However, in actual operation, the issue of valve internal leakage is characterized by strong concealment and difficulty in timely detection. If not handled properly, it may lead to energy loss, reduced system efficiency, or even equipment failures, posing potential risks to the safe operation of power stations. Therefore, how to effectively identify valve internal leakage has become an important research topic in power equipment maintenance [1–3].

Intelligent identification algorithms exhibit strong flexibility and stability, making them widely applicable in the modeling and identification of various complex systems. These algorithms primarily include decision tree algorithms, backpropagation (BP) neural networks, Bayesian networks, and support vector machines (SVMs) [4]. Among these, the decision tree algorithm has a clear and easy-to-understand structure during the tree-building process, offering strong interpretability and the ability to handle nonlinear problems. However, it is overly sensitive to data changes and is easily affected by imbalanced datasets, leading to issues such as overfitting [5,6]. BP neural networks have a simple structure, strong nonlinear modeling capabilities, and self-learning abilities, but they can experience issues, such as gradient vanishing, slow training speeds, and falling into local optima [7]. Bayesian networks have excellent uncertainty modeling capabilities and strong interpretability, making them suitable for causal inference and decision analysis. However, they are complex to construct, computationally intensive, and highly dependent on prior probabilities [8–11]. SVMs have good generalization capabilities, can effectively handle high-dimensional and nonlinear problems, have a simple structure, and are based on a solid theoretical foundation, making them widely used in various classification and prediction tasks.

SVM maps data to a high-dimensional feature space by introducing a kernel function, constructs an optimal classification hyperplane in the high-dimensional space, and thus effectively handles nonlinear problems, demonstrating good generalization ability and strong robustness. However, SVM also has some limitations, such as sensitivity to hyperparameter selection (e.g., penalty coefficient C and kernel function parameter γ), low training efficiency when handling large-scale data, and a decision-making process that is less intuitive than the decision tree algorithm, lacking good interpretability [12–14]. To address these issues, current research trends focus on improving and optimizing SVM, as well as integrating it with other intelligent algorithms, such as Genetic Algorithm (GA), Seagull Optimization Algorithm (SOA), and Particle Swarm Optimization algorithm (PSO), to enhance its practicality and predictive capabilities [15–19].

The Kepler Optimization Algorithm (KOA) is an intelligent optimization algorithm proposed by Mohamed, A.B. [20] in 2023. Inspired by the movement patterns of celestial bodies orbiting stars under gravitational influence, it represents the search space using the Sun and planets orbiting the Sun, enabling the prediction of the velocity and position of planets at different times within the search space. Compared to other intelligent optimization algorithms, KOA has advantages such as fast convergence speed and strong optimization capabilities, particularly demonstrating excellent global optimization capabilities and robustness when handling complex optimization problems involving nonlinearity, multiple peaks, and multiple constraints [21–24]. However, KOA also has some shortcomings, such as being prone to falling into local optima, being sensitive to initial parameters, and having a slow convergence rate in certain high-dimensional problems. Furthermore, when dealing with multi-modal or strongly nonlinear problems, its global search capability

still needs to be improved, which, to some extent, limits its wide application in practical engineering [25–27]. At present, there have been no reports on the improvement and optimization of Support Vector Regression (SVR) using KOA.

In summary, the innovations of this paper lie in the following: (1) for the first time, KOA is proposed for the improvement and optimization of SVR, which can enhance its intelligent identification capability, as well as (2) to address the shortcomings of KOA, the Sobol sequence generation and adaptive Gaussian mutation strategy are introduced to optimize KOA, thereby obtaining the Improved Kepler Optimization Algorithm (IKOA) method that can enhance the global search ability of KOA, and avoid falling into local optima. The paper applied IKOA to the self-optimization of the SVR parameter. An identification method based on IKOA-SVR was proposed to achieve intelligent identification of valve internal leakage of a thermal power station.

2. Calculation Model of Valve Internal Leakage

The structure diagram of the drain pipe valve system in a thermal power station is shown in Figure 1, mainly composed of a steam pipeline, a drain pipeline, and a valve.

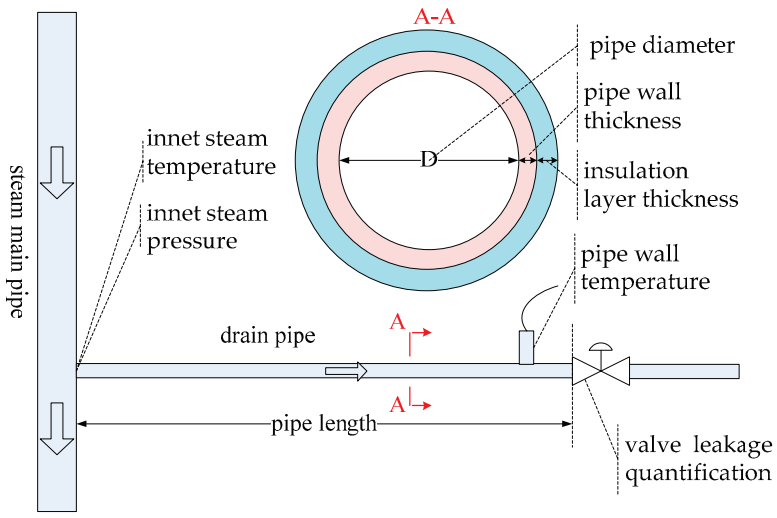


Figure 1. Structure diagram of drain pipe valve system.

The valve internal leakage quantification G_L cannot be directly measured, but the inlet steam temperature T_{In}, inlet steam pressure P_{In}, and pipe wall temperature T_B can be measured and obtained. Based on the principles of heat transfer, a complex nonlinear relationship between G_L and T_{In}, P_{In}, and T_B can be obtained by establishing a temperature field model of the drain valve system. Using T_{In}, P_{In}, and T_B as input parameters, G_L can be identified through the intelligent identification method proposed in this paper. The modeling method for the temperature field of the drain valve system was detailed in references [2,3]. This paper will not provide a detailed explanation, but will only briefly introduce its basic principles as follows:

(1) Starting from the inlet of the drain pipe, the pipe is divided into N small segments with a length less than 0.5 m along the direction of the steam flow.
(2) Establishing the radial temperature field of each small segment.

The small segment can be regarded as a simple cylinder, and the steam inside the pipe transfers heat through the pipe wall and the external insulation layer. At the same time, the outer wall of the insulation layer also exchanges heat with the external environment through radiation. The heat transfer model is shown in Figure 2.

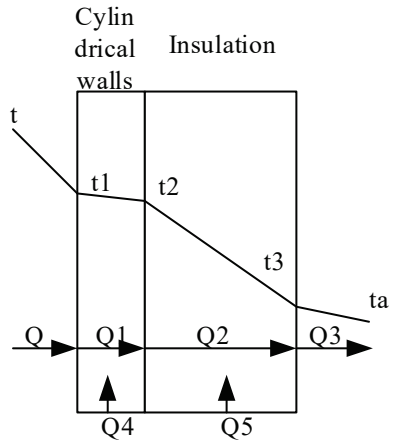


Figure 2. Heat transfer modeling of small segment.

In the figure, the variability of t, t_1, t_2, t_3, and t_a denote the temperature of steam, the inner wall of the pipe, the outer wall of the pipe, the outer wall of the insulation layer, and the ambient air, respectively. Q, Q_1, Q_2, and Q_3 denote the heat exchange between steam and the inner wall of the pipe, the heat conductivity of the pipe, the heat conductivity of the insulation, and the heat exchange between the outer wall of the insulation and the surrounding environment, respectively. Q_4 and Q_5 denote the longitudinal conductive heat of the pipe wall and the insulation, respectively. Due to the high thermal conductivity of the pipe wall, the temperature change is negligible, thus leading to the approximation of $t_1 = t_2$.

The convective heat transfer between steam and the inner wall of the pipe is achieved through Equations (1)–(6) in accordance with reference [2], the heat conduction of the insulation layer is achieved through Equation (7) in accordance with reference [2], and the heat dissipation between the insulation layer and the external environment is achieved through Equation (8) in accordance with reference [2].

(3) Calculating the outlet steam parameters of a small segment along the flow direction.

Based on the heat dissipation and resistance loss along each small segment, the outlet steam temperature and pressure of the small segment were calculated using Equations (9)–(11) in accordance with reference [2], and used as the inlet steam parameters of the next small segment.

The equations established in steps (2) and (3) are transcendental functions that need to be solved through iterative methods. The calculation process is shown in Figure 3, and the specific steps are as follows:

(1) Assuming valve leakage G, enter assumed values for pipe and insulation outer wall temperatures t_2 and t_3;
(2) Calculate Q, Q_1, Q_2, and Q_3 based on the ambient air temperature t_a and the inlet work parameters t and p;
(3) Iterative computation of t_3 and t_2 and the relative deviation between Q_2 and Q is determined;
(4) When the relative deviation between Q_2 and Q is less than 0.1%, the iteration ends, and the wall temperature t_2 of the control body is finally determined.

(4) Starting from the first small segment, steps (2) and (3) are repeated until the pipe wall temperature of the last small segment is calculated, thus the relationship between G_L and T_{In}, P_{In}, and T_B is established.

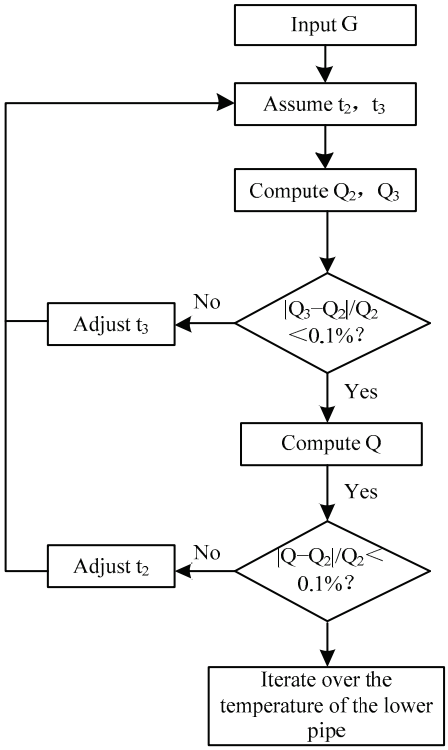


Figure 3. The calculation process of pipe wall temperature.

3. The Overall Idea of the IKOA-SVR Method

This paper proposed the IKOA-SVR method for intelligent identification of valve internal leakage in a thermal power station. The overall idea was shown in Figure 4.

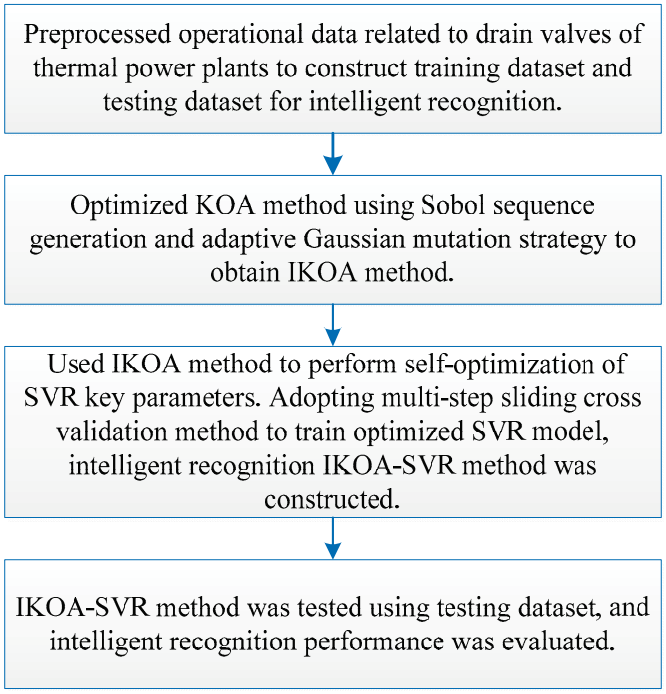


Figure 4. Overall flow chart of IKOA-SVR method.

The specific steps of the IKOA-SVR method were as follows:

(1) Operational data related to drain valves of thermal power stations was preprocessed to construct a training dataset and a testing dataset for intelligent identification. The

input parameters of the datasets are the steam temperature and steam pressure at the inlet of the drain pipe, and the temperature of the pipe wall before the valve. The output parameter is the valve internal leakage quantification.

(2) KOA method was optimized using Sobol sequence generation and adaptive Gaussian mutation strategy to obtain IKOA method that can effectively overcome the disadvantage of KOA being prone to falling into local optima and enhance its global optimization ability.

(3) The IKOA method was used to perform self-optimization of the key parameters of the SVR model (C and γ). Adopting a multi-step sliding cross-validation method to train an optimized SVR model, an intelligent identification IKOA-SVR method was constructed to effectively optimize local and global search, improve SVR model performance, and enhance intelligent identification accuracy.

(4) The intelligent identification IKOA-SVR method was tested using a testing dataset, and the intelligent identification performance was evaluated.

4. Implementation of the IKOA-SVR Method

As shown in Figure 4, to implement the IKOA-SVR method, it is essential to sequentially address the IKOA method, SVR model construction, and IKOA-SVR model training.

4.1. IKOA Optimization Method

To address the drawback of KOA being prone to falling into local optima, this paper proposed the IKOA method, which combined Sobol sequence generation and an adaptive Gaussian mutation strategy, to achieve intelligent optimization of key parameters in SVR models. The IKOA process was shown in Figure 5.

The specific steps of the IKOA method were as follows:

(1) Initialized IKOA parameters, including initial population size N, population dimension dim, maximum iteration number $T_{\max}$, cycle control parameter Tc, initial gravitational parameter μ_0, and gravitational decay factor λ_1. Initialized SVR parameters, including C and γ.

(2) Used the Sobol sequence to generate the initial population and initialized the orbital period and orbital eccentricity.

The Sobol sequence was used to generate candidate solutions (planets) within the specified ranges of C and γ. Each planet contains a set of hyperparameters C and γ, forming the initial population. Calculating the fitness of each planet and selecting the planet with the minimum fitness as the Sun, the initial position of the population can be written as

$$X_i^j = X_{i,low}^j + K_i \cdot (X_{i,up}^j - X_{i,low}^j), \tag{1}$$

where X_i^j is the position of the i-th planet in the j-th dimension; K_i is the random number generated by the Sobol sequence for the i-th planet, $K_i \in \left[0, 1\right]$; and $X_{i,up}^j$ and $X_{i,low}^j$ are the upper and lower bounds of the j-th dimension of the i-th planet, respectively.

The equations for initializing the orbital period and orbital eccentricity are as follows:

$$T_i = |r|, i = 1, \ldots, N; \tag{2}$$

$$e_i = rand_{[0,1]}, i = 1, \ldots, N, \tag{3}$$

where r is a random number generated based on a normal distribution.

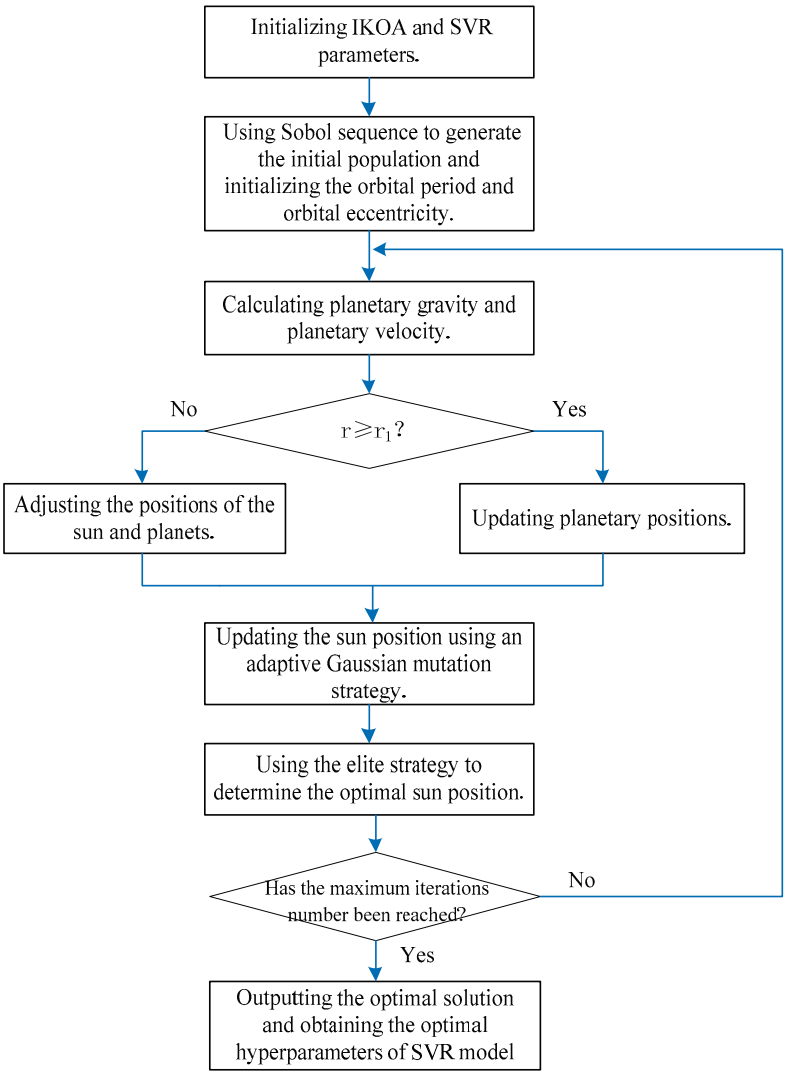


Figure 5. Flowchart of IKOA method.

Traditional KOA uses a random number generation method for the initial population positions, which introduces uncertainty in the initial positions and may fall into local optima [20,21]. In contrast, by using the Sobol sequence to generate initial solutions, the generated random numbers can be evenly distributed in the solution space. To facilitate efficient optimization by the IKOA, we linearly normalized the search ranges of the SVM hyperparameters C and γ to the interval [0, 1]. The algorithm operates within a unified two-dimensional search space [0, 1]. The population size is set to 100, the dimension is 2, and the upper and lower bounds of the population position are 1 and 0, respectively. The initial population distributions generated by the random number method and the Sobol sequence method within the normalized space are shown in Figures 6 and 7, respectively (Note: Figures 6 and 7 are for visualization purposes only and use a sample size of 100 individuals; the actual experiment employed 30 individuals). For each fitness evaluation, the dimensional values of an individual are mapped back to their actual hyperparameter values. As shown in the figures, there is a significant difference in the scatter distribution characteristics between the two methods. The scatter points generated by random number generation are typically more densely distributed in certain regions and relatively sparse in others, with obvious blank areas. In contrast, the scatter points generated by the Sobol sequence generation are more uniformly distributed across the entire space, ensuring a more comprehensive coverage of the entire search space and

avoiding large areas of sample overlap or omission, thereby enhancing the global exploration ability of the algorithm in the initial stage and laying a better foundation for finding high-quality solutions.

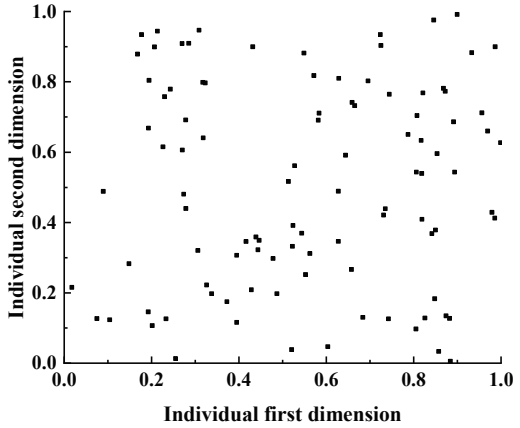


Figure 6. Scatter points generated by random number generation method.

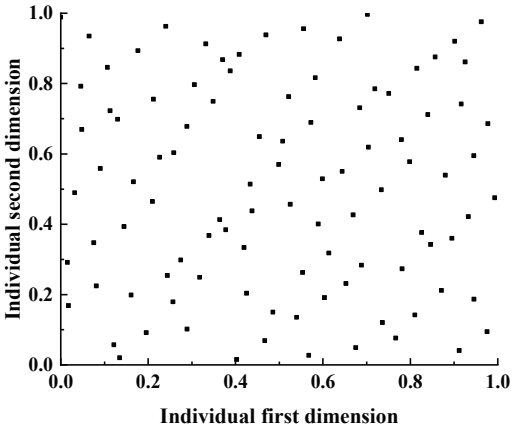


Figure 7. Scatter points generated by Sobol sequence generation method.

(3) Calculating planetary gravity and planetary velocity.

The expressions of planetary gravity are

$$F_{g_i}(t) = e_i \cdot \mu(t) \cdot \frac{\bar{M}_s \cdot \bar{m}_i}{\bar{R}_i^2(t) + \varepsilon} + r_1; \tag{4}$$

$$R_i(t) = \sqrt{\sum_{j=1}^{\dim} (X_{Sj}(t) - X_{ij}(t))^2}; \tag{5}$$

$$\mu(t) = \mu_0 \times \exp(-\lambda_1 \frac{t}{T_{\max}}), \tag{6}$$

where $F_{g_i}(t)$ is the planetary gravity of the i-th planet at the t-th iteration; F_g is the planetary gravity; e_i is the orbital eccentricity of the i-th planet; $\mu(t)$ is the gravitational parameter, which is an exponentially decreasing function of the number of iterations t, used to control search accuracy; $\bar{M}_s, \bar{m}_i$ are the normalized values of the solar mass and planetary mass, respectively; $\bar{R}_i^2(t)$ is the squared normalized Euclidean distance between the Sun and the planet at the t-th iteration; r_1 is a randomly generated number between 0 and 1; ε is a small value to prevent the denominator from becoming zero; $X_s(t), X_i(t)$ are the positions of the Sun and the i-th planet at the t-th iteration, respectively; $X_{Sj}(t)$ is the position of the Sun in

the j-th dimension at the t-th iteration; and $X_{ij}(t)$ is the position of the i-th planet in the j-th dimension at the t-th iteration.

The equations of solar mass, planetary mass, and fitness are as follows:

$$M_s = r_2 \times \frac{fit_s(t) - worst(t)}{\sum_{k=1}^{N} (fit_k(t) - worst(t))}; \tag{7}$$

$$M_s = r_2 \times \frac{fit_s(t) - worst(t)}{\sum_{k=1}^{N} (fit_k(t) - worst(t))}; \tag{8}$$

$$fit_s(t) = best(t) = \min_{k \in 1,2,\ldots,N} fit_k(t); \tag{9}$$

$$worst(t) = \max_{k \in 1,2,\ldots,N} fit_k(t), \tag{10}$$

where M_s is the solar mass; m_i is the planetary mass; r_2 is a randomly generated number between 0 and 1; $fit_i(t)$ is the fitness of the i-th planet in the t-th iteration; $fit_k(t)$ is the fitness of the k-th planet, $k \in 1, 2, \ldots, N$, introduced to avoid conflicts with $fit_i(t)$; $fit_s(t)$ is the minimum value among the fitnesses of all planets in the t-th iteration; $best(t)$ is the minimum value among the fitnesses of all planets in the t-th iteration; $worst(t)$ is the maximum values among the fitnesses of all planets in the t-th iteration.

Planetary velocity represents the step size and direction of planetary movement in the search space. $\overline{R_i(t)}$ represents the normalized distance between the Sun and the planet; when $\overline{R_i(t)} \leq 0.5$, the local exploitation formula is adopted to calculate planetary velocity; and when $\overline{R_i(t)} > 0.5$, the global exploration formula is used. The expression is as follows:

$$V_i(t) = \begin{cases} \ell \cdot (2r_4 X_i - X_b) + \ddot{\ell} \cdot (X_a - X_b) + \left(1 - \overline{R_i(t)}\right) \cdot \mathcal{F} \cdot U_1 \cdot r_5 \cdot \left(X_{i,up} - X_{i,low}\right), \overline{R_i(t)} \leq 0.5 \\ r_4 \cdot \ell \cdot (X_a - X_i) + \left(1 - \overline{R_i(t)}\right) \cdot \mathcal{F} \cdot U_2 \cdot r_5 \cdot \left(r_3 X_{i,up} - X_{i,low}\right), \overline{R_i(t)} > 0.5 \end{cases}, \tag{11}$$

where $V_i(t)$ is the planetary velocity of the i-th planet at the t-th iteration; X_a, X_b are the positions of two randomly selected planets; $X_{i,up}, X_{i,low}$ are the maximum and minimum values of the planet positions at the t-th iteration; $R_i(t)$ is the Euclidean distance between the Sun and the i-th planet at the t-th iteration; r_3, r_4, r_5 are randomly generated numbers between 0 and 1; U_1 and U_2 are variables that can be either 0 or 1; when $r_5 \leq r_4$, U_1 takes 0, and otherwise it takes 1; when $r_3 \leq r_4$, U_2 takes 0, and otherwise it takes 1; $\mathcal{F}$ is a variable that can be either 1 or -1; and when $r_4 \leq 0.5$, $\mathcal{F}$ takes 1, and otherwise takes -1. Some variables in Equation (11) can be written as

$$\ell = U \cdot (r_3 \cdot (1 - r_4) + r_4) \cdot \left[\mu(t) \cdot (M_S + m_i) \left| \frac{2}{R_i(t) + \varepsilon} - \frac{1}{a_{i(t)} + \varepsilon} \right| \right]^{\frac{1}{2}}; \tag{12}$$

$$\ddot{\ell} = \left(1 - U\right) \cdot \left(r_3 \cdot \left(1 - r_5\right) + r_5\right) \cdot \left[\mu(t) \cdot (M_S + m_i) \left| \frac{2}{R_i(t) + \varepsilon} - \frac{1}{a_{i(t)} + \varepsilon} \right| \right]^{\frac{1}{2}}; \tag{13}$$

$$\ell = \left[\mu(t) \cdot (M_S + m_i) \left| \frac{2}{R_i(t) + \varepsilon} - \frac{1}{a_{i(t)} + \varepsilon} \right| \right]^{\frac{1}{2}}; \tag{14}$$

$$a_{i(t)} = r_3 \times \left[T_i^2 \times \frac{\mu(t) \times (M_s + m_i)}{4\pi^2}\right]^{\frac{1}{3}}, \tag{15}$$

where $a_{i(t)}$ is the semi-major axis of the elliptical orbit of the i-th planet, controlling the amplitude of the planet's motion; the larger $a_{i(t)}$ is, the broader the search range is; U is a variable that takes 0 or 1; when $r_5 \leq r_6$, U takes 0, and otherwise takes 1; r_6 is a randomly

generated number between 0 and 1; and ε is a small value to prevent the denominator from becoming zero.

When $r \geq r_1$, proceed to step (4) to update the planetary positions, and when $r < r_1$, proceed to step (5) to adjust the positions of the Sun and planets.

(4) Updating planetary positions. During the simulation of planetary motion around the Sun, periodically switch the search direction to break through local optimal regions, providing planets with better opportunities to explore the entire space [23]. When $r \geq r_1$, the new planetary position is updated as follows:

$$X_i(t+1) = X_i(t) + \mathcal{F} \cdot V_i(t) + \left(F_{g_i}(t) + |r|\right) \cdot U \cdot (X_{\mathbf{S}}(t) - X_i(t)), \tag{16}$$

where $X_i(t+1)$ is the position of the i-th planet at the $t+1$ iteration; r is a randomly generated number between 0 and 1.

(5) Adjusting the positions of the Sun and planets. When $r < r_1$, update the planetary positions and simulate the changes in the distance between the Sun and planets with the number of iterations. When planets are close to the Sun, the development operator is activated to enhance convergence speed, and KOA focuses on optimizing development; when planets are far from the Sun, they adjust their distances from the Sun, and KOA focuses on optimizing the exploration operation [24]. The mathematical model expression is as follows:

$$X_i(t+1) = X_i(t) \cdot U_1 + (1 - U_1) \cdot \left(\frac{X_i(t) + X_S + X_a(t)}{3.0} + h \cdot \left(\frac{X_i(t) + X_S + X_a(t)}{3.0} - X_b(t) \right) \right), \tag{17}$$

where h is the adaptive factor used to control the distance between the Sun and the planet in the t-th iteration, expressed as

$$h = \frac{1}{e^{\eta r}}, \tag{18}$$

where η is a linear decreasing factor from 1 to -2, expressed as

$$\eta = (a_2 - 1) \times r_4 + 1, \tag{19}$$

and where a_2 is the loop control parameter that gradually decreases from -1 to -2 over $T_{\max}$ loops during the entire optimization process, expressed as

$$a_2 = -1 - \left(\frac{t \% \frac{T_{max}}{T_c}}{\frac{T_{max}}{T_c}} \right). \tag{20}$$

(6) Updating the Sun's position using an adaptive Gaussian mutation strategy.

KOA may encounter situations where the Sun's position remains unchanged after multiple iterations, and the Sun's position cannot escape the local optimum after the perturbation weakens in the later stages of exploration. This premature convergence problem is particularly prominent in multi-peak optimization problems, leading to the final optimization result deviating from the global optimum solution [25]. To address this issue, this paper introduces an adaptive Gaussian mutation strategy into the algorithm's iteration process. By dynamically adjusting the intensity and frequency of Gaussian mutations during algorithm execution, individuals can escape from local optima and regain search activity when trapped in a local optimum. Compared to fixed-parameter mutation methods, the adaptive mechanism uses the current individual's position as an adaptive adjustment factor to dynamically adjust the perturbation intensity of Gaussian mutation and flexibly regulate the mutation range. This approach maintains local search

precision while enhancing global exploration capabilities. The adaptive Gaussian mutation strategy is expressed as

$$X_s' = \begin{cases} X_s \times (1 + \mathcal{N}(0, \sigma)), & b_1 < b_2 \\ X_s, & b_1 \geq b_2 \end{cases}, \tag{21}$$

where X_s' is the Sun's position after mutation; X_s is the Sun's position before mutation; b_1 and b_2 are randomly generated numbers between 0 and 1; $\mathcal{N}(0, \sigma)$ is a normal random variable with a mean of 0 and a standard deviation of σ; σ is the standard deviation of normal mutation, which controls the mutation intensity, and its expression is as follows:

$$\sigma = e^{-2 \cdot \frac{t}{T_{\max}}}. \tag{22}$$

From the above equation, the standard deviation curve with the number of iterations can be obtained, as shown in Figure 8.

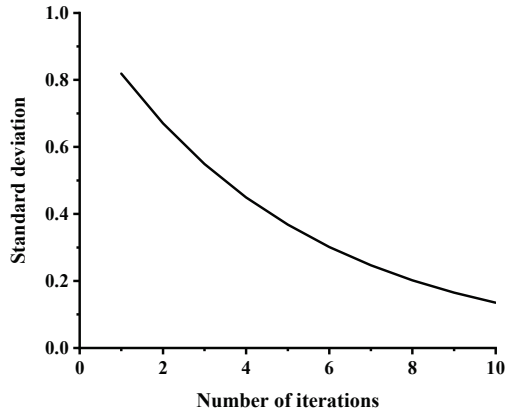


Figure 8. Standard deviation curve with the number of iterations.

(7) Using the elite strategy to determine the optimal Sun position

The elite strategy is used to determine the optimal Sun position, expressed as

$$X_{s,new}(t+1) = \begin{cases} X_s(t+1), X_s(t+1) \leq X_s(t) \\ X_s(t), \quad X_s(t+1) > X_s(t) \end{cases}, \tag{23}$$

where $X_{s,new}(t+1)$ represents the retained position of the Sun.

(8) Determining whether the maximum number of iterations has been reached. If so, the iteration ends, and the results C and γ are output. If not, return to step (3).

4.2. Building the SVR Model

SVR is a machine learning algorithm based on the principle of minimizing structural risk. Its fundamental objective is not merely to fit the training data, but to minimize the model's generalization error by seeking a regression function f(x) that is as "smooth" as possible while maintaining predictive accuracy.

The implementation of SVR relies on the configuration of three core hyperparameters: C, γ, and the insensitivity loss parameter ε. These three parameters collectively determine the model's performance: C: balances the model's structural risk against empirical risk. A larger C value forces the model to reduce training error, but this may come at the cost of sacrificing the model's generalization ability, leading to overfitting. Conversely, a smaller C value places greater emphasis on reducing model complexity, potentially resulting in

the model underfitting the training data. γ: When using a radial basis function (RBF) kernel, γ determines the spatial influence range of individual training samples. A high γ value indicates a small influence range, where only nearby data points affect predictions, potentially causing the model to focus excessively on local details and overfit. A low γ value indicates a large influence range, resulting in a smoother model but potentially failing to capture complex data variations, leading to underfitting. ε: Defines the width of the ε-insensitivity band. Loss is only computed when the prediction error of a sample point exceeds this range. Thus, ε controls the model's precision and tolerance for error. A larger ε increases the model's tolerance for error, resulting in a simpler model; a smaller ε makes the model more sensitive to error, leading to a more complex model.

Mathematically, for the training set $(x_i, y_i)_{i=1}^n$, SVR performs regression by constructing a separating hyperplane or an approximating function in the high-dimensional feature space:

$$f(x) = w^T \phi(x) + b, \tag{24}$$

where $f(x)$ is the objective function; $\phi(x)$ is a mapping function; w^T is the weight vector; and b is the bias term.

By introducing slack variables ξ_i, ξ_i^*, the optimal $f(x)$ can be found by solving the following convex quadratic programming problem:

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*). \tag{25}$$

The constraints are as follows:

$$\begin{cases} y_i - f(x_i) \le \varepsilon + \xi_i \\ f(x_i) - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases}. \tag{26}$$

By introducing the Lagrange multiplier α_i, α_i' and taking the partial derivative with respect to the parameter, the minimization problem can be transformed into the following dual problem:

$$\begin{array}{l} minR(\alpha', \alpha) = \frac{1}{2}\sum\limits_{i=1j=1}^{n}(\alpha_i' - \alpha_i)(\alpha_j' - \alpha_j)K(x_i, x_j) + \\ \varepsilon\sum\limits_{i=1}^{n}(\alpha_i' + \alpha_i) - \sum\limits_{i=1}^{n} y_i(\alpha_i' - \alpha_i) \end{array}, \tag{27}$$

where $K(x_i, x)$ is the kernel function.

The final SVR decision function is

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x_i, x) + b. \tag{28}$$

4.3. IKOA-SVR Model Training

Based on the intelligent identification training dataset, a Gaussian radial basis kernel function is used as the kernel function for the SVR model. A multi-step sliding cross-validation method is proposed to train the SVR model, which involves randomly and uniformly dividing the training set into n subsets. The training set is randomly and uniformly divided into n subsets, and two subsets are selected as the internal testing dataset in sequence (without repetition), while the remaining subsets are used as the internal training dataset. There will be C(n,2) situations. Each situation is trained using the internal training dataset and tested using the internal testing dataset, with the cross-

validation process repeated C(n,2) times to ensure that each pair of subsets is used as the internal testing dataset exactly once.

Compared to traditional K-fold cross-validation, which uses only a single data subset as the testing set in each iteration, the multi-step sliding cross-validation method selects two data subsets to form the testing set. One primary source of uncertainty in traditional K-fold cross-validation is the variance within the testing set during a single iteration. Multi-step sliding cross-validation increases the number of testing samples by using two subsets as the testing set in each iteration, effectively smoothing out randomness in single-run evaluations and thereby reducing the variance in performance assessments. Finally, averaging the results from C(n,2) exhaustive tests further ensures the high stability of the final evaluation outcomes.

Traditional K-fold cross-validation only tests a model's ability to learn from $n - 1$ subsets and generalize to a single unknown subset. In contrast, multi-step sliding cross-validation comprehensively evaluates a model's capacity to learn from $n - 2$ subsets and generalize to any combination of two unknown subsets. This approach enables the model to not only adapt to variations within a single data distribution but also overcome challenges posed by different combinations of data distributions, thereby enhancing its generalization capability against data heterogeneity.

The C(n,2) iterations far exceed the n iterations of traditional K-fold cross-validation. This approach systematically eliminates random biases arising from specific data partitioning schemes, ensuring evaluation outcomes are not dependent on any single random split. Consequently, the model assessment achieves greater comprehensiveness and thoroughness.

For each test, an intelligent identification model for valve internal leakage is constructed using Mean Squared Error (MSE) as the fitness function. The specific formula is as follows:

$$\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2, \tag{29}$$

where m is the number of samples in the training dataset; y_i is the actual value of the i-th sample in the training dataset; and $\hat{y}_i$ is the predicted value of the i-th sample in the training dataset.

To objectively and quantitatively evaluate the model performance proposed in this paper, in addition to MSE, four performance metrics are used to evaluate the model performance, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and determination coefficient (R^2). The final performance evaluation result of the model is obtained by taking the average of the performance results from C(n,2) tests. By using five performance metrics to evaluate the identification results, the identification accuracy and reliability of the model can be comprehensively measured from different perspectives.

The equations for these performance metrics are as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}; \tag{30}$$

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|; \tag{31}$$

$$\mathrm{MAPE} = \frac{100\%}{m}\sum_{i=1}^{m}\left|\frac{\hat{y}_i - y_i}{y_i}\right|; \tag{32}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (\overline{y}_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \overline{y}_i)^2}. \quad (33)$$

MSE, RMSE, and MAE are used to measure the deviation degree between the identification value and the actual value. The smaller the value, the smaller the identification error and the higher the accuracy. MAPE represents the identification error as a percentage of the actual value, which can more intuitively reflect the relative error size of the identification result. The R^2 reflects the fitness degree of the model, and the closer the R^2 is to 1, the closer the identification result is to the actual value.

5. Case Studies

The study focuses on the main steam drain pipe valves of a 600 MW thermal power station. The rated temperature of the main steam is 600 °C, and the rated pressure is 24.2 MPa. The structure diagram of the main steam drain pipe valve is shown in Figure 1. The calculation conditions are as follows: drain pipe length is 8.0 m, drain pipe diameter is 0.09 m, drain pipe wall thickness is 0.008 m, and drain pipe insulation layer thickness is 0.11 m.

The dataset is constructed with the inlet steam temperature T_{In} and inlet steam pressure P_{In}, with pipe wall temperature T_B as input parameters and the valve internal leakage quantification G_L as output parameters. Due to the complex operating conditions and wide range of parameter variations in the unit, it is difficult to establish a dataset covering various operating conditions based on the actual operating data of the unit. Therefore, this paper simulates and generates a dataset based on the calculation model for valve internal leakage established in Section 2 [28].

The model in this paper is developed based on the PyCharm 2024.1.1 integrated environment and implemented using the Python 3.8 programming language. The computer configuration for model training is 13th Gen Intel (R) Core (TM) i5-13500H (2.60 GHz) CPU and NVIDIA GeForce RTX 4050 Laptop 6 GB GPU.

To evaluate the identification performance of the proposed IKOA-SVR method, two types of test cases are designed based on different testing datasets. The first is the In-Distribution (ID) test, where the testing dataset shares the same distribution as the training data, used to validate the model's performance on data similar to the training data. The other is the Out-Of-Distribution (OOD) test, where the testing dataset differs from the training data distribution. This evaluates the model's generalization capability, robustness, and resilience to unknown scenarios, which is crucial for assessing its practicality in real industrial settings. For the OOD test dataset sources, this paper further analyzes two scenarios: noise injection and Region-Holdout.

5.1. In-Distribution (ID) Test

Based on the main steam rated parameters, the parameter ranges and variation increments for generating the simulation dataset are shown in Table 1.

Based on Table 1, a total of 5760 simulated datasets are obtained. After normalizing the data, the dataset is randomly divided into a training dataset and a testing dataset, with the training dataset accounting for 80% and the testing dataset accounting for 20%, forming the final intelligent identification training dataset and testing dataset.

The initial parameters of IKOA are set as follows: initial population size N is 30; population dimension *dim* is 2; maximum number of iterations $T_{\max}$ is 10; cycle control parameter *Tc* is 3; initial gravitational parameter μ_0 is 0.1; and gravitational decay factor λ_1

is 15. The initial parameters of SVR are set as follows: the range of C is [0.1, 1000], and the range of γ is [0.01, 1].

Table 1. Parameter Range for Simulated Dataset Generation.

No.	Parameter Name	Start Value	End Value	Step Length
1	Main Steam Temperature/°C	570	610	5
2	Main Steam Pressure/MPa	10	26	2
3	Valve Internal Leakage/kg/h	20	200	2

The IKOA method is used for self-optimization of SVR key parameters. During the optimization process, the values of parameters C and γ are continuously adjusted. The MSE on the testing dataset is used as the fitness function to find the optimal hyperparameter combination, as shown in Figure 9.

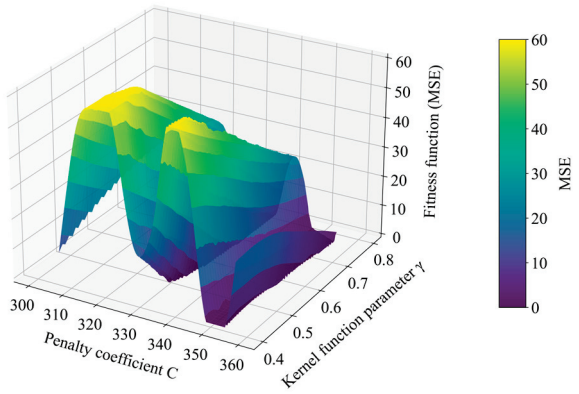


Figure 9. Optimization maps of C and γ parameter of IKOA method.

It can be observed that there are many extrema points in the optimization process. The algorithm must effectively avoid falling into local optima and achieve global optimization when solving the fitness function. During the optimization process, IKOA can repeatedly escape local optima and continue to search for better solutions, demonstrating excellent ability to escape local optimum traps. The optimal hyperparameters C and γ obtained through IKOA are 354.83 and 0.589. The optimization results indicate that IKOA effectively enhances the global search capability of the traditional KOA while maintaining stability and significantly improving the convergence performance.

To verify the superiority of parameter identification of IKOA, a comparative analysis was conducted on six algorithms: IKOA, KOA, PSO, Random Search (RS), Grid Search (GS), and Bayesian Optimization (BO). Figure 10 shows the MSE curves of four algorithms during the training process as a function of the number of iterations. According to Figure 10, IKOA tends to stabilize after four iterations, with the MSE approaching 0.59, which is the smallest value; after six iterations, KOA tends to stabilize with the MSE approaching 2.75, which is greater than IKOA; after five iterations, BO tends to stabilize with the MSE approaching 3.32, ranking third; and after six iterations, PSO tends to stabilize with the MSE approaching 5.81, which is higher than others. It can be seen that IKOA has the fastest convergence speed, the smallest identification error, and the highest identification accuracy compared to KOA, PSO, and BO.

IKOA-SVR, KOA-SVR, PSO-SVR, RS-SVR, GS-SVR, and BO-SVR methods were used for intelligent identification of valve internal leakage on the training and testing datasets, respectively. Table 2 shows the analysis results of five performance metrics in the training and testing datasets.

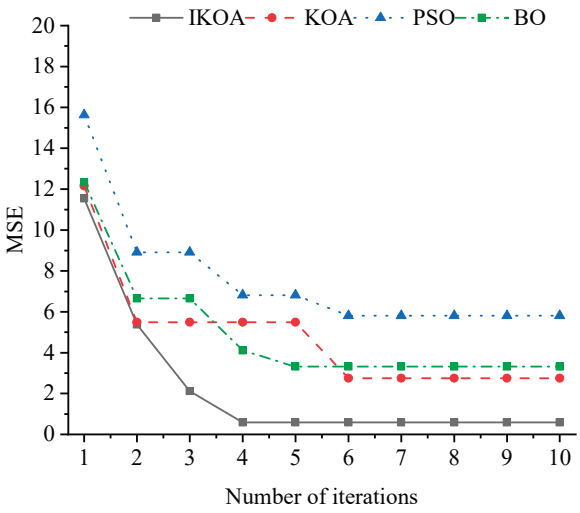


Figure 10. MSE curves of four algorithms during the training process.

Table 2. Analysis results of five performance metrics of six methods.

Method	Training Dataset						Testing Dataset				
	MSE	RMSE	R^2	MAE	MAPE	Operating Time (s)	MSE	RMSE	R^2	MAE	MAPE
IKOA-SVR	0.59	0.77	0.9998	0.45	0.0038	2794	0.65	0.81	0.9998	0.49	0.0043
KOA-SVR	2.75	1.66	0.9990	1.05	0.0093	733	3.08	1.75	0.9988	1.12	0.0096
PSO-SVR	5.81	2.41	0.9978	1.57	0.0140	623	6.30	2.51	0.9977	1.65	0.0144
RS-SVR	8.07	2.84	0.9970	1.86	0.0169	50	9.46	3.08	0.9963	2.22	0.0230
GS-SVR	3.91	1.98	0.9985	1.14	0.0097	173	4.46	2.11	0.9973	1.23	0.0108
BO-SVR	3.32	1.82	0.9987	1.15	0.0101	762	3.63	1.77	0.9984	1.16	0.0104

Table 2 shows that for the testing dataset, IKOA-SVR achieves the lowest prediction error in terms of forecasting accuracy. Its MSE is only 0.65, representing a significant reduction of approximately 79% compared to the next best model: KOA-SVR (3.08). Simultaneously, it demonstrates the best performance across all metrics, including RMSE (0.81), MAE (0.49), and MAPE (0.0043). In terms of goodness of fit, IKOA-SVR also demonstrated superior performance. Its R^2 reaches 0.9998, approaching 1, indicating that the model constructed by this algorithm explains the vast majority of variance in the target variable and can be effectively applied to new, unseen data.

Figure 11 compares the six methods on five performance metrics in the testing dataset. According to Table 2 and Figure 11, for the testing dataset, the analysis results of the five performance metrics of the six methods are similar to those of the training dataset. In terms of the difference in MSE between the training and testing datasets, it can be found that the IKOA-SVR method has the smallest difference, with a value of 0.06, indicating that no overfitting has occurred. At the same time, it should be noted that for the training dataset, the IKOA-SVR method takes 2794 s, with the longest training duration. The reason is that IKOA-SVR uses a multi-step sliding cross-validation method to train the model and searches more meticulously during the iterative optimization process.

In order to better observe the identification results, 30 samples from the testing dataset are selected for analysis. Figure 12 shows the intelligent identification results of valve internal leakage, and Figure 13 shows the scatter plot of the identification results. As shown in the figures, the identification results curve of IKOA-SVR is closest to the actual value curve, and the scatter points are also closely around the consistency line, followed by KOA-SVR, and the worst is RS-SVR. This indicates that IKOA-SVR has the highest identification accuracy, followed by KOA-SVR, and RS-SVR is the worst.

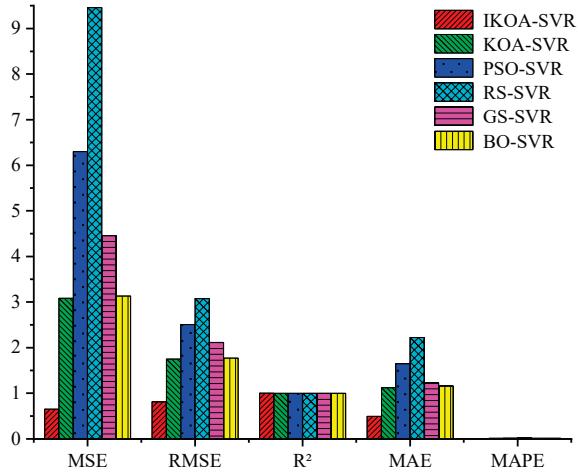


Figure 11. Comparison of 5 performance metrics with six methods in the testing dataset.

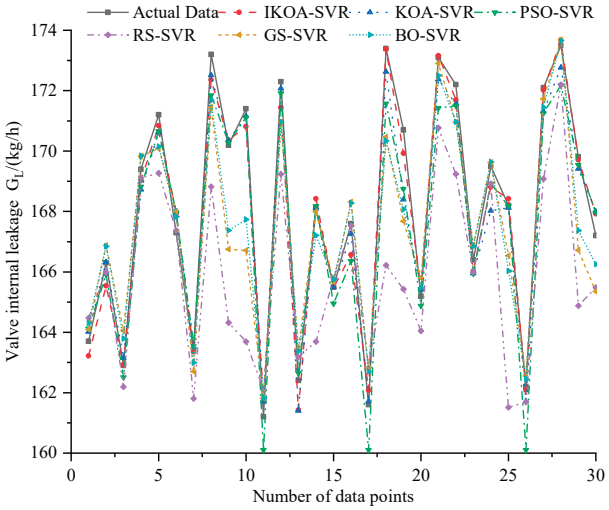


Figure 12. Intelligent identification results of valve internal leakage.

The main improvements of the IKOA in this paper are two-fold: Sobol sequence generation and adaptive Gaussian mutation strategy. To further evaluate the effectiveness of each improvement in the IKOA, an ablation study was conducted using the control variable method. The objective was to investigate the impact of individual improvements on KOA-SVR performance. The ablation study results based on the testing dataset are shown in Table 3.

As shown in Table 3, the ablation study results demonstrate the effectiveness of the IKOA across various improvement measures. Taking the MSE metric as an example, the KOA exhibits the highest MSE value of 3.08. In contrast, the MSE for the "Only Sobol sequence generation" is 2.18, outperforming KOA. This indicates that using Sobol sequences to generate a uniform initial population enhances optimization performance. The MSE of the "Only adaptive Gaussian mutation strategy" is 0.88, outperforming the previous two and approaching the 0.65 achieved by IKOA. This indicates that the adaptive Gaussian mutation strategy contributes most significantly to improving optimization performance. The ablation study results for metrics such as RMSE, MAE, MAPE, and R^2 are similar to those of MSE.

Figure 14 shows the intelligent identification results of the ablation study. As shown in the figure, the identification results curve of IKOA-SVR is closest to the actual value curve, followed by the "Only adaptive Gaussian mutation strategy", with KOA-SVR performing

the worst. "Only Sobol sequence generation" slightly outperforms KOA-SVR. This demonstrates that compared to the standalone KOA-SVR, incorporating these improvements enhances all performance metrics, with the "Only adaptive Gaussian mutation strategy" yielding the most pronounced effect.

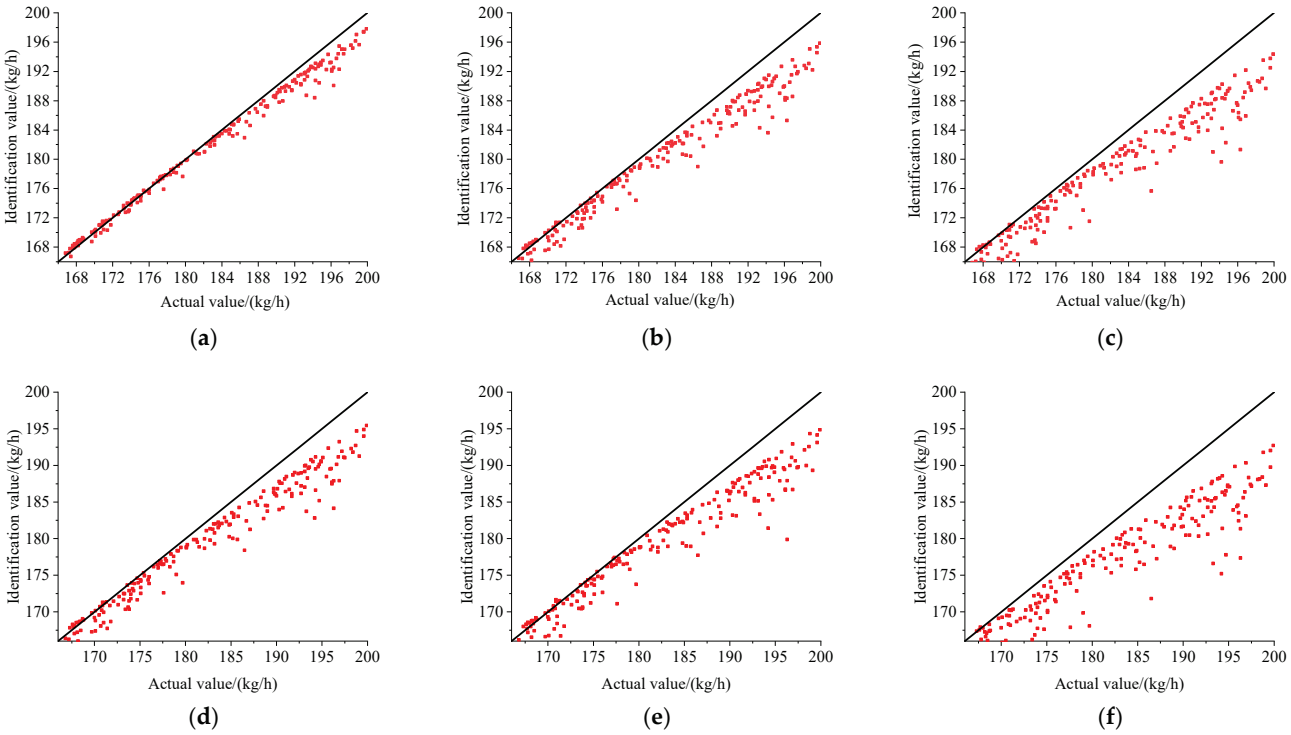


Figure 13. Scatter plots of the identification results. (**a**) IKOA-SVR. (**b**) KOA-SVR. (**c**) PSO-SVR. (**d**) RS-SVR. (**e**) GS-SVR. (**f**) BO-SVR.

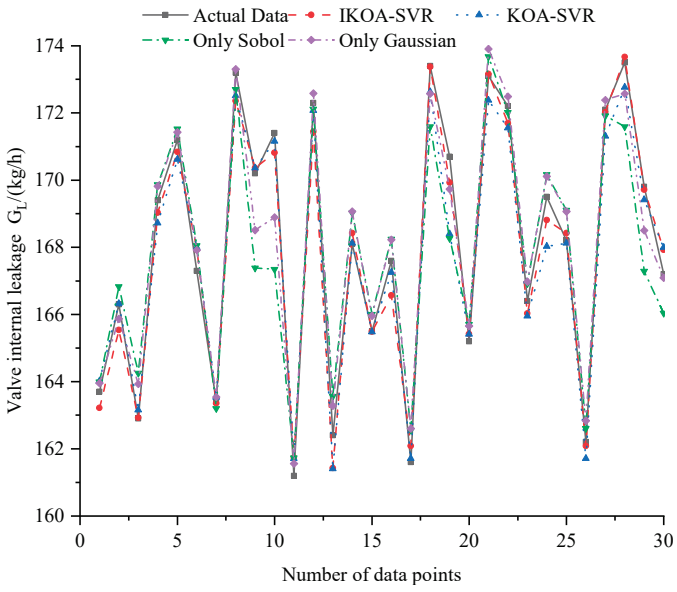


Figure 14. The intelligent identification results of ablation study.

Table 3. Ablation study results.

	MSE	RMSE	R^2	MAE	MAPE
Only Sobol sequence generation	2.18	1.48	0.9991	0.84	0.0082
Only adaptive Gaussian mutation strategy	0.88	0.94	0.9996	0.53	0.0052
IKOA-SVR	0.65	0.81	0.9998	0.49	0.0043
KOA-SVR	3.08	1.75	0.9988	1.12	0.0096

5.2. OOD Test Based on Noise Injection

By incrementally adding white noise to the testing dataset, we systematically simulate data distribution shifts ranging from minor to significant, enabling quantitative analysis of model performance degradation. The OOD test dataset construction method is as follows: Using the testing dataset from Section 5.1 as a baseline, Gaussian white noise is systematically added to each input parameter of this dataset to simulate potential data distribution shifts encountered in real-world applications, thereby constructing the OOD test dataset. To ensure noise intensity correlates with the original data scale, the amplitude of white noise is scaled relative to the standard deviation σ_j of each input parameter in the testing dataset. The white noise level $\epsilon_{i,j}$ is incremented α from 1% to 10% in 1% increments, with $\epsilon_{i,j}$ following a Gaussian distribution. The noise-augmented testing dataset x'_{test} can then be expressed as

$$x'_{\text{test},i,j} = x_{\text{test},i,j} + \epsilon_{i,j} \tag{34}$$

where $x_{\text{test},i,j}$ represents the original testing dataset, and $\epsilon_{i,j}$ denotes white noise, calculated as follows:

$$\epsilon_{i,j} \sim \mathcal{N}(0, (\alpha/100 \times \sigma_j)^2) \tag{35}$$

Based on the testing dataset with added noise, the OOD test is conducted on IKOA-SVR. The test results for various metrics are shown in Table 4. Taking the scenarios with 1%, 5%, and 10% added noise as examples, the resulting identification curves are depicted in Figure 15.

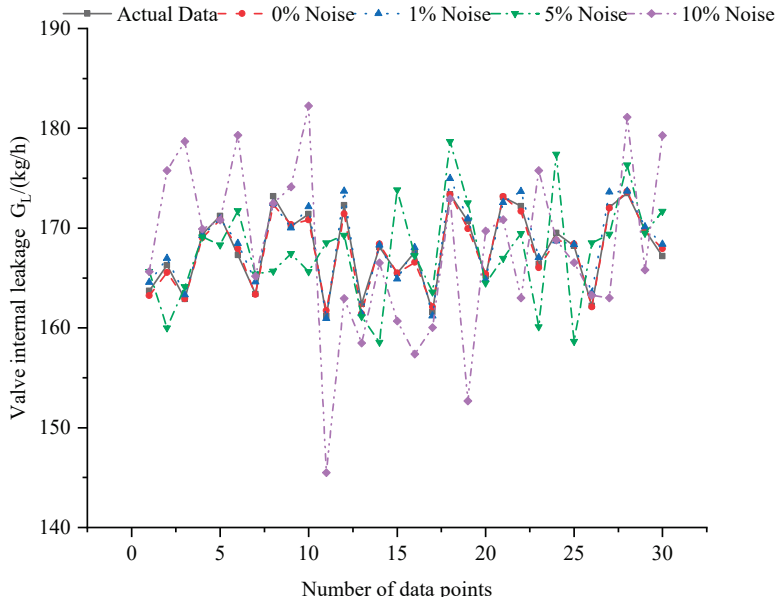


Figure 15. IKOA-SVR identification curves with different white noise.

Table 4. OOD test results for various metrics of IKOA-SVR.

White Noise Ratio	R^2	MSE	RMSE	MAE	MAPE
0%	0.9998	0.65	0.81	0.49	0.0043
1%	0.9991	1.57	1.25	0.71	0.0062
2%	0.9974	2.37	1.54	1.16	0.0084
3%	0.9943	5.17	2.27	1.25	0.0117
4%	0.9891	7.17	2.68	1.62	0.0145
5%	0.9863	10.99	3.31	2.01	0.0180
6%	0.9782	15.17	3.90	2.52	0.0214
7%	0.9734	19.89	4.46	3.09	0.0251
8%	0.9671	25.91	5.09	3.65	0.0293
9%	0.9595	30.23	5.50	4.23	0.0340
10%	0.9508	36.99	6.08	4.84	0.0391

As shown in Table 4, with increasing white noise, MSE, RMSE, MAE, and MAPE all increase to varying degrees, while R^2 continues to decrease. This indicates a decline in the robustness of IKOA-SVR. Similar observations can be made in Figure 15: as white noise increases, the identification curve deviates further from the actual values, indicating a deterioration in identification accuracy. Taking the R^2 metric as an example, at a 1% white noise level, the model's R^2 remains as high as 0.9991. This indicates that under minor distribution shifts, the model's performance degradation is negligible. Even when the noise proportion increases to 5%, the R^2 value remains robustly at 0.9863, while the MSE also remains relatively small. This demonstrates that IKOA-SVR exhibits strong robustness against mild to moderate data perturbations and possesses reliable performance and practical potential when handling noisy and fluctuating data commonly encountered in real industrial processes.

Using R^2 as the performance metric, the performance degradation curves of the six methods as white noise intensifies are shown in Figure 16. The figure reveals that all six methods exhibit identical performance trends with white noise variation: their performance curves decline as white noise intensifies, indicating continuous degradation. Comparatively, the performance curve of IKOA-SVR exhibits the gentlest decline, indicating minimal degradation, while GS-SVR shows the steepest decline, indicating the most significant degradation.

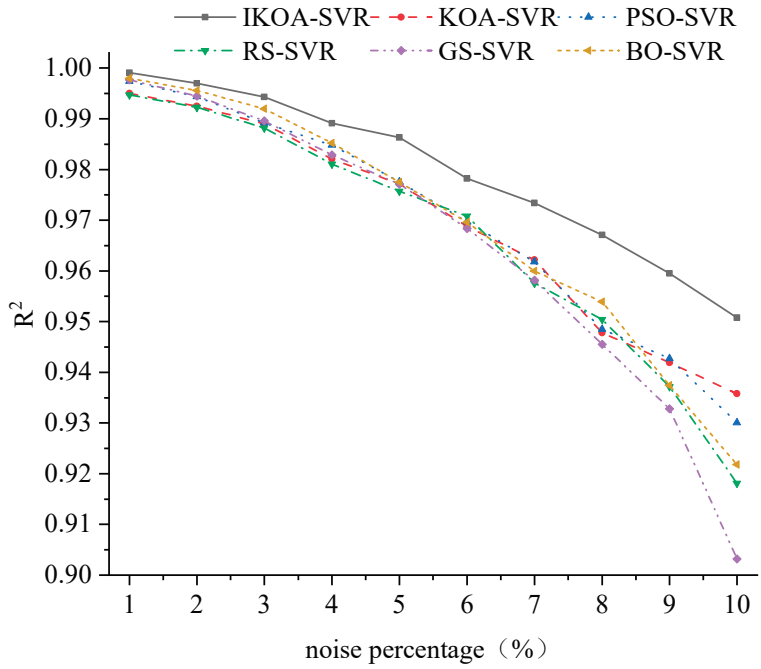


Figure 16. Performance degradation curves of six methods.

5.3. OOD Test Based on Region-Holdout

To rigorously evaluate the generalization capability and robustness of the developed model when confronted with specific input space regions not covered by the training data distribution, this paper additionally incorporates the OOD test based on Region-Holdout. When simulating the dataset, the original dataset is divided into three non-contiguous regions. The regions at both ends of the dataset are used as the training dataset, while the middle region serves as the testing dataset. This ensures the testing data distribution is clearly outside the training data distribution, thereby simulating novel operating conditions or unexplored parameter ranges that may be encountered in actual systems.

The parameter ranges and variation increments used when simulating the training dataset are shown in Table 5. The parameter ranges and variation increments used when simulating the testing dataset are shown in Table 6.

Table 5. Parameter ranges for simulated training dataset generation.

Data Segment	Parameter Name	Start Value	End Value	Step Size
Data Segment1	Main Steam Temperature/°C	570	595	3
	Main Steam Pressure/MPa	10	23	1
	Valve Internal Leakage/kg/h	30	100	2
Data Segment2	Main Steam Temperature/°C	605	610	2
	Main Steam Pressure/MPa	25	27	0.5
	Valve Internal Leakage/kg/h	140	200	2

Table 6. Parameter ranges for simulated testing dataset generation.

No.	Parameter Name	Start Value	End Value	Step Size
1	Main Steam Temperature/°C	595	605	2
2	Main Steam Pressure/MPa	23	25	0.4
3	Valve Internal Leakage/kg/h	100	140	2

The OOD test results for six methods are shown in Table 7. A radar chart of OOD test performance for six methods is shown in Figure 17.

Table 7. OOD test results of six methods.

Method	R^2	MSE	RMSE	MAE	MAPE
IKOA-SVR	0.9834	7.58	2.75	1.53	0.0137
KOA-SVR	0.9554	15.35	3.92	2.57	0.0228
PSO-SVR	0.9265	19.22	4.38	3.08	0.0263
RS-SVR	0.9656	11.31	3.36	2.01	0.0180
GS-SVR	0.9222	19.81	4.45	3.03	0.0264
BO-SVR	0.9589	14.88	3.86	2.58	0.0225

Combining Tables 2 and 7 reveals that during the OOD test, the MSE, RMSE, MAE, and MAPE metrics of all six methods increased to varying degrees, while R^2 decreased. This indicates that when testing data lies outside the distribution of training data, the generalization capability and robustness of all six methods diminish. Relatively speaking, IKOA-SVR exhibits the smallest changes across all metrics, indicating its robustness is the best among the six methods. Taking the R^2 metric as an example, IKOA-SVR still maintains a high level of 0.9834. This demonstrates that the model retains good extrapolation capability in unknown regions beyond the training data boundaries. Despite the

clear regional differences in data distribution between the testing dataset and the training dataset, IKOA-SVR still maintains high prediction accuracy and goodness-of-fit.

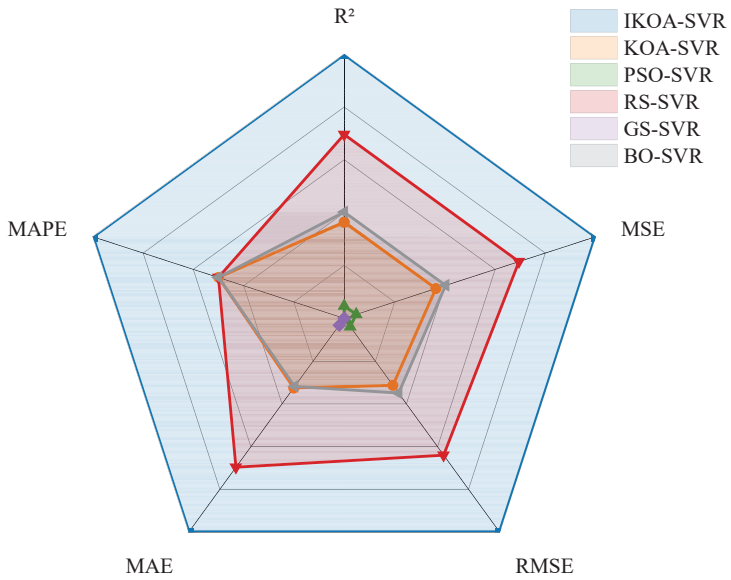


Figure 17. Radar chart of OOD test performance for six methods.

Figure 17 presents a normalized performance radar chart for six methods across five metrics. In this chart, the optimal performance for each metric is mapped to 1, while the worst performance is mapped to 0. Consequently, the area enclosed by the polygon directly reflects the overall quality of the model's performance. The diagram reveals that IKOA-SVR (blue polygon) demonstrates absolute dominance in performance. Its polygon vertices achieve the maximum value of 1.0 across all five evaluation axes, proving that it simultaneously attains the highest prediction accuracy (maximum R^2) while minimizing all types of prediction errors (MSE, RMSE, MAE, and MAPE). This highlights its exceptional and comprehensive optimization capabilities. The other five methods lag significantly behind, highlighting pronounced performance gaps. This strongly confirms that the superiority of IKOA-SVR is comprehensive and robust, rather than an isolated performance under specific evaluation metrics.

6. Conclusions

Valves are critical auxiliary equipment in thermal power stations. Accurately assessing valve internal leakage quantification is of great significance for ensuring the safe and economical operation of a thermal power station. This paper proposed an intelligent valve internal leakage identification method that integrated the IKOA method with SVR. Taking the main steam drain pipe valve as an example, the method was validated through simulation case studies. Using MSE, RMSE, MAE, MAPE, and R^2 as performance metrics, IKOA-SVR, KOA-SVR, PSO-SVR, RS-SVR, GS-SVR, and BO-SVR methods were used to compare and analyze the training and testing datasets, respectively. The main conclusions of the paper are as follows:

(1) The proposed IKOA method, which introduced Sobol sequence generation and adaptive Gaussian mutation strategy, can effectively solve the problems of local optima and overfitting commonly encountered in the KOA optimization process, thus possessing stronger global search capability.

(2) By optimizing SVR key parameters using the IKOA method and training the IKOA-SVR model with a multi-step sliding cross-validation method, the difficulty in selecting key parameters of the SVR model can be effectively addressed, thereby improving

identification accuracy. However, while improving search accuracy, it also increases the running time, resulting in the slowest model training speed.

(3) Whether it is the training dataset or the testing dataset, compared to the other five methods, the IKOA-SVR method has the smallest MSE, RMSE, MAE, and MAPE values for identification results, and the largest R^2 value, indicating that the IKOA-SVR method has the strongest generalization ability, the best effect in improving fitting ability, the smallest identification error, the highest identification accuracy, and is closer to the actual value.

(4) During the OOD test based on noise injection, the performance curves of all six methods decay as the white noise intensity increases in the testing dataset. By comparison, IKOA-SVR exhibits the least performance degradation and maintains strong robustness when confronted with mild to moderate data perturbations. During the OOD test based on Region-Holdout, both the generalization capability and robustness of all six methods decreased. In comparison, IKOA-SVR exhibits the smallest changes across all metrics, demonstrating the most optimal robustness.

This paper proposed the IKOA-SVR method for intelligent identification of valve internal leakage in a thermal power station, but there are still the following issues that need to be further improved in future research.

(1) The constructed dataset is simulated based on the calculation model of valve internal leakage established in Section 2, rather than actual operating data from the unit, and its reliability needs further verification.

(2) Many intelligent identification methods, such as GA, SOA, WOA, etc., have the potential to solve this problem. The paper only compared IKOA-SVR, KOA-SVR, PSO-SVR, RS-SVR, GS-SVR, and BO-SVR, which is not comprehensive enough. In the future, comparative analysis should be conducted with more intelligent identification methods.

(3) The IKOA-SVR method has a slow training speed and a long testing running time. To increase its practicality, the method should be further optimized while ensuring identification accuracy to reduce running time.

Author Contributions: Conceptualization, F.J., T.J., and C.H.; methodology, R.G. and X.Y.; software, X.Y. and Z.G.; validation, Z.G. and C.H.; data curation, F.J. and X.Y.; writing—original draft preparation, R.G. and X.Y.; writing—review and editing, X.Y. and Z.G.; supervision, R.G.; project administration, T.J. and Z.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Science and Technology Project of State Grid Shanxi Electric Power Company (Project Name: Online Monitoring and Maintenance Strategy Optimization System for Valve Internal Leakage in Thermal Power Plants; No.5205ww24000F).

Data Availability Statement: The original data presented in the study are openly available at https://doi.org/10.5281/zenodo.17293750.

Conflicts of Interest: Authors Fengsheng Jia, Tao Jin, and Ruizhou Guo were employed by the company Shanxi Century Central Test Electricity Science & Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

Abbreviation	Meaning
KOA	Kepler optimization algorithm
SVR	Support vector regression
SVM	Support vector machine

GA	Genetic algorithm
SOA	Seagull optimization algorithm
PSO	Particle swarm optimization
RS	Random search
GS	Grid search
BO	Bayesian optimization
MSE	Mean squared error
RMSE	Root mean squared error
R^2	Determination coefficient
MAPE	Mean absolute percentage error
MAE	Mean absolute error
Parameter	Meaning
N	Initial population size
dim	Population dimension
$T_{\max}$	Maximum iteration number
Tc	Cycle control parameter
μ_0	Initial gravitational parameter
λ_1	Gravitational decay factor
C	Penalty coefficient
γ	Kernel function parameter
X_i^j	The position of the i-th planet in the j-th dimension
K_i	The random number generated by the Sobol sequence for the i-th planet
$X_{i,up}^j$	The upper bounds of the j-th dimension of the i-th planet
$X_{i,low}^j$	The lower bounds of the j-th dimension of the i-th planet
$F_{g_i}(t)$	The planetary gravity of the i-th planet at the t-th iteration
e_i	The orbital eccentricity of the i-th planet
M_s	The solar mass
m_i	The planetary mass
$\overline{M}_s$	The normalized values of the solar mass
$\overline{m}_i$	The normalized values of the planetary mass
$r_1, r_2, r_3, r_4, r_5, r_6, b_1, b_2$	Random numbers between 0 and 1
$X_s(t)$	The positions of the Sun at the t-th iteration
$X_i(t)$	The positions of the i-th planet at the t-th iteration
$X_{Sj}(t)$	The position of the Sun in the j-th dimension at the t-th iteration
$X_{ij}(t)$	The position of the i-th planet in the j-th dimension at the t-th iteration
$fit_i(t)$	The fitness of the i-th planet
$R_i(t)$	The Euclidean distance between the Sun and the i-th planet
$\overline{R_i(t)}$	The normalized distance between the Sun and the i-th planet
$V_i(t)$	The planetary velocity of the i-th planet at the t-th iteration
X_a, X_b	The positions of two randomly selected planets
$X_{i,up}$	The maximum values of the planet positions at the t-th iteration
$X_{i,low}$	The minimum values of the planet positions at the t-th iteration
$a_{i(t)}$	The semi-major axis of the elliptical orbit of the i-th planet
U, U_1, U_2	A variable that takes 0 or 1
$\mathcal{F}$	A variable that takes 1 or -1
η	A linear decreasing factor from 1 to -2
a_2	The loop control parameter that gradually decreases from -1 to -2 over $T_{\max}$ loops during the entire optimization process
X_s'	The Sun's position after mutation
X_s	The Sun's position before mutation
y_i	The true value of the i-th sample in the training datasets
$\hat{y}_i$	The predicted value of the i-th sample in the training datasets

References

1. Huang, X.; Qu, W.Z.; Xiao, L. Identification method of internal leakage in nuclear power plants valves using convolutional block attention module. *Nucl. Eng. Des.* **2024**, *424*, 113239. [CrossRef]
2. Jin, T.; Guo, R.Z.; Jia, F.S.; Yuan, X.H.; Guo, Z.H.; He, C.B. Quantitative analysis and influencing factors research on valve internal leakage in thermal power unit. In Proceedings of the 7th Asia Energy and Electrical Engineering Symposium, Chengdu, China, 28 March 2025; pp. 748–753.
3. Liu, Y.; Li, L.P.; Liu, G.C.; Huang, Z.J. Study of the methods for quantitatively diagnosing the inner leakage flow rate of a steam trap. *J. Eng. Therm. Energy Power* **2014**, *29*, 309–314.
4. Li, W.; Wang, G.G.; Amir, H. A Survey of learning-based intelligent optimization algorithms. *Arch. Comput. Methods Eng.* **2021**, *28*, 3781–3799. [CrossRef]
5. Zhang, B.G.; Yu, H.C.; Zhao, J. Application of OvR-GA-CART decision tree algorithm in predicting steel leakage in thin slab continuous casting. *Trans. Indian Inst. Met.* **2025**, *78*, 89.
6. Giovanni, B.; Fabio, I.; Marco, C. Bearing fault detection and recognition from supply currents with decision trees. *IEEE Access* **2024**, *12*, 12760–12770.
7. Duan, X.M.; Cen, W.; He, P.D.; Zhao, S.X.; Li, Q.; Xu, S.; Geng, A.L.; Duan, Y.X. Classification Algorithm for DC Power Quality Disturbances Based on SABO-BP. *Energies* **2024**, *17*, 361. [CrossRef]
8. Li, X.; Zhou, S.J.; Wang, F.W. A CNN-BiGRU sea level height prediction model combined with bayesian optimization algorithm. *Ocean. Eng.* **2025**, *315*, 119849. [CrossRef]
9. Li, B.L.; Shao, Y.; Lian, Y.F.; Li, P.; Lei, Q. Bayesian optimization-based LSTM for short-term heating load forecasting. *Energies* **2023**, *16*, 6234. [CrossRef]
10. Chen, J.; Peng, T.; Qian, S.J. An error corrected deep Autoformer model via Bayesian optimization algorithm and secondary decomposition for photovoltaic power prediction. *Appl. Energy* **2025**, *377*, 124738. [CrossRef]
11. Jiang, Z.; Zhou, J.; Ma, Y.Z. Fault diagnosis for rolling bearing based on parameter transfer Bayesian network. *Qual. Reliab. Eng. Int.* **2022**, *38*, 4291–4308. [CrossRef]
12. Wan, X.; Cai, X.L.; Dai, L.L. Prediction of building HVAC energy consumption based on least squares support vector machines. *Energy Inform.* **2024**, *7*, 113. [CrossRef]
13. Kumar, R.; Anand, R. Bearing fault diagnosis using multiple feature selection algorithms with SVM. *Prog. Artif. Intell.* **2024**, *13*, 119–133. [CrossRef]
14. Ginzarly, R.; Moubayed, N.; Hoblos, G.; Kanj, H.; Alakkoumi, M.; Mawas, A. Assessing HMM and SVM for Condition-Based Monitoring and Fault Detection in HEV Electrical Machines. *Energies* **2025**, *18*, 3513. [CrossRef]
15. Wu, J.K.; Chen, X.H. An identification method for voltage sag in distribution systems using SVM with grey wolf algorithm. *Int. J. Power Energy Syst.* **2024**, *44*, 1–10. [CrossRef]
16. Cuentas, S.; Garcia, E.; Penabaena, R. An SVM-GA based monitoring system for pattern recognition of autocorrelated processes. *Soft Comput.* **2022**, *26*, 5159–5178. [CrossRef]
17. Duan, Y.; Gao, C.; Xu, Z.H.; Ren, S.Y.; Wu, D.H. Multi-objective optimization for the low-carbon operation of integrated energy systems based on an Improved Genetic Algorithm. *Energies* **2025**, *18*, 2283. [CrossRef]
18. Mohammad, E.; Sammen, S.; Fatemeh, P. A hybrid novel SVM model for predicting CO_2 emissions using multi objective Seagull Optimization. *Environ. Sci. Pollut. Res.* **2021**, *28*, 66171–66192.
19. Yang, Q.L.; Duan, J.H.; Bian, H. Equivalent inertia prediction for power systems with virtual inertia based on PSO-SVM. *Electr. Eng.* **2025**, *107*, 2997–3010. [CrossRef]
20. Mohamed, A.B.; Mohamed, R.; Shaimaa, A. Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion. *Knowl.-Based Syst.* **2023**, *268*, 110454.
21. Mokhtar, A.; Messaoud, B.; Mouassa, S. Optimal power flow of thermal-wind-solar power system using enhanced Kepler optimization algorithm: Case study of a large-scale practical power system. *Wind. Eng.* **2024**, *48*, 708–739.
22. Hu, G.; Gong, C.S.; Li, X.X.; Xu, Z.Q. CGKOA: An enhanced Kepler optimization algorithm for multi-domain optimization problems. *Comput. Methods Appl. Mech. Eng.* **2024**, *425*, 116964. [CrossRef]
23. Ma, H.H.; Liao, S. Multi-strategy improved Kepler Optimization Algorithm. *Commun. Comput. Inf. Sci.* **2024**, *2062*, 296–308.
24. Mohammed, H.A.; Sulaiman, Z.A.; Abdullah, M.S.; Ahmed, R.G. Enhanced Kepler optimization method for nonlinear multi-dimensional optimal power flow. *Axioms* **2024**, *13*, 419. [CrossRef]
25. Nawal, G.; Ahmed, B.; Hanaa, M. Improving the Kepler optimization algorithm with chaotic maps: Comprehensive performance evaluation and engineering applications. *Artif. Intell. Rev.* **2024**, *57*, 313. [CrossRef]
26. Li, J.C.; Masato, N.; Zhang, Y. Improved Kepler Optimization Algorithm based on mixed strategy. *Lect. Notes Comput. Sci.* **2024**, *14788*, 157–170.

27. Cai, L.; Zhao, S.J.; Meng, F.S. Adaptive K-NN metric classification based on improved Kepler optimization algorithm. *J. Supercomput.* **2025**, *81*, 66. [CrossRef]
28. Rezazadeh, N.; De Luca, A.; Perfetto, D.; Salami, M.R.; Lamanna, G. Systematic critical review of structural health monitoring under environmental and operational variability: Approaches for baseline compensation, adaptation, and reference-free techniques. *Smart Mater. Struct.* **2025**, *34*, 073001. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

computation

MDPI

Article

mDA: Evolutionary Machine Learning Algorithm for Feature Selection in Medical Domain

Ibrahim Aljarah [1,*], Abdullah Alzaqebah [2], Nailah Al-Madi [3], Ala' M. Al-Zoubi [4] and Amro Saleh [3]

[1] The Department of Artificial Intelligence, The University of Jordan, Amman 11942, Jordan
[2] Department of Computer Science, The World Islamic Sciences and Education University, Amman 11947, Jordan
[3] Department of Computer Science, King Hussein School of Computing Sciences, Princess Sumaya University for Technology, Amman 11941, Jordan
[4] Department of Data Science and Artificial Intelligence, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman 11733, Jordan
* Correspondence: i.aljarah@ju.edu.jo

Abstract: The rapid expansion of medical data, characterized by its complex high-dimensional attributes, presents numerous promising opportunities and substantial challenges in healthcare analytics. Adopting effective feature selection techniques is essential to take advantage of the potential of such data. This research presents a modified algorithm called (mDA), which is the hybrid algorithm between the Evolutionary Population Dynamics and the Dragonfly Algorithm. This method combines Evolutionary Population Dynamics's strength with the Dragonfly Algorithm's flexible capabilities, offering a robust evolutionary machine learning approach specifically designed for medical data analysis. By integrating the dynamic population modeling of Evolutionary Population Dynamics with the adaptive search techniques of Dragonfly Algorithm, the proposed mDA significantly improves accuracy, reduces the number of features, and obtains the minimum average of the fitness scores. Comparative experiments conducted on seven diverse medical datasets against other established algorithms confirm the superior performance of the proposed mDA, establishing it as a valuable approach in examining complex medical data.

Keywords: feature selection; dragonfly algorithm; optimization; medical data analytics; evolutionary algorithms; classification

1. Introduction

The swift progress in the development of medical technologies and the widespread use of electronic health records have created enormous amounts of medical data. This high-dimensional data holds the potential to transform healthcare by facilitating more precise diagnoses, customized treatments, and predictive analytics. Nonetheless, this data's vast volume and complexity pose substantial challenges, particularly in extracting meaningful insights. A major challenge is the existence of irrelevant or redundant features that can hinder the effectiveness of machine learning models, causing overfitting and diminished generalization ability.

In recent decades, metaheuristic and evolutionary algorithms have proven to be highly effective in solving a variety of optimization problems [1–5]. The Dragonfly Algorithm (DA), a contemporary metaheuristic inspired by the behavior of dragonflies [6], stands out as a recently successful algorithm capable of surpassing other well-established optimizers in the literature. It has been employed in diverse real-world applications, including economic emission dispatch in power systems [7,8], simulation building [9], wireless

node localization in computer networks [10], and machine learning [11,12]. The DA has demonstrated excellent performance across numerous continuous, discrete, single-objective, and multi-objective optimization problems, outperforming several state-of-the-art metaheuristic and evolutionary algorithms such as Particle Swarm Optimization (PSO) and Differential Evolution (DE). Recent studies have also highlighted the growing role of intelligent and language-aware AI systems in healthcare and social media analytics, demonstrating effective applications of machine learning and natural language processing in Arabic and multilingual contexts [13–15].

Recently, the authors in [6] presented a binary version of the DA, known as BDA, which utilizes a transfer function (TF) to transform a continuous search space into a discrete one. An initial evaluation of BDA's effectiveness was performed on various feature selection challenges, with the findings indicating the method's satisfactory performance [16].

The persuasive advantages of the EPD operator prompted us to incorporate it into the newly developed Dragonfly Algorithm (DA) to evaluate its efficacy on FS problems. In this research, we adapted the DA by choosing the top three solutions along with one that is randomly generated to reposition a solution from the lower half of the population. This strategy allows solutions with lower fitness to influence the population's structure. Comprehensive results and extensive comparisons indicate that the EPD significantly boosts the DA's performance, enhancing the proposed method's ability to surpass other optimizers and achieve superior solutions with better convergence characteristics. This study presents an EPD-enhanced DA-based optimizer aimed at improving the basic DA's performance on FS tasks. Our main contributions in this research include:

- The notable advantages of the EPD operator encouraged us to utilize it with the recently introduced DA and examine its efficiency in FS problems in the medical domain.
- In the suggested method, a solution from the worst 50% of the population is repositioned by choosing one of the top three solutions and a randomly generated solution.
- The suggested method has been evaluated on seven medical datasets, each with unique configurations and attributes, to illustrate its effectiveness, solution quality, and efficiency in feature selection tasks.
- The EPD operator is combined with the modified DA (mDA) for the first time to address feature selection problems.

The paper is structured as follows: Section 2 covers related work. Section 3 highlighted the basics of DA, binary DA, and the EPD operator. Section 4 outlines the proposed methodology. Section 5 presents the results. Section 6 highlighted the clinical discussion of the findings. Lastly, Section 7 provides the conclusions and suggests avenues for future research.

2. Related Works

In our literature review, we employed a systematic research approach to locate and evaluate pertinent studies. We conducted searches in key scientific databases such as Elsevier, IEEE, Springer, and MDPI, utilizing specific keywords like metaheuristics, evolutionary computation, nature-inspired approaches, hybrid approaches, local search, Evolutionary Population Dynamics (EPD), and the Dragonfly Algorithm (DA). Our selection criteria were aimed at studies that proposed, examined, or implemented hybrid metaheuristic algorithms, with a special focus on methods integrating EPD and DA or similar evolutionary strategies. This approach guaranteed a thorough and focused review of existing literature, enabling us to evaluate the theoretical and practical contributions of current methods and to distinctly define the innovation of the proposed mDA algorithm.

Numerous studies have attempted to apply DA or enhance its effectiveness in addressing practical challenges such as photovoltaic systems [17], prolonging RFID network lifespan [18], 0-1 knapsack problems [19], and the economic emission dispatch problem [7]. In 2017, the researchers in [20] introduced a memory-based hybrid DA combined with PSO principles for global optimization problems. Moreover, the authors in [21] developed a modified DA with elite opposition learning for global optimization.

DA is applied within the healthcare care domain for the purpose of feature selection. The research in [22] aimed to categorize breast cancer tumors as either benign or malignant. Implementing the Dragonfly algorithm allows a curated selection of features to be identified, thereby augmenting the precision of classification models. The algorithm enhances the feature selection methodology by systematically identifying the most salient features and concurrently discarding redundancies. This methodological framework enhances diagnostic accuracy in the medical field, particularly in differentiating among various classifications of breast cancer tumors. DA was employed in [23] for the purpose of feature selection within the domain of machine learning; the investigation did apply this algorithm to a dataset pertaining to the classification of chronic kidney disease, thereby demonstrating considerable enhancements in classification precision. Although the primary emphasis was placed on improving classification results, the efficacy of the dragonfly algorithm in feature selection could yield advantages in medical contexts for applications such as disease diagnosis or prognosis.

Subsequent investigations could delve into the algorithm's potential in analyzing medical data to advance predictive models within healthcare environments. In [24], DA was utilized in medical image registration within the context of this investigation. It was evaluated against alternative bio-inspired algorithms, including particle swarm optimization and artificial bee colony methods. The outcomes of the simulations demonstrated that the dragonfly algorithm yielded superior quality in image registration results, though with a prolonged convergence time. This contradiction between the quality of registration and the computational duration is of principal importance when selecting an algorithm for medical applications, such as the monitoring of tumor progression. Consequently, DA has shown significant potential in the domain of medical image registration tasks, providing high-quality outcomes despite the associated increase in computational time. In the segmentation of thermographic images for early diagnosis of breast diseases, ref. [25] mimicking the swarming behaviors of dragonflies, the algorithm balances exploration and exploitation phases to compute optimal thresholds for image segmentation aims to provide a reliable method for clinicians to analyze thermography images effectively, assisting in the early detection of breast cancer.

Building upon these earlier medical applications of the DA algorithm, recent literature has demonstrated a clear transition toward hybrid evolutionary–deep learning paradigms for FS and transformer-based architectures with intrinsic interpretability mechanisms. A comprehensive literature survey [26] distills the current status of evolutionary feature selection, focusing on integration approaches for attention, adaptive population modeling, and multi-objective optimization. These approaches have a significant impact on structuring the proposed mDA algorithm.

Similarly, parallel literature on deep attention networks and vision transformer models [27,28] has been observed for integration purposes in healthcare data, including imaging and electronic health records. These models utilize attention pooling, hierarchical fusion, and representation mechanisms that inherently produce salient features that meet the criteria for feature selection. The interpretability naturally incorporated by these transformer models makes it easier to validate them for clinical use.

Methodologically, recent breakthroughs have been made in domain-aware transformer frameworks [29] that incorporate priors informed by physics and biology into their optimization approaches. These models provide enhanced semantic coherence and can be conceptualized alongside traditional wrapper approaches to FS that rely on domain information to inform and guide the process of identifying salient features. This continues to strengthen the role of jointly evolving search approaches with knowledge-informed architectures for robustness and interpretability.

From a visualization and explainability perspective, recent work continues to proliferate in post-hoc attribution methods such as SHAP and LIME [30] to effectively integrate these attribution approaches with FS algorithms, guaranteeing clinical plausibility and interpretability for these models. Empirical analyses [31] have verified that clinician-centered explanations can outperform standard SHAP explanations in both clinician trust and diagnostic accuracy, underscoring the need for interpretability and user-centered design in medical AI. Complementary analyses have concurrently argued for the use of attention maps, gradient attribution, and perturbation-based validations to ensure the relevance of selected features in clinical decision-support systems.

Taken together, these recent developments represent a clear paradigm shift toward interpretable, domain-grounded, and hybridized feature selection frameworks. They have established a coherent research stream integrating evolutionary search, deep representational learning, and explainable AI, a trajectory that is directly reflected in the conception and design of our proposed mDA model.

3. Preliminaries

3.1. Dragonfly Algorithm (DA)

The Dragonfly Algorithm is a newly introduced swarm intelligence algorithm [6]. This algorithm imitates the predation and migration behaviors of conceptualized dragonflies. The predation behavior, termed a static swarm (feeding), involves dragonflies flying in small clusters within a confined area to locate food sources. On the other hand, the migration behavior, called dynamic swarm (migratory), involves dragonflies flying in larger groups in a single direction to facilitate the swarm's migration.

Like other algorithms inspired by nature, DA operates in two phases: the exploration phase, which is based on static swarming behavior, and the exploitation phase, which is based on dynamic swarming behavior.

Five individual behaviors are used to simulate the swarming actions of dragonflies. In the equations below, X denotes the current search agent's position, X_j indicates the j-th neighbor of the X search agent, and N represents the size of the neighborhood [32]:

- *Separation* is a strategy used by a search agent to maintain distance from other nearby search agents. This action is represented mathematically as Equation (1):

$$S_i = -\sum_{j=1}^{N} X - X_i \tag{1}$$

- *Alignment* describes how a single entity adjusts its speed to align with the speeds of other nearby entities. This action is mathematically represented by Equation (2):

$$A_i = \frac{\sum_{j=1}^{N} V_j}{N} \tag{2}$$

where V_j denotes the speed of the j-th neighboring entity.

- *Cohesion* indicates the propensity of individuals to move towards the nearby center of mass, as represented mathematically by Equation (3):

$$C_i = \frac{\sum_{j=1}^{N} x_j}{N} - X \tag{3}$$

- *Attraction* describes the inclination of individuals to move toward the food source. The mathematical representation of the attraction between the food source and the ith solution is given by Equation (4):

$$F_i = F_{loc} - X \tag{4}$$

where F_{loc} denotes the location of the food source.

- *Distraction* This describes the natural inclination of individuals to flee from a threat. The separation between the adversary and the ith solution is represented mathematically by Equation (5):

$$E_i = E_{loc} + X \tag{5}$$

where E_{loc} represents the enemy's position.

Within the DA, the fitness and position of the food source are intended to be revised using the top-performing candidate (search agent) up to that point. Moreover, the fitness and position of the adversary should be adjusted based on the least successful candidate. This leads to a convergence toward favorable regions and a divergence from less favorable areas within the search space.

According to the PSO algorithm framework, the DA updates a dragonfly's position using two vectors: the step vector (ΔX), akin to the velocity vector in PSO, and the position vector. The step vector indicates the direction in which the dragonflies move. The step vector is modeled as Equation (6):

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + wX_t \tag{6}$$

In this context, s, w, a, c, f, and e denote the weights for the separation S_i, alignment A_i, cohesion C_i, attraction towards the food source F_i, and distraction from the enemy E_i for each i-th individual, respectively. These weights allow the DA to exhibit varying degrees of exploration and intensification during the optimization process. A comprehensive study of how these parameters affect the DA and their specific values is available in [6].

The position of an individual is updated as in Equation (7):

$$X_{t+1} = X_t + \Delta X_{t+1} \tag{7}$$

where t is the current iteration.

The pseudocode for the DA is presented in Algorithm 1. The process begins with generating a random initial population, where the dragonflies' positions and step vectors are arbitrarily assigned. During each iteration, the algorithm performs the following actions repeatedly until a stopping condition is met. Firstly, each population member is assessed via a fitness function. Secondly, the primary coefficients are revised. Thirdly, using Equations (1)–(5), the separation (S), alignment (A), cohesion (C), food source (F), and enemy (E) are updated. Lastly, the step vectors and positions are adjusted based on Equations (6) and (7), correspondingly.

Finally, the best solution found so far is returned.

Algorithm 1 Pseudocode of the DA

Initialize the population $X_i (i = 1, 2, \ldots, n)$
Set $\Delta X_i (i = 1, 2, \ldots, n)$
while (termination condition is not met) **do**
  Assess each dragonfly
  Update (F) and (E)
  Modify the primary coefficients $(i., e., w, s, a, c, f, and\ e)$
  Compute S, A, C, F, and E by applying Equations (1)–(5)
  Revise step vectors (ΔX_{t+1}) via Equation (6)
  Revise X_{t+1} using Equation (7)
Return the best solution

3.2. Binary Dragonfly Algorithm (BDA)

In binary optimization problems, the search space is modeled as a hypercube, and an individual's position can be altered by flipping one or more bits in its position vector $x = \{x_1, x_2, \ldots, x_d\}$. Since the original DA was intended for continuous optimization problems, it updates the individual's position by adding the step vector to the current position vector. However, this method does not apply to binary optimization problems like feature selection. As indicated in a previous study [33], employing a transfer function is an efficient and practical method to adapt a continuous algorithm to a binary context. This paper utilizes both S-shaped and V-shaped transfer functions.

In general, transfer functions are utilized to compute the likelihood of altering the elements of a position to either 0 or 1, depending on the value of the step vector (velocity) of the search agent indexed by i in the dimension indexed by d during the current iteration (t) as an input parameter. In earlier research [6], the transfer function of Equation (8) was employed to determine the likelihood of converting continuous positions to binary.

$$T(v_d^i(t)) = |(v_d^i(t))/\sqrt{1 + (v_d^i(t))^2}| \tag{8}$$

The result $T(v_k^i(t))$, obtained from Equation (8) is subsequently employed to transform the i-th component of the position vector into either 0 or 1 as per Equation (9)

$$X(t+1) = \begin{cases} \neg X_t & r < T(v_k^i(t)) \\ X_t & r \geq T(v_k^i(t)) \end{cases} \tag{9}$$

where r is a random number in the [0,1] interval.

The step vector represents the current individual's dynamism and determines the movement's extent. A smaller step vector value implies that the individual is nearing the optimal solution and should make minor adjustments (exploitation). Conversely, a higher step vector value indicates the search agent is distant from the optimal solution and necessitates significant changes (exploration) [34]. In a binary algorithm utilizing the step vector to determine the probability of position changes, transfer functions immensely influence the equilibrium between exploration and exploitation. The probability calculation will stay constant throughout the optimization process if the transfer function remains unchanged. Modifying the transfer function can enhance the influence of the step vector on position alterations for both exploration and exploitation of the search space.

3.3. Evolutionary Population Dynamics (EPD)

Evolutionary algorithms (EAs) are stochastic search methods where a group of potential solutions (population) is initialized and then gradually enhanced to meet the specified objectives better. Certain EAs employ mutation mechanisms to modify selected solutions,

while others use crossover operators. These operators aim to evolve the chosen solutions, which are often the most optimal ones. Evolutionary Population Dynamics (EPD) involves eliminating the least optimal solutions in a population and repositioning them near the best ones. This approach is fundamentally based on the theory of self-organized criticality (SOC) [35], which suggests that a local alteration in the population can influence the entire population, leading to delicate balances without an external organizing force [36]. Genetic algorithms merge the best solutions using evolutionary operators such as crossover and mutation. Conversely, EPD excludes the least optimal solutions from the current population. Two metaheuristic methods inspired by the SOC concept are evolutionary programming using self-organizing criticality (EPSCO) [37] and extremal optimization (EO) [38]. EPD is a straightforward and effective mechanism that can be integrated into various optimizers. It begins by removing the least optimal solutions from the swarm and subsequently repositioning these removed solutions around the top search agents.

4. Methodology

Feature selection is presented as a binary optimization challenge, limiting solutions to binary outcomes. Therefore, the binary version of the DA can be utilized to tackle this challenge. In this research, a vector consisting of zeroes and ones represents a solution to a FS problem, with a zero indicating that the associated feature is not selected and a one indicating that the feature is selected. The length of the solution vector corresponds to the number of features present in the original dataset. This study introduces eight wrapper feature selection techniques that leverage the BDA. Each technique uses a transfer function to convert a continuous value into a binary form. The KNN classifier [39] is employed to evaluate the selected feature subsets. The fitness function takes into account both classification accuracy and the number of features selected, in line with the understanding that feature selection is a multi-objective task. The objective function is shown in Equation (10):

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|C|}{|N|} \tag{10}$$

where $\gamma_R(D)$ denotes the classification error rate, $|C|$ signifies the count of selected features, and $|N|$ represents the total number of features in the initial dataset. The parameters α and β correspond to the significance of classification accuracy and subset length, respectively. α ranges within the interval [0,1], and β is defined as $(1-\alpha)$, adapted from [40].

4.1. Applying the EPD Strategy to BDA

As previously discussed, the EPD approach discards the least efficient solutions from the population and replaces them by creating new solutions in the vicinity of the more effective ones. This EPD strategy serves as a simple yet efficient operator for methods based on populations [36], and hence it is incorporated into the traditional DA as it is also a stochastic population-based optimizer. EPD improves the exploitation capability of BDA by removing the poorest solutions from the group and creating new nearby solutions around the superior ones.

To incorporate the EPD technique within the BDA algorithm, the swarm of dragonflies is split into two groups after sorting by their fitness scores. The less fit group is removed and reinitialized using four different strategies derived from the better half of the population.

In this study, we integrated the EPD scheme with the binary DA. The hybridization model utilizes a random selection operator. Specifically, one of the top three dragonflies in the population is chosen along with a randomly selected dragonfly. Subsequently, the 'poor' solution's leader is chosen randomly. To execute this concept, a random selection

mechanism is used to pick the solutions. Additionally, this method incorporates a basic mutation operator.

In this method, the top three individuals are chosen, and a fourth solution is created randomly. Each of the worst half solutions is repositioned around one of these four solutions based on a generated random number. The procedure is simple: a random number X_r is generated in each iteration, and one of four choices is applied to reposition the suboptimal solution: if $X_r \in [0, 0.25]$, the best solution is used; if $X_r \in [0.25, 0.5]$, the second-best solution is used; if $X_r \in [0.5, 0.75]$, the third-best solution is chosen; and if $X_r \in [0.75, 1]$, a random solution is used.

The selected solution will be used as a starting point to reposition the poor solution. Repositioning the poor solutions around the best solutions aims to heighten the median of the swarm in each step. However, this process may cause a premature convergence of the algorithm. As a remedy, a randomly generated solution is used in the first rule to promote exploration and prevent trapping in local optima.

The overall pseudo code of the mDA algorithm is described in Algorithm 2.

Algorithm 2 Pseudocode of the mDA

Initialize the population $X_i (i = 1, 2, \ldots, n)$
Initialize $\Delta X_i (i = 1, 2, \ldots, n)$
while (end condition is not satisfied) **do**
  Evaluate each dragonfly
  Update (F) and (E)
  Update the main coefficients $(i., e., w, s, a, c, f, and\ e)$
  Calculate S, A, C, F, and E (using Equations (1)–(5))
  Update step vectors (ΔX_{t+1}) using Equations (8) and (9)
  for $i = (n/2) + 1$ to n **do**
    Update the position of *i-th* dragonfly using EPD approach
Return the best solution using Equation (10)

It should be noted that the computational complexity of the proposed modified Differential Algorithm (mDA) is not substantially different from that of the original Differential Algorithm (DA). The DA has a computational complexity of $O(t \times d \times n^2)$, where t represents the number of iterations, d stands for the number of variables, and n denotes the number of solutions. The introduction of binary operators does not alter this complexity, as they are incorporated into the original DA's position updating method. However, to reinitialize 50% of the solutions, an additional complexity of $O(n/2)$ is introduced, making the overall computational complexity of the proposed mDA $O(t \times d \times n^2 + n/2)$. It is important to note that because half of the solutions need to be re-evaluated for their objective value, the mDA requires $n/2$ more function evaluations than the DA.

4.2. Experiments

We conducted our experiments using seven publicly accessible medical datasets: Breastcancer, BreastEW, Colon, HeartEW, Leukemia, Lymphography, and PenglungEW. The comparative algorithms included Traditional PSO, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Salp Swarm Algorithm (SSA), Grasshopper Algorithm (GOA), and Harris Hawks Optimization (HHO). The evaluation metrics comprised Accuracy, mean number of selected features, and mean best fitness.

4.3. Dataset Description

We assess the performance of the proposed nDA model compared to the binary DA version and other methods using seven renowned medical datasets obtained from the UCI benchmark repository [41,42].

Table 1 summarizes the datasets employed in our experiments. It enumerates seven unique datasets, each selected for its significance to particular medical and biological issues. The table describes the number of features, cases, and categories for each dataset, which are essential measures for assessing the data's complexity and breadth. Detailed information of each dataset is given below:

- Breastcancer: This dataset contains 9 features with 699 instances, divided into 2 classes, making it suitable for binary classification tasks related to breast cancer detection.
- BreastEW: Comprising 30 features and 569 instances, this dataset also targets breast cancer but with a different feature set, reflecting varied experimental conditions or data collection methodologies.
- Colon: A relatively small dataset with only 62 instances but a high dimensionality of 2000 features, primarily used for studying gene expression profiles in colon cancer.
- HeartEW: Contains 270 instances and 13 features used in analyzing heart disease with two outcome classes.
- Leukemia: The most feature-rich dataset in the collection, with 7129 features across 72 instances, indicative of high-throughput genetic profiling in leukemia studies.
- Lymphography: Consists of 148 instances and 18 features, used in diagnosing lymph node tumors with binary class outcomes.
- PenglungEW: Distinct from the others, this dataset has 73 instances and 325 features but expands the classification challenge to 7 classes, possibly indicating different stages or types of lung diseases.

Table 1. List of the Datasets Used in the Experiments.

No.	Dataset	No. of Features	No. of Instances	No. of Classes
1.	Breastcancer	9	699	2
2.	BreastEW	30	569	2
3.	Colon	2000	62	2
4.	HeartEW	13	270	2
5.	Leukemia	7129	72	2
6.	Lymphography	18	148	2
7.	PenglungEW	325	73	7

4.4. Evaluation and Experimental Settings

The suggested approach is implemented using the Matlab R2019a tool on an Intel(R) Core i7 processor running at 2.00 GHz with 16 GB of RAM. The suggested and contrasted alternatives are implemented using the same platform and programming language to provide fair comparisons.

In this work, the performance of the algorithms on the findings obtained was validated using the cross-validation method, which randomly splits each data set into 80% training and 20% testing parts [43,44]. Furthermore, each technique was run 20 times to ensure robustness and reliability of the results.

The classification accuracy in Equation (11) is used to evaluate the proposed approach, an important measure to evaluate the classification problems; more accuracy means a better solution. On the other hand, the FS algorithms aim to reduce the dimensionality of the dataset by selecting the minimum number of features concerning the classification accuracy. The accuracy and the number of the selected features are included as objectives of the fitness function; therefore, the objective is to minimize the number of features and increase

the accuracy of the classification. So, the accuracy is converted to a minimization problem by taking the error rate instead of the accuracy (1-accuracy), as shown in Equation (12).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

$$Fitness = \alpha \times (ErrorRate) + \beta \times \frac{|R|}{|N|} \tag{12}$$

where α and β are parameters between 0 and 1 to represent the importance weight of each objective ($\beta = 1 - \alpha$), *ErrorRate* indicates the classification error rate, R represents the number of selected features and the total number of features is denoted as N, based on the literature [43–47]; α is set to 0.99 and β equal to 0.01.

A sensitivity analysis was utilized to choose the experiment settings properly. The population size was set up in the analysis using various alternative values: 10, 20, 30, 50, and 100 search agents. While the maximum number of iterations was tested on three values: 50, 100, and 150. As clearly shown in Table 2, the population size with a value equal to 30 performed better. In addition, the maximum number of iterations is set to 100, based on the sensitivity results and according to previous studies [45–47]. On the other hand, the KNN is the most often used classifier with the different datasets available in the UCI repository. And according to [43,44], the value of k is set to 5.

Table 2. A sensitivity analysis of the classification accuracy for mDA algorithm with various numbers of iterations and population size.

		Population Size				
Benchmark	**Iterations**	**10**	**20**	**30**	**50**	**100**
Breastcancer	50	0.9594	0.9611	0.9581	0.9571	0.9606
	100	0.9588	0.9576	**0.9678**	0.9597	0.9581
	150	0.9611	0.9579	0.9573	0.9603	0.9606
BreastEW	50	0.9298	0.9322	0.9293	0.9301	0.9287
	100	0.9288	0.9368	**0.9382**	0.9354	0.9307
	150	0.9356	0.9344	0.9353	0.9362	0.9337
Colon	50	0.6265	0.6306	0.6143	0.6643	0.6143
	100	0.6163	0.6235	**0.7174**	0.6143	0.5847
	150	0.6204	0.5959	0.6357	0.6357	0.6051
HeartEW	50	0.7921	0.7907	0.7979	0.7900	0.7894
	100	0.8021	0.7968	**0.8227**	0.8079	0.7898
	150	0.7852	0.7921	0.7984	0.7813	0.7928
Leukemia	50	0.7553	0.8009	0.7491	0.7684	0.7719
	100	0.7728	0.7640	**0.8211**	0.7386	0.7447
	150	0.7307	0.7763	0.7149	0.7526	0.7763
Lymphography	50	0.7389	0.7369	0.7477	0.7406	0.7466
	100	0.7243	0.7390	**0.7627**	0.7513	0.7411
	150	0.7390	0.7165	0.7311	0.7245	0.7441
PenglungEW	50	0.5560	0.5138	0.5379	0.5319	0.4888
	100	0.4966	0.5310	**0.6035**	0.4966	0.5241
	150	0.5284	0.5724	0.4845	0.5500	0.5345

The sensitivity analysis of the classification accuracy for the proposed alogrithm (mDA), showing how it performs across a variety of medical datasets with differing numbers of iterations and population sizes. The results reveal that the Breastcancer dataset consistently exhibits high classification accuracy, which marginally improves as the number

of iterations increases, indicating the algorithm's stability across different population sizes. In contrast, the Colon dataset shows notable variability in accuracy, especially at larger population sizes, which might suggest a sensitivity to the dimensionality of the feature space or a propensity for overfitting. The Leukemia and HeartEW datasets display a positive trend in accuracy with increasing iterations, underscoring that more iterations aid in achieving better generalization of the model. However, the PenglungEW dataset, which is categorized into multiple classes, tends to have lower overall accuracy, pointing to the inherent challenges somehow of multi-class classification. This analysis provides essential insights into the robustness and efficiency of the mDA algorithm, highlighting its potential advantages and limitations when applied to different types of medical datasets.

Table 3 enumerates the parameters utilized in the experiments outlined in the study, providing a clear framework for the setup and execution of the classification models tested. The population size for the experiments was set at 30 based on the sensitivity analysis experiment, with a maximum of 100 iterations per run to allow the algorithms sufficient opportunity to converge toward optimal solutions. The K-nearest neighbors algorithm (KNN) used a value of K equal to 5, optimizing the balance between bias and variance in the classification. Two parameters, α, and β, were set at 0.99 and 0.01 respectively in the fitness function, indicating a strong preference for one aspect of the model's performance over another, likely focusing on maximizing accuracy while controlling for overfitting or complexity in the model. This structured approach in parameter selection underscores the methodological rigor of the experiments, aiming to achieve both precise and generalizable outcomes.

Table 3. List of the used Parameters in the Experiments.

No.	Parameter	Value
1.	Training Data	80%
2.	Testing Data	20%
3.	Population Size	30
4.	Max Number of iterations	100
5.	K (Knn)	5
6.	Number of runs for each technique	20
7.	α in fitness function	0.99
8.	β in fitness function	0.01

5. Results

Table 4 provides a comparative analysis of classification accuracy between the proposed algorithm (mDA) and other well-known optimization algorithms across various medical datasets. The table includes algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Salp Swarm Algorithm (SSA), Grasshopper Optimisation Algorithm (GOA), Harris Hawks Optimization (HHO), and standard (DA).

The mDA algorithm generally shows superior or competitive performance compared to other algorithms. For instance, in the Breastcancer dataset, mDA achieves a classification accuracy of 0.9678, slightly outperforming the nearest competitor, GOA, which scored 0.9673. Similarly, in the BreastEW dataset, mDA's accuracy of 0.9382 is among the highest, surpassed only by GA's 0.9581. However, in the Colon and PenglungEW datasets, while mDA does not lead, it still maintains a respectable performance, indicating its robustness across different types of data challenges. This comparative analysis highlights the efficacy of mDA in achieving high classification accuracy and showcases its potential as a reliable tool in medical data analysis.

Table 4. A comparative analysis of classification accuracy between the proposed mDA and other algorithms.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	95.81 ± 0.21	96.42 ± 0.78	96.14 ± 0.81	95.91 ± 1	96.73 ± 2.6	94.99 ± 1.8	96.24 ± 0.01	**96.78**± 1.18
BreastEW	91.87 ± 2.71	93.31 ± 2.51	**94.52** ± 5.32	93.00 ± 1.34	94.22 ± 1.55	93.96 ± 0.43	94.26 ± 1.75	93.82 ± 0.85
Colon	64.59 ± 0.85	67.25 ± 1.7	63.98 ± 1.91	61.22 ± 0.12	59.29 ± 1.16	66.63 ± 0.11	69.29 ± 3.12	**71.74** ± 1.53
HeartEW	81.90 ± 0.01	79.93 ± 0.22	79.28 ± 1.19	79.61 ± 3.42	74.31 ± 2.3	78.54 ± 4.48	79.47 ± 1.24	**82.27** ± 0.01
Leukemia	72.11 ± 0.5	79.65 ± 1.55	81.14 ± 3.69	62.11 ± 3	64.83 ± 1.52	78.42 ± 2.99	81.84 ± 0.84	**82.11** ± 2.47
Lymphography	75.98 ± 1.01	75.42 ± 2.26	72.42 ± 1.64	74.75 ± 2.24	75.00 ± 0.16	71.91 ± 0.92	68.14 ± 5.25	**76.27** ± 4.24
PenglungEW	53.79 ± 1.74	47.24 ± 0.81	49.74 ± 0.01	61.55 ± 1.14	45.00 ± 1.77	40.00 ± 4.86	**63.02** ± 1.77	60.35 ± 3.27
Ranking (Wins)	0	0	1	0	0	0	1	5
Ranking (Ties)	0	0	0	0	0	0	0	0
Ranking (Losses)	7	7	6	7	7	7	6	2
Ranks (F-test)	4.8571	4.0000	4.5714	5.5714	5.5714	6.1429	3.4286	1.8571

Table 5 presents a comparative analysis of the average number of selected features across various algorithms when applied to different medical datasets. The table highlights the efficiency of each algorithm in feature selection, which is critical for reducing model complexity while maintaining or enhancing prediction accuracy. For instance, in the Breastcancer dataset, mDA and HHO select the fewest features, demonstrating their capability to achieve efficient feature reduction. In contrast, the Colon dataset shows significant variation in the number of features selected, with mDA choosing 923.20 features on average, which is lower than GWO and SSA but higher than DA and HHO.

Table 5. A comparison results in terms of average number of selected features.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	4.45	**4.00**	5.55	5.05	5.00	**4.00**	5.00	5.00
BreastEW	14.00	15.90	23.60	18.05	15.75	19.85	14.80	**13.60**
Colon	697.40	943.50	1150.95	1164.40	**654.10**	973.80	873.30	923.20
HeartEW	10.35	7.80	11.35	10.85	**7.60**	8.15	8.85	8.40
Leukemia	2952.00	3443.35	3785.75	3929.25	**1320.05**	3585.65	3088.25	3332.30
Lymphography	10.85	12.40	14.60	12.20	**9.25**	11.25	11.35	10.30
PenglungEW	**72.20**	138.70	182.70	192.20	85.10	160.20	123.70	127.50
Ranking (Wins)	1	0	0	0	4	0	0	1
Ranking (Ties)	0	1	0	0	0	1	0	0
Ranking (Losses)	6	6	7	7	3	6	7	6
Ranks (F-test)	2.7143	4.3571	7.5714	7.1429	2.1429	4.7857	3.8571	3.4286

In more complex datasets like Leukemia, which initially have thousands of features, the variation in selected features is stark, with mDA selecting 3332.30 features, substantially fewer than GWO and SSA but more than GOA and DA. This indicates that while mDA generally selects more features compared to some algorithms, it may balance feature reduction and maintaining classification performance.

Overall, this table effectively demonstrates how different optimization algorithms manage the trade-off between reducing the number of features and retaining sufficient information for accurate classification, providing insights into their applicability in various scenarios.

Table 6 summarizes the feature selection ratio and dimensionality reduction efficiency obtained by the proposed mDA across all benchmark datasets. The results show that mDA consistently achieves significant feature-space reduction, eliminating on average nearly half of the original features (≈49%) while maintaining strong classification performance. In particular, for medium-dimensional datasets such as BreastEW and PenglungEW, mDA attains up to 60% reduction efficiency, highlighting its robustness in identifying highly relevant features. Although the reduction rate is moderate in high-dimensional datasets such as Colon and Leukemia (≈53%), this behavior reflects the algorithm's tendency to preserve discriminative information essential for accurate classification. Overall, the results confirm that mDA provides an effective balance between compactness and accuracy, outperforming several baseline algorithms.

Table 6. Feature Selection Ratio and Dimensionality Reduction Efficiency using mDA Algorithm.

Dataset	Total-Features	Selected Features (mDA)	Feature-Selection Ratio (%)	Dimensionality Reduction Efficiency (%)
Breastcancer	9	5.00	55.56	44.44
BreastEW	30	13.60	45.33	54.67
Colon	2000	923.20	46.16	53.84
HeartEW	13	8.40	64.62	35.38
Leukemia	7129	3332.30	46.76	53.24
Lymphography	18	10.30	57.22	42.78
PenglungEW	325	127.50	39.23	60.77

Table 7 offers a comparative analysis of average best fitness values achieved by various algorithms when applied to different datasets in the research. This table shows the effectiveness of each algorithm based on their fitness values, a measure reflecting how well each algorithm has optimized a predefined objective function. For example, the Breastcancer dataset shows that mDA achieves an impressive fitness value of 0.0233, indicating its superior performance in optimizing the classification task compared to all other algorithms. Similarly, in the Leukemia dataset, mDA reports the lowest fitness value of 0.0064, suggesting a significant optimization capability over the alternatives.

However, in the Colon dataset, GA displays lower fitness values, indicating potential areas where mDA could be less effective or require adjustments to enhance its optimization performance. The trend across different datasets highlights the varying effectiveness of these algorithms in specific scenarios, offering insights into their potential applications and limitations.

Overall, this comparative analysis provides a clear perspective on the optimization capabilities of each algorithm, with mDA generally demonstrating robust performance across most datasets, underscoring its utility in solving complex classification problems effectively.

Furthermore, we used the F-test to statistically validate the difference in the performance obtained. The results of the F-test are displayed under each table. However, the comparative performance is summarized in each table using the Wins–Ties–Losses (WTL) criterion. More wins are an indication of superior overall performance consistency.

Figure 1 depicts the convergence curves for the standard (DA) and the modified DA (mDA) over a series of iterations applied to the given dataset. The x-axis represents the number of iterations, ranging from 0 to 100, while the y-axis measures the average best-so-far fitness values, which indicate the optimization performance of each algorithm at each iteration.

Table 7. Comparison of the average best fitness values between the proposed mDA and other methods.

Benchmark	GA	PSO	GWO	SSA	GOA	HHO	DA	mDA
Breastcancer	0.0274 ± 0.0009	0.0239 ± 0.0029	0.0249 ± 0.005	0.0233 ± 0.0115	0.0250 ± 0.0085	0.0257 ± 0.0063	0.0250 ± 0.0001	**0.0233** ± 0.0017
BreastEW	0.0208 ± 0.0016	0.0271 ± 0.0035	0.0316 ± 0.0086	0.0286 ± 0.0074	0.0261 ± 0.0018	0.0260 ± 0.0027	0.0257 ± 0.0054	**0.0169** ± 0.002
Colon	**0.0459** ± 0.0086	0.1229 ± 0.0094	0.1341 ± 0.0159	0.1068 ± 0.0015	0.0699 ± 0.0006	0.0726 ± 0.0072	0.0872 ± 0.0066	0.0763 ± 0.0122
HeartEW	0.1077 ± 0.0004	0.0963 ± 0.0001	0.1142 ± 0.0015	0.1046 ± 0.0067	**0.0920** ± 0.0038	0.0979 ± 0.0136	0.1038 ± 0.0131	0.1023 ± 0.0001
Leukemia	0.0154 ± 0.0019	0.0187 ± 0.0072	0.0939 ± 0.004	0.0559 ± 0.0076	0.0392 ± 0.0096	0.0441 ± 0.0001	0.0278 ±0.002	**0.0064** ± 0.0027
Lymphography	0.0652 ± 0.0026	0.0765 ± 0.0053	0.0878 ± 0.0145	**0.0601** ± 0.0003	0.0781 ± 0.0021	0.0709 ± 0.0019	0.0650 ± 0.0082	0.0686 ± 0.0052
PenglungEW	0.0526 ± 0.0083	0.0469 ± 0.013	0.1038 ± 0.0001	0.0409 ± 0.0045	**0.0376** ± 0.0061	0.0459 ± 0.0088	0.0388 ± 0.0054	0.0381 ± 0.0003
Ranking (Wins)	1	0	0	1	2	0	0	3
Ranking (Ties)	0	0	0	0	0	0	0	0
Ranking (Losses)	6	7	7	6	5	7	7	4
Ranks (F-test)	4.2857	4.7143	7.4286	4.6429	3.7857	4.7143	3.9286	2.5000

From the graph, it is evident that mDA (shown in red) starts with higher fitness values than traditional DA (shown in black), and maintains a consistent improvement, demonstrating a steeper and more continuous decline in the fitness values as the iterations progress. This suggests that mDA converges more effectively towards the optimum solution than DA in most cases, showing a more gradual decline in fitness values. By iteration 100, mDA achieves significantly lower fitness values than DA, indicating better optimization performance for all datasets.

The convergence curve for mDA is relatively smoother and steeper, which underscores its enhanced capability to quickly and efficiently refine solutions, likely due to improved algorithmic modifications that enhance its search and optimization processes within the feature space of the datasets. These graphs effectively illustrate the comparative advantage of mDA in achieving lower fitness values faster, which is indicative of better overall performance in optimizing the classification task for this particular dataset.

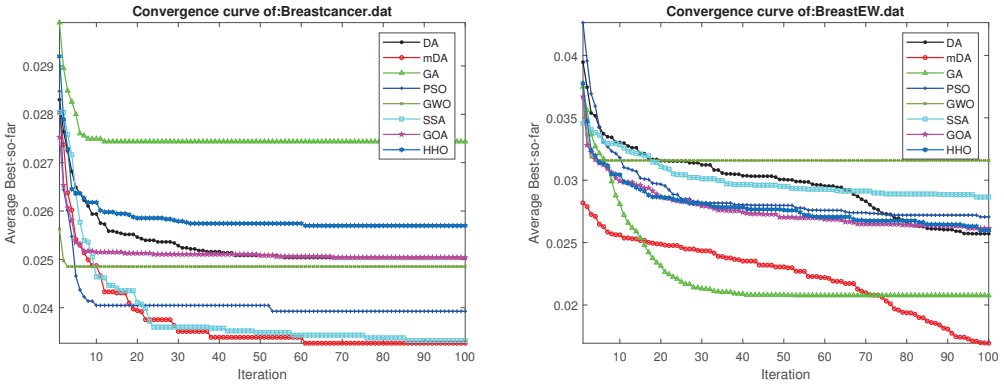


Figure 1. *Cont.*

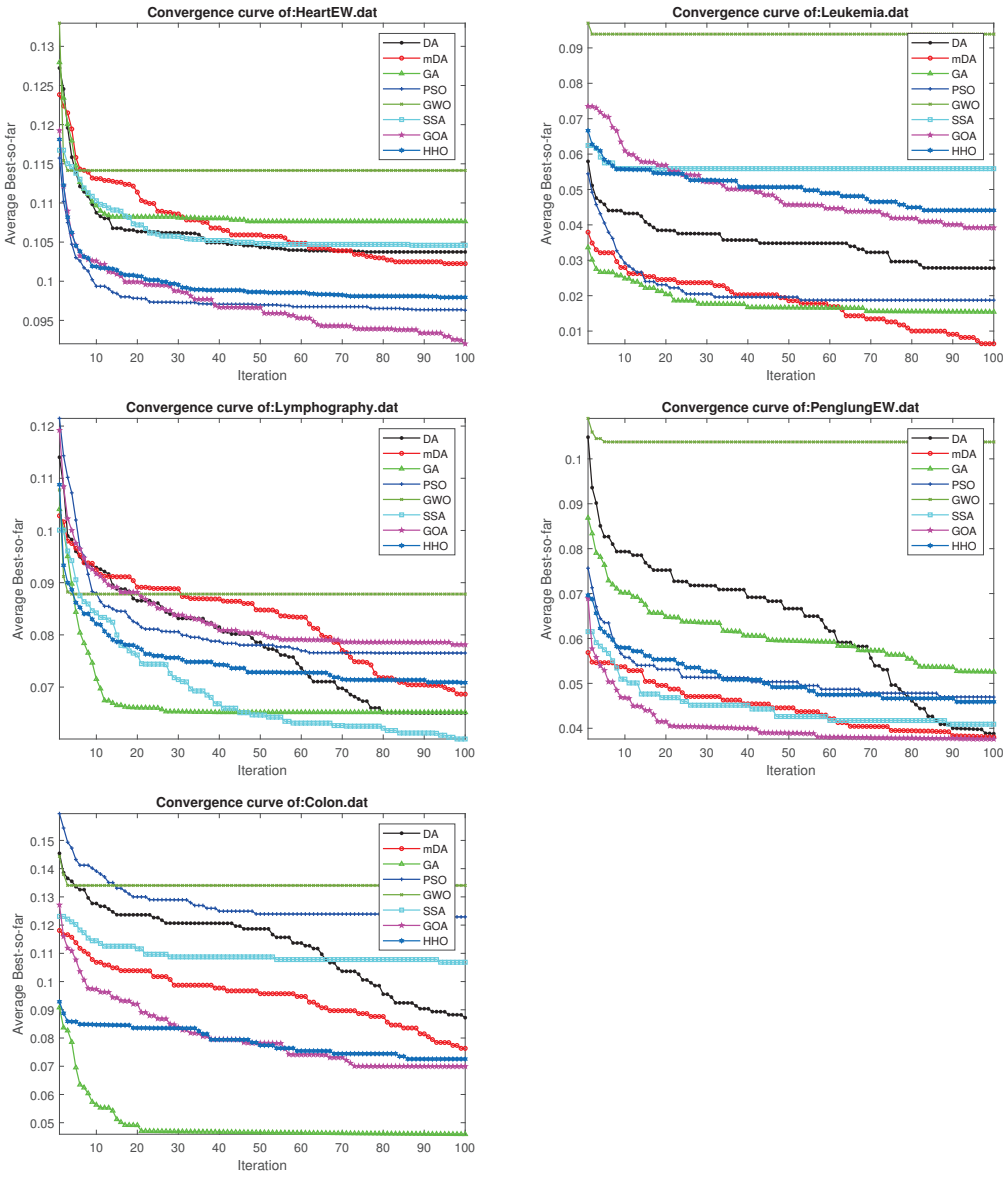


Figure 1. Convergence curves for the tested algorithms over the tested datasets.

To statistically assess the difference in performance between the algorithms compared, we employed a range of non-parametric tests that can be applied to multiple algorithms and datasets. The first step was to test the hypothesis that the performances of all algorithms are equivalent using the Friedman test, which evaluates the null hypothesis that all algorithms achieve equivalent performance across datasets. This test is appropriate here because it is rank-based and does not rely on normality assumptions. As shown in Table 8, the Friedman statistic was $\chi^2 = 201.465$ with 7 degrees of freedom, resulting in an extremely small p-value (5.618×10^{-40}). This strongly rejects the null hypothesis and confirms that there are significant performance differences between the algorithms.

Table 8. Friedman test for differences among algorithms across all datasets and runs.

Test	Value	df
χ^2 statistic	201.465	7
p-value	5.618×10^{-40}	—

Then we computed the average rank of each algorithm across all datasets, presented in Table 9. The proposed mDA algorithm achieved the lowest average rank (2.304), indicating the most consistent superior performance relative to the other baseline methods.

Table 9. Average ranks of the algorithms across all datasets and runs (lower is better).

Algorithm	Average Rank
mDA	2.304
BDA	3.732
BPSO	4.196
bGWO	4.511
BGA	4.896
BSSA	5.321
BGOA	5.421
HHO	5.618

To identify which performance differences were statistically significant at the pairwise level, a Nemenyi post-hoc test was performed. The results in Table 10 show that mDA is significantly better than all other algorithms at the $\alpha = 0.05$ significance level, with all p-values being extremely small. This confirms that the superiority of mDA is not due to random fluctuations but reflects genuine performance advantages across benchmarks.

Table 10. Nemenyi post-hoc test results comparing the proposed mDA to other algorithms (significance at $\alpha = 0.05$).

Algorithm	p-Value	Significant ($\alpha = 0.05$)
BGA	1.110223×10^{-16}	Yes
BPSO	2.827673×10^{-9}	Yes
bGWO	1.327716×10^{-12}	Yes
BSSA	0	Yes
BGOA	0	Yes
HHO	0	Yes
BDA	2.917804×10^{-5}	Yes

The Critical Difference (CD) diagram in Figure 2 provides a visual representation of these statistical findings. Algorithms whose rank differences exceed the CD threshold (0.97) are considered significantly different. As depicted, mDA is clearly separated from all other methods and lies well outside the CD interval of competing approaches, reinforcing the conclusion that its performance advantage is statistically significant.

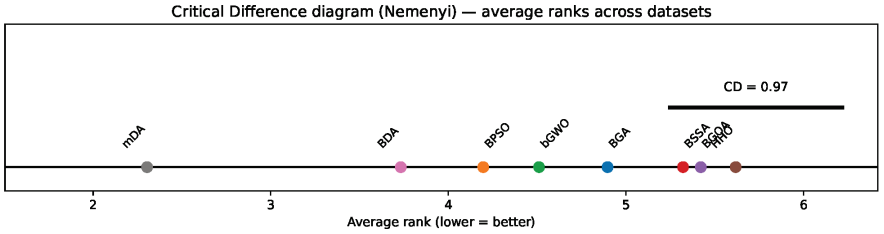


Figure 2. Critical Difference (Nemenyi) diagram showing the average ranks of the algorithms.

Taken all together, the Friedman test result, the result of the post-hoc test using the Nemenyi method, and the CD diagram provide a strong indication that the new mDA algorithm performs better than all of the baselines. The above findings provide strong evidence for the reliability of the advantages of the mDA algorithm.

Moreover, a Filter Feature Selection (FFS-) technique was performed for further analysis. Table 11 compares the proposed approach and various classification methods combined with filter feature selection. The results illustrate the proposed approach's superior performance on most datasets. For instance, the mDA algorithm achieved the best results on five out of seven datasets: breast cancer, BreastEW, Heart, and PenglungEW, with accuracies of 0.9678, 0.9382, 0.8227, 0.7627, and 0.6035, respectively. As for the remaining datasets, Colon and Leukaemia achieved the highest accuracies of 0.9838 and 0.8611 using J48 and AdaBoost, respectively. This discrepancy arises because the Colon and Leukaemia datasets may possess specific characteristics that do not align well with the strengths of the mDA algorithm. These characteristics could include different distributions of features, higher levels of noise, or imbalanced class distributions, which the mDA algorithm might not handle as effectively as other algorithms like J48 or AdaBoost.

Table 11. Classification accuracy results comparing the proposed mDA algorithm with other algorithms using filter feature selection.

	FFS-J48	FFS-RF	FFS-k-NN	FFS-AdaBoost	FFS-Bagging	mDA
Breast-cancer	0.7428	0.6974	0.7284	0.7162	0.6841	**0.9678**
BreastEW	0.6296	0.6301	0.7305	0.6269	0.6465	**0.9382**
Colon	**0.9838**	0.9516	0.9193	0.9677	0.8870	0.7174
Heart	0.7814	0.7962	0.7614	0.8159	0.8155	**0.8227**
Leukaemia	0.8198	0.7297	0.6527	**0.8611**	0.6527	0.8211
Lymphography	0.7432	0.6486	0.7364	0.7498	0.7229	**0.7627**
PenglungEW	0.4505	0.5616	0.5890	0.4931	0.4109	**0.6035**

6. Clinical Discussion

Beyond the quantitative improvements achieved by the proposed mDA algorithm across the seven medical benchmarks, we further examined the clinical relevance of the most frequently selected features to ensure that the selected subsets align with established biomedical understanding.

For the **Breastcancer** and **BreastEW** datasets, the selected attributes consistently highlighted morphological and cytological descriptors such as cell uniformity, epithelial cell size, bare nuclei, and chromatin texture. These indicators have a correspondence with histopathologic criteria used by pathologists for distinguishing between benign and malignant lesions and for estimating tumor aggressiveness, thus verifying that the algorithm can identify salient features for diagnosis.

In the **HeartEW** dataset, the retained variables primarily encompassed cardiorespiratory and electrocardiographic markers—including chest pain type, ST-segment slope, and exercise induced responses, that are well-established predictors of ischemia and cardiovascular risk in coronary artery disease (CAD). The resulting can mirror the features commonly employed in clinical scoring systems for non-invasive cardiac diagnosis.

For the **ColonEW** and LeukemiaEW gene-expression benchmarks, mDA identified compact and biologically interpretable gene signatures associated with pathways in cell-cycle control, apoptosis regulation, and hematopoietic proliferation. These subsets capture known genomic signals implicated in oncogenic progression, reflecting the model's capacity to balance sparsity with biological coherence.

In the **Lymphography** dataset, the selected variables predominantly involved patterns of lymph node enlargement, capsularity, and structural irregularity, which correspond to the clinical staging criteria used in lymphoma assessment. Such features directly relate to tumor dissemination and disease extent, indicating that mDA emphasizes anatomically meaningful discriminants.

Finally, for the multi-class **PenglungEW** dataset, the retained subset covered heterogeneous cytological and morphological indicators that capture inter-subtype variation across lung cancer classes. The inclusion of a slightly larger feature set here reflects the need to preserve separability across diversehistopathological subtypes. Together, these findings demonstrate that the mDA optimizer systematically gravitates toward clinically coherent and biologically plausible feature sets rather than arbitrary statistical artifacts. This not only enhances interpretability but also reinforces the practical translational value of the proposed framework in medical diagnostic contexts.

7. Conclusions and Future Works

This research introduces an enhanced hybrid DA optimizer incorporating EPD, aimed at boosting the performance of the standard DA in handling FS tasks. The mDA methodology was extensively applied to seven medical datasets. Detailed comparisons of the mDA's overall classification accuracy, selected features, fitness, and convergence characteristics were made against several well-known metaheuristic-based techniques. The exhaustive comparative results and analysis demonstrated the superior effectiveness of the proposed algorithm for various FS tasks within the medical field.

The conducted experiments confirm the proposed mDA algorithm's efficacy for feature selection within medical fields. Its capability to attain higher accuracy with a reduced number of features highlights its potential as a valuable tool for medical data analysis, providing a balance between clarity and predictive performance.

Future research could explore applying the EPD approach to various other population-based optimization algorithms. The effectiveness of the proposed binary DA and EPD-based techniques could also be utilized to address additional data-mining challenges. In future work, we plan to compare the proposed mDA with various FS methods within the domain. Furthermore, the proposed binary Dragonfly Algorithm (DA) and EPD-based techniques can be effectively utilized to tackle other complex data-mining challenges, especially those dealing with high-dimensional, unstructured, or heterogeneous datasets. As an illustration, future research could investigate areas such as LASSO and ridge regression, image-based feature selection, multimodal data analysis, and other intricate fields to demonstrate the scalability and practical applicability of the proposed methods. Future work should not only focus on technical advancements but also address significant socio-technical and ethical open research questions (ORQs) highlighted by this study and related literature. These encompass concerns such as transparency, fairness, responsible usage, computational sustainability, and the impact of human–AI interaction in the deployment of EPD- and DA-based algorithms.

Author Contributions: I.A.: Conceptualization, Methodology, Supervision, Project administration; I.A. and A.A.: Data curation, Formal analysis, Software; I.A., A.A., N.A.-M. and A.M.A.-Z.: Writing—review and editing, validation; A.S.: Software, Resources, Writing—review & editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: This research has been done during Ibrahim Aljarah's sabbatical leave from the University of Jordan for the academic year 2023–2024, and we express our gratitude to the University of Jordan for its support in enabling this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Eshtay, M.; Faris, H.; Obeid, N. Improving Extreme Learning Machine by Competitive Swarm Optimization and its application for medical diagnosis problems. *Expert Syst. Appl.* **2018**, *104*, 134–152. [CrossRef]
2. Khalilpourazari, S.; Pasandideh, S.H.R.; Niaki, S.T.A. Optimization of multi-product economic production quantity model with partial backordering and physical constraints: SQP, SFS, SA, and WCA. *Appl. Soft Comput.* **2016**, *49*, 770–791. [CrossRef]
3. Zelinka, I. A survey on evolutionary algorithms dynamics and its complexity–Mutual relations, past, present and future. *Swarm Evol. Comput.* **2015**, *25*, 2–14. [CrossRef]
4. Heidari, A.A.; Abbaspour, R.A.; Jordehi, A.R. Gaussian bare-bones water cycle algorithm for optimal reactive power dispatch in electrical power systems. *Appl. Soft Comput.* **2017**, *57*, 657–671. [CrossRef]
5. Heidari, A.A.; Abbaspour, R.A.; Jordehi, A.R. An efficient chaotic water cycle algorithm for optimization tasks. *Neural Comput. Appl.* **2017**, *28*, 57–85. [CrossRef]
6. Mirjalili, S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput. Appl.* **2016**, *27*, 1053–1073. [CrossRef]
7. Bhesdadiya, R.H.; Pandya, M.H.; Trivedi, I.N.; Jangir, N.; Jangir, P.; Kumar, A. Price penalty factors based approach for combined economic emission dispatch problem solution using Dragonfly Algorithm. In *Proceedings of the 2016 International Conference on Energy Efficient Technologies for Sustainability (ICEETS), Nagercoil, India, 7–8 April 2016*; IEEE: New York, NY, USA, 2016; pp. 436–441. [CrossRef]
8. Suresh, V.; Sreejith, S. Generation dispatch of combined solar thermal systems using dragonfly algorithm. *Computing* **2017**, *99*, 59–80. [CrossRef]
9. Hamdy, M.; Nguyen, A.T.; Hensen, J.L.M. A performance comparison of multi-objective optimization algorithms for solving nearly-zero-energy-building design problems. *Energy Build.* **2016**, *121*, 57–71. [CrossRef]
10. Daely, P.T.; Shin, S.Y. Range based wireless node localization using Dragonfly Algorithm. In *Proceedings of the 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, Austria, 5–8 July 2016*; IEEE: New York, NY, USA, 2016; pp. 1012–1015. [CrossRef]
11. Elhariri, E.; El-Bendary, N.; Hassanien, A.E. Bio-inspired optimization for feature set dimensionality reduction. In *Proceedings of the 2016 3rd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), Zouk Mosbeh, Lebanon, 13–15 July 2016*; IEEE: New York, NY, USA, 2016; pp. 184–189. [CrossRef]
12. Salam, M.A.; Zawbaa, H.M.; Emary, E.; Ghany, K.K.A.; Parv, B. A hybrid dragonfly algorithm with extreme learning machine for prediction. In *Proceedings of the 2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Sinaia, Romania, 2–5 August 2016*; IEEE: New York, NY, USA, 2016; pp. 1–6. [CrossRef]
13. Kanan, T.; AbedAlghafer, A.; AlZu'bi, S.; Hawashin, B.; Mughaid, A.; Kanaan, G.; Kamruzzaman, M. An Intelligent Health Care System for Detecting Drug Abuse in Social Media Platforms Based on Low Resource Language. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2023**, *32*, 691–703. [CrossRef]
14. Kanan, T.; Hawashin, B.; Alzubi, S.; Almaita, E.; Alkhatib, A.; Maria, K.A.; Elbes, M. Improving arabic text classification using p-stemmer. *Recent Adv. Comput. Sci. Commun. (Former. Recent Patents Comput. Sci.)* **2022**, *15*, 404–411. [CrossRef]
15. Mansour, A.M.; Obeidat, M.A.; Hawashin, B. A novel multi agent recommender system for user interests extraction. *Clust. Comput.* **2023**, *26*, 1353–1362. [CrossRef]
16. Mafarja, M.M.; Eleyan, D.; Jaber, I.; Hammouri, A.; Mirjalili, S. Binary dragonfly algorithm for feature selection. In *Proceedings of the 2017 International Conference on New Trends in Computing Sciences (ICTCS), Amman, Jordan, 11–13 October 2017*; IEEE: New York, NY, USA, 2017; pp. 12–17. [CrossRef]
17. Raman, G.; Raman, G.; Manickam, C.; Ganesan, S.I. Dragonfly algorithm based global maximum power point tracker for photovoltaic systems. In *Proceedings of the International Conference in Swarm Intelligence, Brussels, Belgium, 7–9 September 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 211–219. [CrossRef]
18. Hema, C.; Sankar, S.; Sandhya. Energy efficient cluster based protocol to extend the RFID network lifetime using dragonfly algorithm. In *Proceedings of the 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 6–8 April 2016*; IEEE: New York, NY, USA, 2016; pp. 530–534. [CrossRef]
19. Abdel-Basset, M.; Luo, Q.; Miao, F.; Zhou, Y. Solving 0–1 knapsack problems by binary dragonfly algorithm. In *Proceedings of the International Conference on Intelligent Computing, Liverpool, UK, 7–10 August 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 491–502. [CrossRef]
20. KS, S.R.; Murugan, S. Memory based hybrid dragonfly algorithm for numerical optimization problems. *Expert Syst. Appl.* **2017**, *83*, 63–78. [CrossRef]
21. Song, J.; Li, S. Elite opposition learning and exponential function steps-based dragonfly algorithm for global optimization. In *Proceedings of the 2017 IEEE International Conference on Information and Automation (ICIA), Macao, China, 18–20 July 2017*; IEEE: New York, NY, USA, 2017; pp. 1178–1183. [CrossRef]

22. Majeed, N.M.; Ramo, F.M. Implementation of Features Selection Based on Dragonfly Optimization Algorithm. *Technium* **2022**, *4*, 44–52. [CrossRef]
23. Raouf, Z.T.; Abd, D.H. Feature Selection for Binary Dataset using Dragonfly Algorithm. In *Proceedings of the 2023 16th International Conference on Developments in eSystems Engineering (DeSE), Istanbul, Türkiye, 18–20 December 2023*; IEEE: New York, NY, USA, 2023; pp. 480–485. [CrossRef]
24. Sarvamangala, D.; Kulkarni, R.V. A comparative study of bio-inspired algorithms for medical image registration. In *Advances in Intelligent Computing*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 27–44. [CrossRef]
25. Díaz-Cortés, M.A.; Ortega-Sánchez, N.; Hinojosa, S.; Oliva, D.; Cuevas, E.; Rojas, R.; Demin, A. A multi-level thresholding method for breast thermograms analysis using Dragonfly algorithm. *Infrared Phys. Technol.* **2018**, *93*, 346–361. [CrossRef]
26. Song, X.; Zhang, Y.; Zhang, W. Evolutionary computation for feature selection in classification: A comprehensive survey of solutions, applications and challenges. *Swarm Evol. Comput.* **2024**, *90*, 101661. [CrossRef]
27. Zhang, Y.; Wang, J.; Gorriz, J.M.; Wang, S. Deep Learning and Vision Transformer for Medical Image Analysis. *J. Imaging* **2023**, *9*, 147. [CrossRef]
28. Chaddad, A.; Peng, J.; Xu, J.; Bouridane, A. Survey of Explainable AI Techniques in Healthcare. *Sensors* **2023**, *23*, 634. [CrossRef]
29. Feng, J.; Zhou, H.; Xu, W.; Liu, Q. Sliding-attention transformer neural architecture for predicting T cell receptor–antigen–human leucocyte antigen binding. *Nat. Mach. Intell.* **2024**, *6*, 981–994. [CrossRef]
30. Kababulut, F.Y.; Gürkan Kuntalp, D.; Düzyel, O.; Özcan, N.; Kuntalp, M. A New Shapley-Based Feature Selection Method in a Clinical Decision Support System for the Identification of Lung Diseases. *Diagnostics* **2023**, *13*, 3558. [CrossRef]
31. Hur, S.; Lee, Y.; Park, J.; Jeon, Y.J.; Cho, J.H.; Cho, D.; Lim, D.; Hwang, W.; Cha, W.C.; Yoo, J. Comparison of SHAP and clinician friendly explanations reveals effects on clinical decision behaviour. *npj Digit. Med.* **2025**, *8*, 578. [CrossRef]
32. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. *ACM Siggraph Comput. Graph.* **1987**, *21*, 25–34. [CrossRef]
33. Mirjalili, S.; Lewis, A. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization. *Swarm Evol. Comput.* **2013**, *9*, 1–14. [CrossRef]
34. Mirjalili, S.; Wang, G.G.; Coelho, L.d.S. Binary optimization using hybrid particle swarm optimization and gravitational search algorithm. *Neural Comput. Appl.* **2014**, *25*, 1423–1435. [CrossRef]
35. Bak, P.; Tang, C.; Wiesenfeld, K. Self-organized criticality: An explanation of the 1/f noise. *Phys. Rev. Lett.* **1987**, *59*, 381. [CrossRef]
36. Lewis, A.; Mostaghim, S.; Randall, M. Evolutionary population dynamics and multi-objective optimisation problems. In *Multi-Objective Optimization in Computational Intelligence: Theory and Practice*; IGI Global: Hershey, PA, USA, 2008; pp. 185–206. [CrossRef]
37. Lewis, A.; Abramson, D.; Peachey, T. An evolutionary programming algorithm for automatic engineering design. In *Proceedings of the International Conference on Parallel Processing and Applied Mathematics, Czestochowa, Poland, 7–10 September 2003*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 586–594. [CrossRef]
38. Boettcher, S.; Percus, A. Extremal optimization: Methods derived from co-evolution. *arXiv* **1999**, arXiv:math/9904056. [CrossRef]
39. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **1992**, *46*, 175–185. [CrossRef]
40. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary antlion approaches for feature selection. *Neurocomputing* **2016**, *213*, 54–65. [CrossRef]
41. Zawbaa, H.M.; Emary, E.; Parv, B.; Sharawi, M. Feature selection approach based on moth-flame optimization algorithm. In *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada, 24–29 July 2016*; IEEE: New York, NY, USA, 2016; pp. 4612–4617. [CrossRef]
42. Alzaqebah, A.; Al-Kadi, O.; Aljarah, I. An enhanced harris hawk optimizer based on extreme learning machine for feature selection. *Prog. Artif. Intell.* **2023**, *12*, 77–97. [CrossRef]
43. Mafarja, M.; Aljarah, I.; Heidari, A.A.; Faris, H.; Fournier-Viger, P.; Li, X.; Mirjalili, S. Binary dragonfly optimization for feature selection using time-varying transfer functions. *Knowl.-Based Syst.* **2018**, *161*, 185–204. [CrossRef]
44. Hammouri, A.I.; Awadallah, M.A.; Braik, M.S.; Al-Betar, M.A.; Beseiso, M. Improved Dwarf Mongoose Optimization Algorithm for Feature Selection: Application in Software Fault Prediction Datasets. *J. Bionic Eng.* **2024**, *21*, 2000–2033. [CrossRef]
45. Mafarja, M.; Mirjalili, S. Whale optimization approaches for wrapper feature selection. *Appl. Soft Comput.* **2018**, *62*, 441–453. [CrossRef]
46. Mafarja, M.; Aljarah, I.; Faris, H.; Hammouri, A.I.; Ala'M, A.Z.; Mirjalili, S. Binary grasshopper optimisation algorithm approaches for feature selection problems. *Expert Syst. Appl.* **2019**, *117*, 267–286. [CrossRef]
47. Alzaqebah, A.; Aljarah, I.; Al-Kadi, O. A hierarchical intrusion detection system based on extreme learning machine and nature-inspired optimization. *Comput. Secur.* **2023**, *124*, 102957. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

symmetry

MDPI

Article

PS-InSAR Monitoring Integrated with a Bayesian-Optimized CNN–LSTM for Predicting Surface Subsidence in Complex Mining Goafs Under a Symmetry Perspective

Tianlong Su [1,2], Linxin Zhang [3], Xuzhao Yuan [1,2], Xiaoquan Li [1,2,*], Xuefeng Li [1,2,*], Xuxing Huang [4], Zheng Huang [1,2] and Danhua Zhu [5]

[1] School of Resources, Environment and Materials, Guangxi University, Nanning 530004, China
[2] Key Laboratory of Minerals Engineering, Guangxi Higher School, Nanning 530004, China
[3] Guangxi Industrial Design Group Co., Ltd., Nanning 530023, China
[4] Changsha Institute of Mining Research Co., Ltd., Changsha 410012, China
[5] Archives Office, Guangxi Minzu University, Nanning 530006, China
* Correspondence: lxq0927@gxu.edu.cn (X.L.); lixfcshncn@126.com (X.L.)

Abstract: Mine-induced surface subsidence threatens infrastructure and can trigger cascading geohazards, so accurate and computationally efficient monitoring and forecasting are essential for early warning. We integrate Persistent Scatterer InSAR (PS-InSAR) time series with a Bayesian-optimized CNN–LSTM designed for spatiotemporal prediction. The CNN extracts spatial deformation patterns, the LSTM models temporal dependence, and Bayesian optimization selects the architecture, training hyperparameters, and the most informative exogenous drivers. Groundwater level and backfilling intensity are encoded as multichannel inputs. Endpoint anchoring with affine calibration aligns the historical series and the forward projections. PS-InSAR indicates a maximum subsidence rate of 85.6 mm yr^{-1}, and the estimates are corroborated against nearby leveling benchmarks and $FLAC^{3D}$ simulations. Cross-site comparisons show acceleration followed by deceleration after backfilling and groundwater recovery, which is consistent with geological engineering conditions. A symmetry-aware preprocessing step exploits axial regularities of the deformation field through mirroring augmentation and documents symmetry-breaking hotspots linked to geological heterogeneity. These choices improve generalization to shifted and oscillatory patterns in both the spatial CNN and the temporal LSTM branches. Short-term forecasts from the BO–CNN–LSTM indicate subsequent stabilization with localized rebound, highlighting its practical value for operational planning and risk mitigation. The framework combines automated hyperparameter search with physically consistent objectives, reduces manual tuning, enhances reproducibility and generalizability, and provides a transferable quantitative workflow for forecasting mine-induced deformation in complex goaf systems.

Keywords: surface subsidence; PS-InSAR; Bayesian optimization; CNN–LSTM; time-series forecasting

1. Introduction

Mineral resources underpin national economic development, and rapid growth has markedly increased national demand. In parallel, the scale and intensity of mining have expanded, bringing mine safety to the fore. Statistics indicate that accidents related to underground extraction-induced goafs account for more than 30% [1]. Goaf formation associated with mining activities poses substantial geotechnical risks—including roof

falls and surface subsidence—and may escalate into catastrophic collapse or inrush flooding [2–4]. These hazards compromise operational safety and adversely affect communities near mining areas.

To investigate the patterns and spatial distribution of surface subsidence induced by underground mining, researchers have applied a range of measurement techniques to assess goaf stability and surface impacts. Conventional ground-based methods rely on discrete benchmarks and instruments—such as total stations, leveling surveys, Global Navigation Satellite Systems (GNSS), and terrestrial laser scanning (TLS) [5–10]—which offer high point-wise accuracy but are constrained by sparse spatial sampling and therefore cannot fully characterize basin-scale subsidence. With the advancement of remote sensing, UAV photogrammetry, airborne and terrestrial LiDAR, and InSAR now provide complementary capabilities for extensive deformation monitoring [11–15].

In this context, Persistent Scatterer InSAR (PS-InSAR) has become a key technique for long-term, high-precision deformation monitoring. PS-InSAR is a time-series radar interferometry technique that tracks phase-stable ground targets across many satellite passes to retrieve millimeter-scale surface displacement with wall-to-wall spatial coverage. By focusing on temporally stable point targets, PS-InSAR can achieve millimeter-level accuracy over multi-year periods and has been successfully applied to mining subsidence, urban ground settlement, and infrastructure stability assessments [16–18]. Compared with conventional D-InSAR and SBAS approaches, PS-InSAR is less sensitive to temporal decorrelation and atmospheric phase delays, making it particularly suitable for tracking gradual, multi-stage settlement above clustered goafs.

Against this backdrop, we propose a forecasting framework that retains the practical strengths of time-series InSAR while explicitly encoding domain structure to improve transferability. Specifically, we adopt a symmetry-aware perspective and couple it with a Bayesian-optimized CNN–LSTM. Panel-centric preprocessing aligns geometry and applies mirroring augmentation so that axial and mirror relations in subsidence bowls are visible to the model. Exogenous drivers—groundwater level and backfilling intensity—are introduced as multichannel inputs to reflect engineering interventions. Bayesian optimization provides an efficient, global strategy to select architectures, training hyperparameters, and informative driver subsets. Finally, we establish an interpretation link between departures from symmetry and the spatial distribution of residuals, offering actionable cues for mine management. These elements are organized under chronological splits with explicit settings to preserve transparency and reproducibility.

In this study, PS-InSAR is adopted as the principal technique for surface subsidence analysis. Previous studies have demonstrated the value of multi-sensor fusion: for example, Vishnuvardhan et al. combined MT-InSAR, PS-InSAR, SBAS, and GNSS with Kalman filtering to integrate displacement records and identify critical deformation drivers [19]. Comparative research indicates that PS-InSAR and SBAS surpass D-InSAR in long time series and under atmospheric decorrelation, yielding superior consistency [20,21]. Mining applications have revealed severe surface degradation and linear deformation tendencies [22,23], while integrated approaches with StaMPS-PS, GNSS, GPR, and optical remote sensing have produced reliable subsidence assessments in urban and industrial areas [24,25].

Accurate monitoring, however, is only the preliminary step for hazard mitigation; the fundamental goal is to establish reliable predictive models. Empirical methods such as the probability-integral model [26] are simple but involve parameters with limited physical interpretability and poor transferability. Numerical simulations such as FLAC^{3D}, based on finite-element or finite-difference formulations [27,28], can simulate mining-induced mechanical responses but are computationally expensive and strongly dependent on rigorous parameter calibration.

Given these limitations, the abundance of InSAR observations has motivated a data-driven predictive paradigm. Early machine-learning attempts using support vector machines (SVM) and shallow artificial neural networks (ANN) demonstrated preliminary success [29–31], but such shallow models have limited capability to capture the intricate spatiotemporal couplings of subsidence. Deep learning, with hierarchical feature extraction and nonlinear representational power, provides a more robust alternative [32].

Accordingly, hybrid architectures combining convolutional neural networks (CNN) and long short-term memory (LSTM) networks have emerged. The BO–CNN–LSTM is a spatiotemporal predictor that uses a convolutional neural network to learn spatial deformation patterns and a long short-term memory network to model temporal evolution, while Bayesian optimization automatically selects the architecture and training hyperparameters to improve accuracy and efficiency. LSTM, with its gated memory structure, excels at modeling long-term temporal dependencies and has been widely applied to point-wise subsidence forecasting [33]. CNN is effective in extracting spatial features from gridded data [34]. Within a CNN–LSTM framework, CNN layers derive spatial features from InSAR time series and pass them to LSTM modules, which capture temporal dynamics. This spatiotemporal fusion significantly enhances predictive accuracy by reflecting the inherent three-dimensional and temporal characteristics of subsidence [35,36].

Despite their promise, CNN–LSTM models are highly sensitive to hyperparameter settings. Current practice still relies heavily on trial and error or grid search to tune architecture and training parameters [37,38]. This approach is computationally demanding and often produces suboptimal solutions, leaving the challenge of efficient global optimization unresolved.

To address this challenge, we develop a BO–CNN–LSTM model for spatiotemporal subsidence forecasting. We analyze mine-induced deformation from a symmetry perspective: subsidence bowls are approximately axial or mirror symmetric about the panel centerline, while mapped faults, heterogeneous backfilling, and groundwater variations produce departures from that symmetry. In architecture, convolutions provide translation equivariance for spatial patterns, and LSTMs express time-translation regularity through parameter sharing across steps. We therefore adopt symmetry-aware preprocessing with panel-centric coordinates and baseline removal and apply mirroring augmentation so that axial and mirror relations are explicit to the model. We further identify where departures from symmetry concentrate and assess how they influence predictive accuracy. Bayesian optimization builds a surrogate from prior evaluations to guide an efficient hyperparameter search. The contributions are threefold: a symmetry-aware input and augmentation pipeline tailored to subsidence fields; a coupling of spatial–temporal feature learning with Bayesian optimization under a symmetry-informed design for robust and efficient training; and an interpretation strategy that links spatial patterns of symmetry departure to residual distribution for actionable insights in mine management.

2. Materials and Methods

2.1. Study Area

The study area derives from the concealed hazard investigation program at the mine with latent goaf hazards in Guangxi. The mine district is situated in the core of the Xunyu Plain of Guiping, Guangxi, China, with a total area of 1.7712 km^2. The maximum depth of the orebody reaches 400 m, and the overall mining depth attains 120 m. The geographical coordinates delimiting the study domain are 109°50′30″ to 109°51′15″ east longitude and 23°16′00″ to 23°16′45″ north latitude.

The orebody shown in Figure 1 has been extracted predominantly by the room-and-pillar mining method, with irregular pillars intentionally retained to support the mined-out

voids, thereby forming four vertically superposed goafs designated from top to bottom as No. 1 Goaf through No. 4 Goaf. The present working elevations range from +33 m to −42 m, and pillar heights vary between 4.60 m and 9.69 m. Earlier artisanal exploitation caused damage to the orebody, and subsequent extraction lacked systematic and scientifically justified planning, producing mined-out spaces with complicated three-dimensional morphologies, heterogeneous scales, and partial superposition, which seriously endangers the safety of surface agricultural land and ongoing underground operations. With continuing extraction, no collapse within the goafs and no tensile surface fissuring have been observed in the area to date. Attributes and spatial distributions of the four goafs are provided in Table 1 and Figure 1.

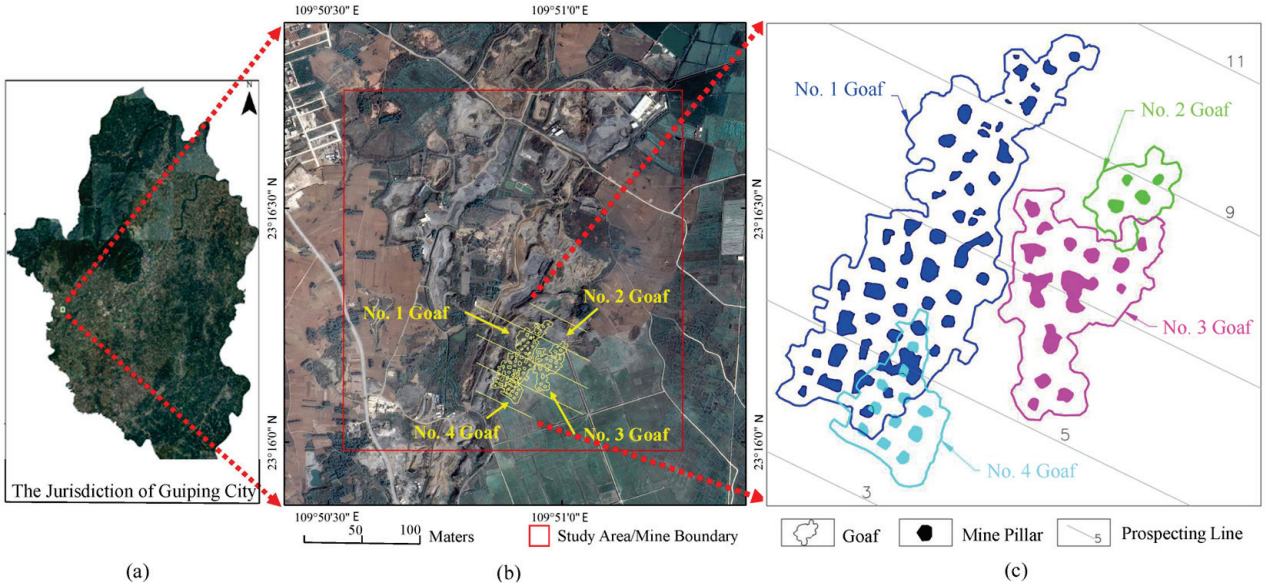


Figure 1. Geographical location map of the mining area. (**a**) The Xunyu Plain of Guiping, Guangxi, China. (**b**) DEM-based mine area extent. (**c**) Nadir view of the goaf area. Numbers 3, 5, 9, and 11 denote the indices of the geological prospecting lines.

Table 1. Goaf parameter information.

Goaf ID	Date of Formation (Year.Month)	Elevation Range (m)	Average Stope Height (m)	Goaf Area (m^2)	Goaf Volume (m^3)
No. 1 Goaf	2022.9	+33.6~−18.9	7.9	21,207.9	167,542.4
No. 2 Goaf	2018.6	+32.1~0.3	9.5	2591.5	24,697.0
No. 3 Goaf	2019.9	+2.4~−21.1	7.1	9117.5	64,460.7
No. 4 Goaf	2021.9	−33.3~−41.5	6.7	4941.0	33,104.5
Total				37,857.9	289,804.6

2.2. Dataset

The synthetic aperture radar dataset employed in this study originates from the European Space Agency Sentinel constellation. According to the formation chronology of the eastern goaf cluster, we selected thirty Sentinel-1A acquisitions on a descending track with VV polarization, spanning 2 September 2020 to 16 December 2023, in the single-look complex acquisition mode S1A*_SLC. The mean temporal separation between successive acquisitions is 27.2 days, and the incidence angle is 43.99 degrees.

For PS-InSAR processing, a digital elevation model with a spatial resolution of 30 m is required to perform differential interferometric removal of the topographic phase. The accuracy of the elevation model governs the effectiveness of the terrain removal during processing. To ensure robust terrain correction, we employed the NASA Shuttle Radar Topography Mission SRTM1 Version 3 elevation data at 30 m resolution.

Because the study area is strongly influenced by hydrological conditions and the underground extraction zone experiences seasonal inundation, we incorporated complementary datasets to improve the interpretation of mining-induced subsidence. These datasets include statistics of joint and fracture systems together with three-dimensional geometry, volumetrics, and positions of the mined-out voids, enabling numerical simulation of goaf stability. In addition, multiyear groundwater level records and subsequent backfilling volumes provided by the mine are indispensable for a comprehensive assessment of surface subsidence.

2.3. Subsidence Monitoring and Prediction Model Development

The temporal evolution of surface subsidence in the mine district is jointly driven by groundwater level fluctuations, mining-induced disturbances, and seasonal forcing, and consequently exhibits slow, nonlinear, and seasonally phase-shifted behavior. Conventional statistical formulations, such as linear regression and autoregressive integrated moving average, often underfit short samples and nonlinear and time-varying sequences. Deep learning can characterize complex temporal dependencies, yet model capacity and hyperparameter sensitivity readily cause small-sample overfitting and instability.

Accordingly, we propose a Bayesian optimization-driven CNN–LSTM hybrid framework (BO–CNN–LSTM). At the monthly temporal scale, the subsidence sequence is modeled through differencing and recursive learning, while robust configurations under constrained sample availability are obtained by automatic hyperparameter search. In parallel, lagged groundwater level to subsidence correlation analysis, together with significance testing, supplies mechanistic statistical evidence for the predictive formulation.

In this study, the BO–CNN–LSTM operates on multivariate monthly time series rather than raw SAR images. For each PS point, we extract a line-of-sight subsidence series from September 2020 to January 2024 and encode each month t as a $7-$dimensional feature vector x_t consisting of the detrended PS-InSAR displacement residual Δy_t (mm) and three sine–cosine pairs that encode annual, semi-annual, and quarterly cycles. In experiments with exogenous drivers, x_t is further augmented by lagged groundwater level and backfilling indicators selected via a simple correlation scan. Over the monitoring period, cumulative subsidence at points A–E ranges from 0 to about -210 mm, while groundwater level varies between roughly 40.8 and 42.5 m; monthly backfilling intensity spans the operator-reported range of approximately 100–660 units. Thus, the network is trained on bounded but clearly non-stationary deformation series, together with hydrological and backfilling signals of realistic magnitude, and this feature representation forms the input to the convolutional and recurrent branches detailed in the following subsections.

2.3.1. PS-InSAR Technology

PS-InSAR achieves millimeter-scale, time-series monitoring of ground deformation by identifying persistent scatterers and unwrapping phase, thereby overcoming the susceptibility of conventional InSAR to spatiotemporal decorrelation. In this study, PS-InSAR is used to monitor surface settlement above the mined-out voids and to produce post-extraction subsidence maps of the mining district. We analyze the spatial patterns and distributional characteristics of settlement to refine subdomains for subsequent numerical

simulation and to delineate priority zones for backfilling. The overall workflow of the BO–CNN–LSTM model for surface subsidence prediction is summarized in Figure 2.

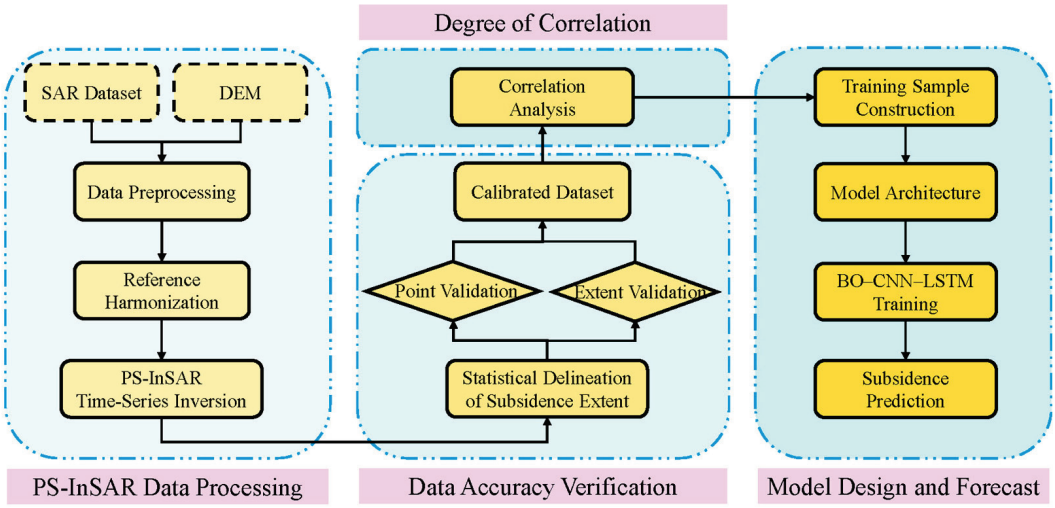


Figure 2. Workflow of the BO-CNN–LSTM model for surface subsidence prediction.

PS-InSAR raster is baseline-corrected per scene to remove arbitrary vertical offsets and to enforce input-level translation invariance before tiling into the CNN branch. Local panel-centric coordinates are used so that the centerline becomes the natural reflection axis for subsequent analysis.

Forty-three SAR scenes from September 2020 to December 2024 were subjected to preprocessing in ENVI 5.6. The geocoded stack was then processed with the PS-InSAR workflow in the Sarscape 5.6 module through the five-stage pipeline illustrated in Figure 3.

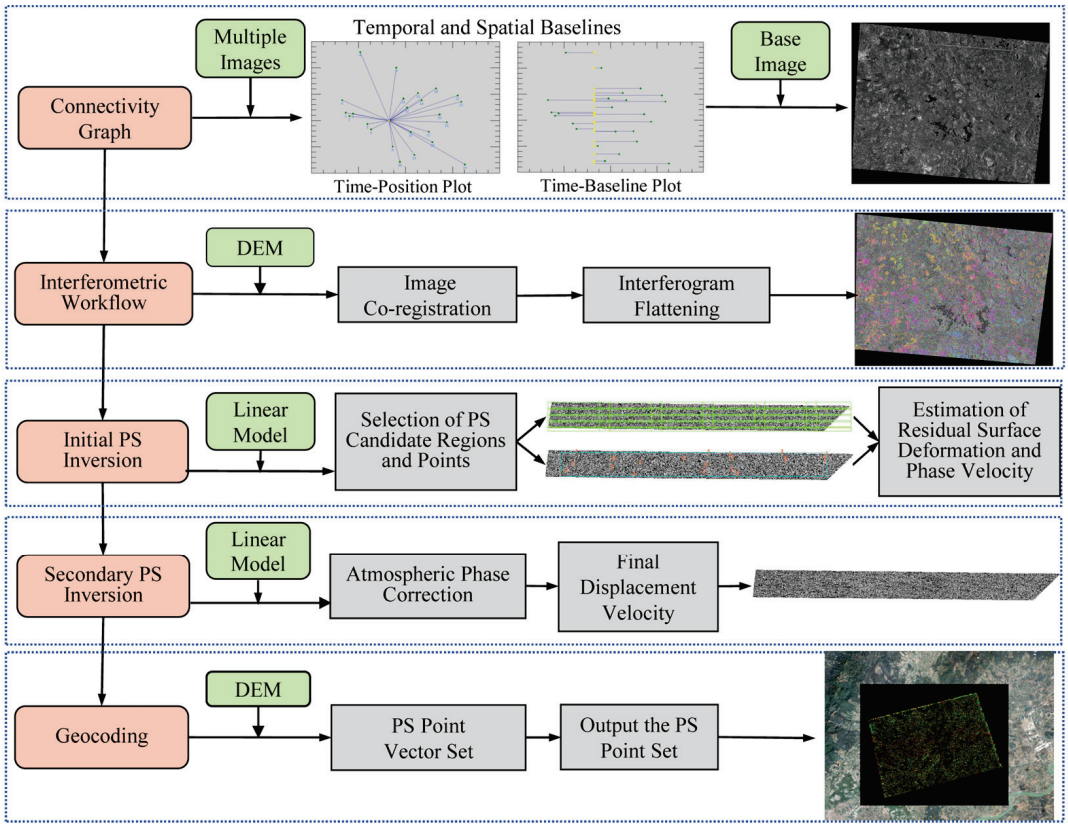


Figure 3. PS-InSAR data processing workflow.

2.3.2. Numerical Modeling

This study adopts an integrative strategy that couples wide-area PS-InSAR ground deformation monitoring with the explicit finite-difference formulation implemented in FLAC^{3D}, thereby constructing a bidirectional verification framework that links the InSAR-derived deformation field with the FLAC^{3D} stress field and plastic zone evolution. During numerical model construction, the FLAC^{3D} mesh size is adjusted according to the spatial resolution of the measured deformation, ensuring that the characteristic element dimension is not greater than the deformation wavelength.

To provide a reproducible, first-order consistency check against the PS-InSAR–based forecasts, the FLAC^{3D} configuration intentionally adopts a quasi-static, drained, small-strain elastoplastic setting with homogenized stratigraphy and simplified goaf/backfill zones, calibrated to available geotechnical tests and site records. Within this scope, several factors are excluded to avoid parameter non-identifiability and uncontrolled uncertainty: (i) fully coupled hydro-mechanical processes, (ii) time-dependent rheology of weak strata and unconsolidated backfill, (iii) explicit discrete-fracture/fault networks and bedding/joint anisotropy beyond an equivalent continuum, (iv) excavation sequencing with detailed face-advance transients and blasting/dynamic loads, (v) thermal effects and thermo-hydro-mechanical couplings, and (vi) spatially variable groundwater recharge/drainage boundaries and seasonal pumping schedules.

Incorporating these processes would require dense hydrogeological instrumentation, fracture-network mapping, and lab rheology at scale; time-stamped operational logs of advance and backfilling; and substantially higher computational budgets for coupled solvers—each of which is infeasible under the current data and time constraints and would likely introduce more degrees of freedom than can be robustly constrained. Consequently, the present FLAC^{3D} results are positioned as scenario-level, quasi-static benchmarks: they are applicable to shallow, slowly evolving goafs with moderate groundwater fluctuations and without strong dynamic excitation, and they are not intended to predict short-term responses dominated by rapid pore-pressure transients, pronounced anisotropy along mapped faults, creep-controlled settlement, or blasting-induced dynamics.

In practical simulations, the rock mass commonly contains multiple joint and fracture sets and is subject to numerous external influences. Accordingly, the present model is established under the following assumptions:

(1) The rock mass is treated as a homogeneous and isotropic continuous medium with uniform properties in all directions;
(2) Only gravitational loading is considered, while external loads and the initial in situ stress field are neglected;
(3) Influences from development openings such as drifts, shafts, and inclines are disregarded;
(4) Hydrogeological effects, including groundwater and structural planes, are not considered.

In the study district, the ground surface is predominantly mantled by a Quaternary soil cover with a thickness of approximately 0 to 2 m, with only limited bedrock outcrops. The stratigraphy is therefore idealized as horizontal, and the upper boundary of the model is assigned an elevation of +42 m. The three-dimensional configuration of the mined-out voids is illustrated in Figure 4.

The computational domain measures 1500 m by 850 m by 300 m, and the aforementioned mechanical parameters for the saturated rock mass are adopted. The bottom, front, back, left, and right boundaries are constrained with fixed conditions, whereas the top boundary is assigned as traction-free. The strength criterion follows the Mohr–Coulomb

yield failure criterion. Before the excavation phase, the displacement and velocity fields are cleared to zero to ensure that all simulated deformation originates from mining activities.

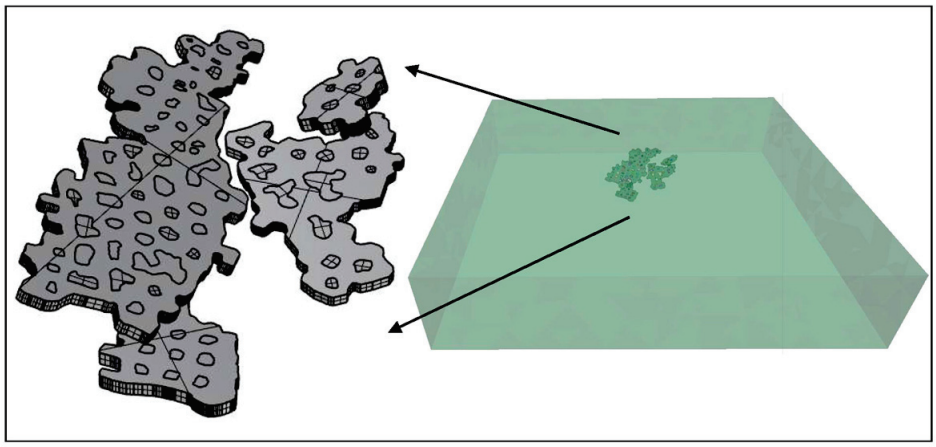


Figure 4. Three-dimensional model of the goaf.

According to the simulation outputs, we interrogate the distributions of the maximum principal stress, the minimum principal stress, the maximum shear stress, the horizontal displacement, the vertical displacement, and the plastic zones to characterize the stress, deformation, and plasticity of the roof strata and the pillars. On this basis, the stability of the mined-out voids is evaluated and cross-validated against the subsidence maps.

Selection of the constitutive formulation underpins the numerical analysis. In view of the rock properties, a Mohr–Coulomb constitutive model is employed [39], and the associated failure criterion is given as follows:

$$f = \sigma_1 - \sigma_3 N_\varphi + 2c\sqrt{N_\varphi} \tag{1}$$

with

$$N_\varphi = \frac{1 + \sin\varphi}{1 - \sin\varphi} \tag{2}$$

where f is the failure indicator; σ_1 and σ_3 are the maximum and minimum principal stresses (MPa); φ is the internal friction angle (degrees); and c is the cohesion of the rock mass (MPa). N_φ is a dimensionless stress ratio defined by φ.

2.3.3. Surface Subsidence Prediction Model

1. LSTM

Long short-term memory is an enhanced recurrent neural network architecture whose central concept is to regulate the propagation and renewal of the internal memory state through a gating mechanism, thereby effectively mitigating the vanishing gradient pathology. An LSTM unit comprises three gating structures, namely the forget gate, the input gate, and the output gate.

Let the input vector at time step t be x_t; the previous hidden state and cell state be h_{t-1} and c_{t-1}; and the current hidden state and cell state be h_t and c_t. The gating operations are formulated as follows.

Forget gate:

$$f_t = \sigma\Big(W_f \cdot [h_{t-1}, x_t] + b_f\Big) \tag{3}$$

Input gate:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

$$\widetilde{c_t} = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{5}$$

Candidate update:

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t} \tag{6}$$

Cell update:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{7}$$

Output gate and hidden state:

$$h_t = o_t \odot tanh(c_t) \tag{8}$$

σ denotes the logistic sigmoid activation; ⊙ represents the Hadamard product; W_f, W_i, W_o, W_c are weight matrices; and b_f, b_i, b_o, b_c are bias vectors. A schematic representation of the LSTM structure is provided in the corresponding Figure 5.

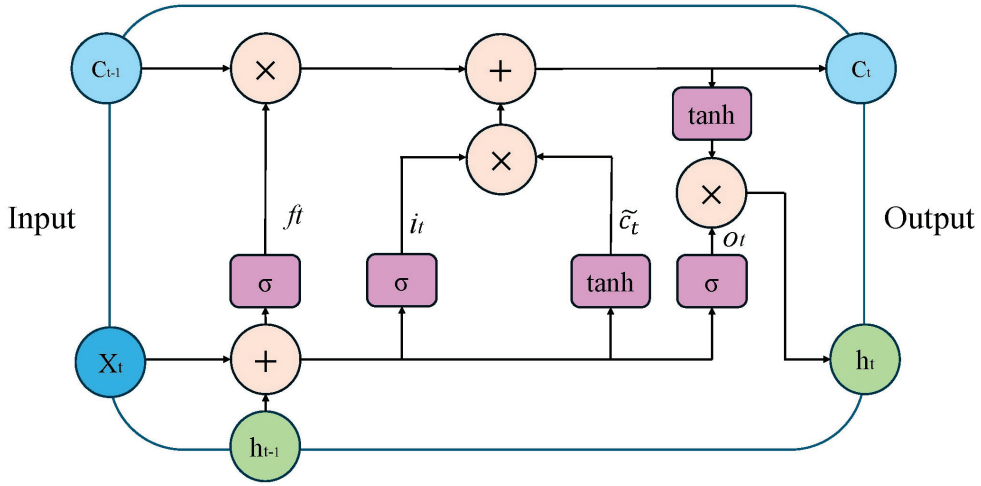


Figure 5. LSTM neural network.

The circled symbol × indicates the Hadamard (⊙) component-wise product, and circled + indicates component-wise summation; each operator corresponds to "⊙" and "+" in the expressions.

For panels with a well-defined centerline, each 2D deformation raster is paired with its mirror image about that axis, and both are used during training. For time series, phase information is preserved while stride-consistent subsequences are created to respect oscillatory regularities captured by LSTMs. These augmentations reduce sampling bias and make the implicit symmetries explicit to the model

2. Bayesian optimization

In complex surface subsidence prediction, the performance of deep learning formulations is highly contingent on the configuration of model hyperparameters. Conventional hyperparameter search procedures, such as grid search and random search, exhibit elevated computational expenditure and modest convergence efficiency. Consequently, this study introduces Bayesian optimization as an efficient black box global optimization approach to automatically identify near-optimal hyperparameter combinations, thereby enhancing predictive accuracy and generalization capability for subsidence modeling.

Bayesian optimization constructs a probabilistic surrogate for the objective function, most commonly a Gaussian process, and is suitable when the objective lacks an analytical expression and each evaluation is computationally expensive. At every iteration, the

method selects a sampling location that is most likely to improve the current global optimum according to the posterior distribution of the surrogate. The procedure comprises two principal components: the surrogate model and the acquisition function.

Let the objective be the minimization of $f(x)$, where $x \in X \subset Rd$ denotes the hyperparameter vector:

$$x^* = \underset{x \in \mathcal{X}}{\operatorname{argmin}} f(x) \tag{9}$$

At iteration t, Bayesian optimization selects the subsequent query location by

$$x_{t+1} = \underset{x \in \mathcal{X}}{\operatorname{argmax}} \alpha(x \mid D_t) \tag{10}$$

where $D_t = \{(x_i, f(x_i))\}_{i=1}^{t}$ represents the extant sampling history and $\alpha(x \mid Dt))$ is the acquisition function that quantifies the prospective utility of evaluating f at x.

Bayesian optimization offers numerous advantages. It is applicable to computationally expensive training procedures, can approach the global optimum with a limited number of evaluations, naturally accommodates parallel and distributed optimization, and integrates uncertainty information into decision-making.

We employ Bayesian optimization to select both the network architecture and the training hyperparameters. A Gaussian-process surrogate with a Matérn-5/2 kernel and automatic relevance determination models the response surface, and an upper confidence bound acquisition function with k = 2.0 balances exploration and exploitation. The objective of optimization is the validation mean absolute error (MAE) computed under a rolling time-series split. During training, we apply early stopping with a patience of 10 epochs, cap the training at 300 epochs, and use a cosine-annealing learning-rate schedule together with gradient clipping. The design begins with 10 Latin hypercube sampling trials to cover the search space, followed by 60 Bayesian optimization iterations, for a total of 70 trials; the complete search space is summarized in Table 2.

Table 2. Hyperparameter search space and prior ranges for Bayesian optimization.

Group	Hyperparameter	Prior/Range
CNN	filters	{16, 32, 64}
	kernel size	{3, 5, 7}
	dilation	{1, 2}
	dropout	Uniform [0.1, 0.5]
LSTM	hidden units	{32, 64, 128}
	layers	{1, 2}
Training	optimizer	{Adam, AdamW, RMSprop}
	learning rate	Log Uniform $[1 \times 10^{-5}, 1 \times 10^{-2}]$
	weight decay (L2)	Log Uniform $[1 \times 10^{-5}, 1 \times 10^{-2}]$
	batch size	{16, 32, 64}
Sequence	look-back window L	{12, 24, 36} (months)
	horizon H	{1, 3, 6, 12} (months)
Exogenous	driver subset	binary mask over {groundwater, backfilling}

Table 2 lists the full hyperparameter search space used in the Bayesian optimization: discrete parameters are treated as categorical variables with uniform priors over the value sets shown, whereas continuous parameters use uniform or log uniform priors over the indicated ranges. Other controls, including the Matérn-5/2 kernel with automatic relevance determination, the UCB parameter k = 2.0, the early-stopping patience

(10 epochs), the maximum epoch cap (300), the cosine-annealing learning-rate schedule, and the gradient-clipping threshold, are fixed across all trials and are not tuned. A complete BO run with 10 warm-up trials and 60 Bayesian iterations is executed on a single workstation, corresponding to a computer budget of a few GPU-hours for the subsidence prediction experiments.

To enhance efficiency and reproducibility, the experiments incorporate pruning, de-duplication, and controlled randomness. Trials are pruned when their interim validation loss at a given epoch exceeds the median of completed trials at the same epoch, thereby terminating unpromising runs early. Duplicate hyperparameter configurations are filtered to avoid repeated evaluations. All trials share a fixed random seed and identical chronological data splits to ensure comparability across iterations. Convergence is declared when the best validation MAE shows no material improvement over successive trials or when the trial budget is exhausted. The final model is selected as the configuration with the lowest validation MAE; in addition, the median and interquartile range of the validation MAE across the ten best configurations are reported to characterize stability.

In this study, it is employed to automatically search the critical hyperparameters of the CNN–LSTM network, including convolution kernel size, hidden layer dimensionality, learning rate, and temporal window length, thereby improving predictive accuracy and generalization for ground subsidence time series modeling.

Beyond these implementation details, our Bayesian optimization scheme is positioned with respect to recent BO-based hyperparameter optimization frameworks for deep learning. In widely used HPO pipelines, such as Bayesian-optimized CNNs, AutoML-style structures, and distribution-aware ensembles, Bayesian optimization is mainly used to tune architectural and training hyperparameters on large, mostly i.i.d. datasets [40–43]. In contrast, the present BO design is explicitly tailored to small-sample, temporally ordered PS-InSAR subsidence series: in addition to the CNN–LSTM architectural and training hyperparameters described above, the search space includes the physically interpretable temporal window length and the inclusion or exclusion of exogenous drivers, and all trials adopt a time-contiguous rolling validation split that respects calendar order. Taken together with the automated pruning and de-duplication strategy described above, this yields a domain-informed, workflow-style BO–CNN–LSTM pipeline that differs from generic BO-HPO schemes and is better aligned with forecasting mining-induced subsidence.

3. Convolutional neural network

A convolutional neural network is a deep learning architecture proficient at extracting spatial attributes from local receptive fields and is widely applied in image recognition, time series modeling, and geoscientific prediction. For surface subsidence forecasting, the deformation sequence can be represented in one-dimensional or two-dimensional form; the CNN module extracts local variation patterns, edge tendencies, and spatial gradient characteristics, which enhance generalization capability and predictive precision.

The input layer consists of subsidence sequences processed by a moving window or by transformed image features. The activation function ReLU introduces nonlinearity and augments representational capacity. Convolutional layers employ multiple kernels that slide across the sequence to capture local features such as localized undulation of ground deformation. In the first stage, the seven-channel input is mapped through a kernel of size

k to C channels, followed by ReLU, then convolved again and followed by ReLU; the sequence length L is preserved to facilitate subsequent LSTM modeling. A permutation operation reorders the tensor to (B, L, C), after which the representation is provided to the LSTM. Figure 6 illustrates the one-dimensional CNN configuration.

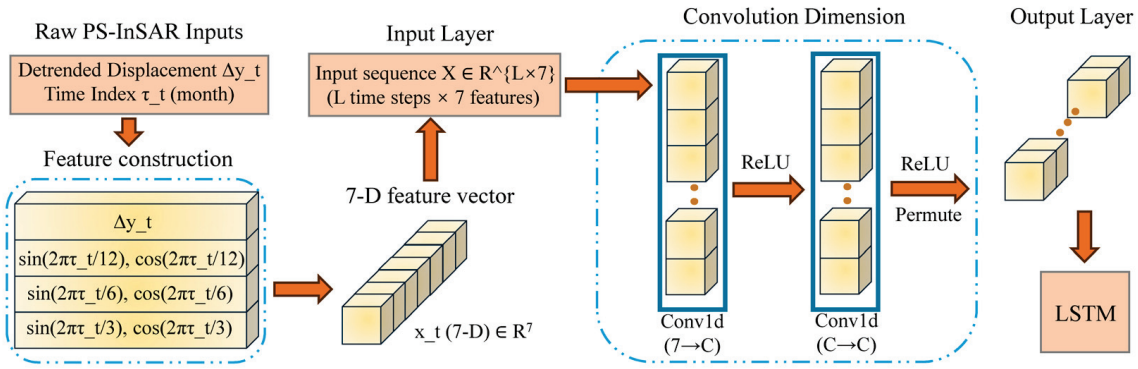


Figure 6. One-dimensional convolutional neural network.

4. Symmetry-Aware Preprocessing

Symmetry-aware preprocessing maps data to panel-centric coordinates, removes long-wavelength baselines, and augments each sample with its mirror counterpart so that axial and mirror relations are explicit to the model. For the Sentinel-1A PS-InSAR displacement time series of shallow, grouped goafs, we adopt a symmetry-aware preprocessing pipeline that performs geometric canonicalization, long-wavelength detrending, and mirror augmentation prior to ingestion by the CNN spatial branch of the BO–CNN–LSTM predictor. The goal is to render inputs comparable across different goafs while remaining robust to variations in orientation and shape.

To align the spatial layout of each goaf/workface polygon and reduce inter-scene rotational discrepancies, a local reference frame is centered on the goaf. The polygon centroid is located, and principal directions are computed from the vertex distribution, enabling orientation alignment and scale normalization so that subsequent detrending and learning proceed in a consistent coordinate system for mining-induced surface-subsidence prediction.

The reflection axis used for mirroring is defined by the dominant spatial organization of each goaf panel to avoid implausible symmetrical projections. For compact, near-elliptical goafs, the axis is taken as the first principal direction through the centroid, representing the primary extension along the extraction direction. For irregular or multi-lobed geometries common in complex goaf clusters, the polygon's medial axis is extracted, and piecewise local axes are defined by the tangent at the nearest skeleton point. When the ratio of the second to the first principal eigenvalue reaches or exceeds 0.35, or when a multi-lobed morphology is detected, the algorithm switches from a single global principal axis to the skeleton-based piecewise scheme, thereby avoiding unrealistic mirroring in highly asymmetric goafs.

Mirroring is then applied by reflecting coordinates about the selected axis, producing a symmetric copy of the detrended PS-InSAR residual field; in the piecewise case, feathering weights are applied at segment junctions to preserve continuity. Coordinates are subsequently rescaled to the unit square $[0, 1] \times [0, 1]$, and the residuals are standardized before entering the CNN spatial branch of the BO-optimized CNN–LSTM. Groundwater level and goaf backfilling intensity are concatenated as exogenous drivers along the temporal dimension into the LSTM branch. This protocol enforces symmetry-consistent inputs and enables faithful reproduction of the preprocessing for mining-induced subsidence across different implementations.

5. BO–CNN–LSTM Architecture

When combined with an LSTM, a convolutional neural network effectively extracts multiscale spatial characteristics from surface subsidence observations, decreases the number of trainable parameters, and improves computational efficiency. The joint CNN and

LSTM formulation simultaneously models spatial structure and temporal evolution and is amenable to Bayesian optimization for automatic hyperparameter selection to further enhance predictive accuracy.

Accordingly, to capture both the spatial variation patterns and the temporal evolution of surface subsidence, we construct a Bayesian optimization plus CNN–LSTM architecture for surface subsidence prediction (BO–CNN–LSTM for SSP), as depicted in Figure 7. The CNN module functions as a spatial feature encoder; the LSTM module serves as a temporal sequence modeler that discovers latent temporal dependencies; and the BO module operates as an automatic hyperparameter tuner that optimizes the performance of the entire CNN–LSTM system.

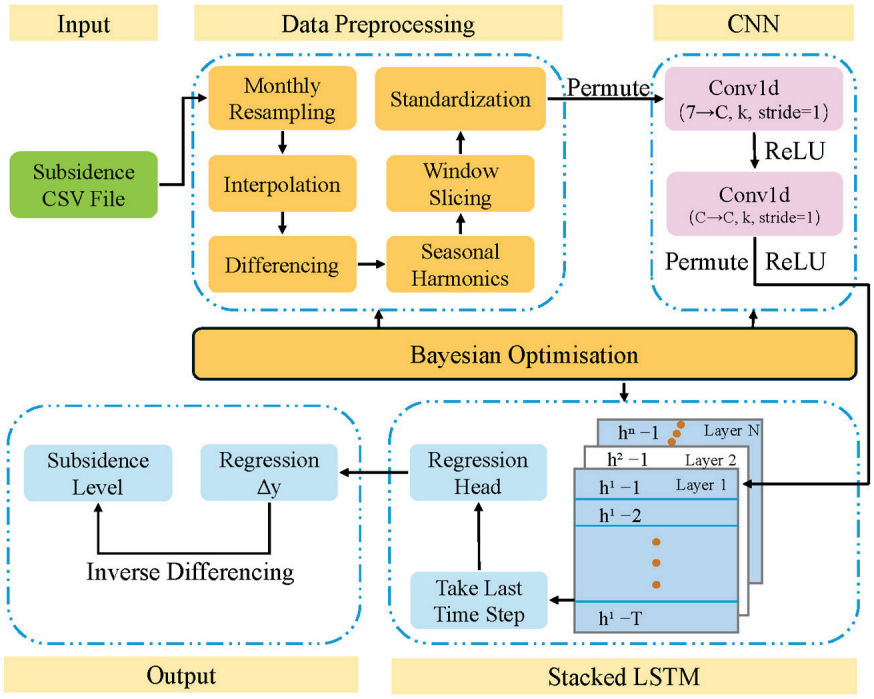


Figure 7. BO-CNN–LSTM architecture for surface subsidence prediction.

Raw Sentinel-1A PS-InSAR displacement stacks undergo quality control with a coherence mask, removal of long-wavelength baselines, light smoothing, resampling to a fixed panel-centric grid, and z-score standardization. Groundwater level and backfilling volume are time-aligned to the PS-InSAR cadence, gap-filled by short linear interpolation, standardized, and concatenated as exogenous channels. The dataset is split chronologically into training, validation, and test windows; a symmetry-aware augmentation mirrors each panel about its principal axis, and the resulting tensors are fed to the CNN spatial encoder and the LSTM temporal model.

Building on these standardized inputs, this study integrates symmetry-aware preprocessing, a translation-equivariant CNN, and a time-translation LSTM with Bayesian optimization to build a reliable and efficient forecasting pipeline for mine-induced subsidence. First, panel-centric coordinates are constructed, and scene baselines are removed so that spatial shifts no longer dominate the input distribution, after which PS InSAR rasters are tiled into patches and paired with their mirror images about the centerline to make axial and mirror relations explicit to the model. Second, the architecture is organized so that the CNN learns spatial patterns under translation equivariance, and the LSTM captures temporal regularities under time-translation symmetry, with consistent stride and periodicity settings across temporal kernels to stabilize long-sequence learning. Third, Bayesian optimization forms a surrogate from previous evaluations and guides a sample-efficient hy-

perparameter search under a fixed training protocol that always includes the mirror-paired augmentation; during the search, we tie regularization strengths across mirror-paired CNN blocks and restrict candidate strides to values that preserve temporal coherence. Finally, models are trained with early stopping and evaluated by root mean square error, mean absolute error, and coefficient of determination on held-out data in order to ensure both accuracy and computational economy.

We adopt a time-consistent sample partitioning strategy: the test window is treated as a complete holdout set, and all history earlier than the test window is used for training; hyperparameters are selected with a contiguous validation window immediately preceding the test window, after which the model is retrained on the entire history, and a single final evaluation and cartographic rendering are conducted only on the test window. This design eliminates temporal leakage induced by random partitions and yields an assessment that is more faithful to real-world deployment.

3. Results

3.1. PS-InSAR Monitoring Results and Accuracy Assessment

The permanent scatterer point targets produced by PS-InSAR encapsulate the subsidence information across the study district and clearly delineate the zones influenced by underground extraction. To visualize the affected extent with greater clarity, the region of interest was constrained by the mapped distribution of the goafs, and Kriging interpolation [44] was applied to the PS point set to derive the post-excavation surface-subsidence map, as shown in Figure 8a.

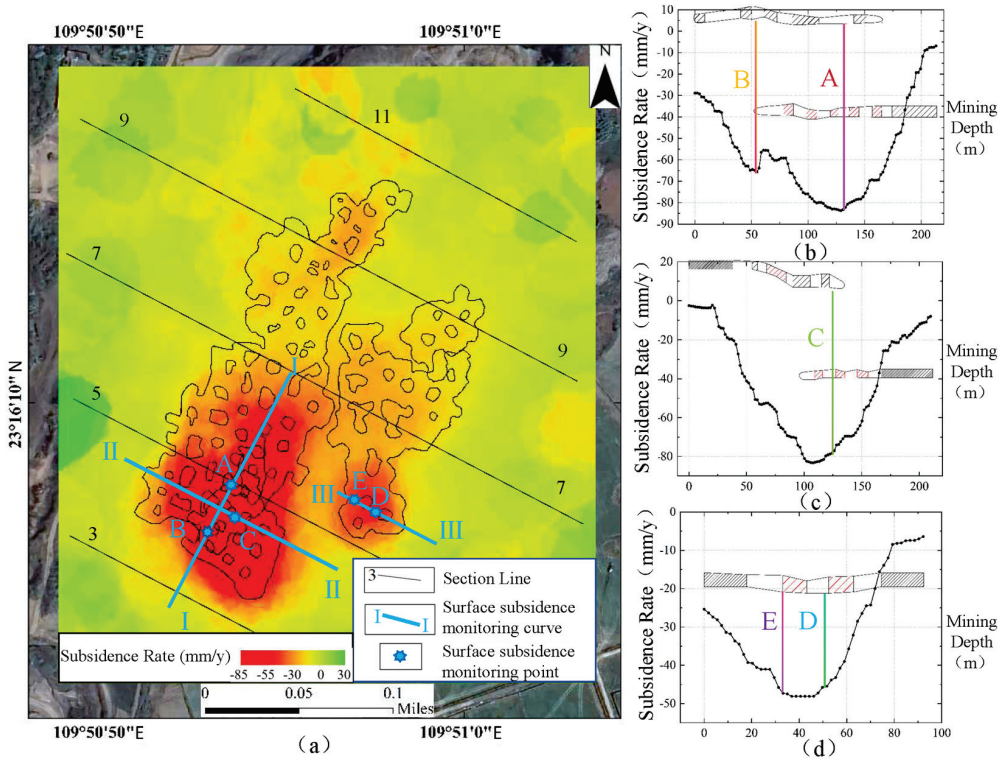


Figure 8. Surface subsidence over the eastern mining section derived from PS-InSAR. (**a**) Kriging interpolation of permanent-scatterer point targets with transects I–III and observation sites A–E indicated. (**b**) Subsidence profile along transect I with pillar lateral view positioned at the corresponding location. (**c**) Subsidence profile along transect II. (**d**) Subsidence profile along transect III. Shaded diagonal areas in (**b**–**d**) indicate the horizontal projection of the underlying goaf, and PS points A–E in (**a**) are mapped to the corresponding locations in the profiles (**b**–**d**), where their subsidence rates relative to the goaf can be read.

By June 2023, the results indicate extensive deformation. North of Exploration Line 7, the subsidence velocity is modest, and deformation is inconspicuous, whereas south of Line 7, the velocity is elevated. Pronounced settlement occurs in the southern sector of No. 1 Goaf and across No. 4 Goaf, with the subsidence center located near the middle of No. 4 Goaf and a maximum rate approaching 85.6 mm yr^{-1}. A secondary zone of notable settlement appears south of No. 3 Goaf, with the center situated in the southern central portion of that goaf and a maximum rate approaching 50 mm yr^{-1}.

Symmetry-augmented training consistently reduces errors in regions where the subsidence bowl is near-axially symmetric and leaves performance unchanged elsewhere. Conversely, the largest residuals concentrate along faults and backfilling boundaries, precisely where symmetry breaks, revealing a physically meaningful failure mode. The spatial pattern indicates that asymmetry is not merely noise but encodes structure-controlled deformation that bounds achievable forecast accuracy.

To investigate the mining impact comprehensively, surface-subsidence profiles were extracted over the zones of pronounced deformation in accordance with the spatial distribution characteristics, and the subsidence behavior was analyzed. The selected transects and the corresponding subsidence curves are presented in Figure 8a–d; Figure 8b–d insert lateral views of the pillars at their actual positions. Red pillars denote unstable pillars identified using the Bieniawski pillar-strength design formulation [45]. Observation sites A–E are located within the red zone of significant settlement.

1. Analysis of transects II and II–II

From Figure 8b,c, subsidence morphology along I–I and II–II is governed by pillar stability, with the principal maximum located in central No. 4 Goaf. Specifically, I–I exhibits a W-type profile with peaks of 67.2 and 85.6 mm yr−1, whereas II–II displays a V-type profile. The I–I curve shows attenuation after the first crest, while II–II does not, indicating that higher pillar competence in upper goafs suppresses upward transmission of deformation.

2. Analysis of transect III–III

From Figure 8d, the III–III section records moderate deformation constrained by structural confinement. In particular, the profile rises and then declines, and the subsidence funnel spatially coincides with unstable pillars; the limited roof span, together with abutment confinement, restricts vertical settlement magnitude.

3. Temporal evolution assessment of surface subsidence

Based on the foregoing analysis, pronounced settlement is concentrated within No. 4 Goaf and the southern sector of No. 3 Goaf. Monitoring sites were therefore selected along the subsidence profiles, and pre-backfilling records from 2023 were employed for time-series evaluation of surface-subsidence tendencies. The rates at the monitoring sites are presented in Figure 9.

Figure 9a shows that within No. 4 Goaf, the subsidence rates at sites A, B, and C increase from a gentle stage to an intensified stage and remain elevated by the end of the monitoring period. The principal cause is instability of the central pillar that induces stress concentration; the roof and the overburden lose effective support, and settlement continues to accelerate. During stress redistribution, a portion of the static load transfers to adjacent pillars, producing synchronous growth of settlement there. The rates can decline only after the system reattains mechanical equilibrium. If multiple pillars are simultaneously unstable, external disturbance may trigger a cascading failure process and propagate to the upper goafs; therefore, No. 4 Goaf requires prioritized remediation.

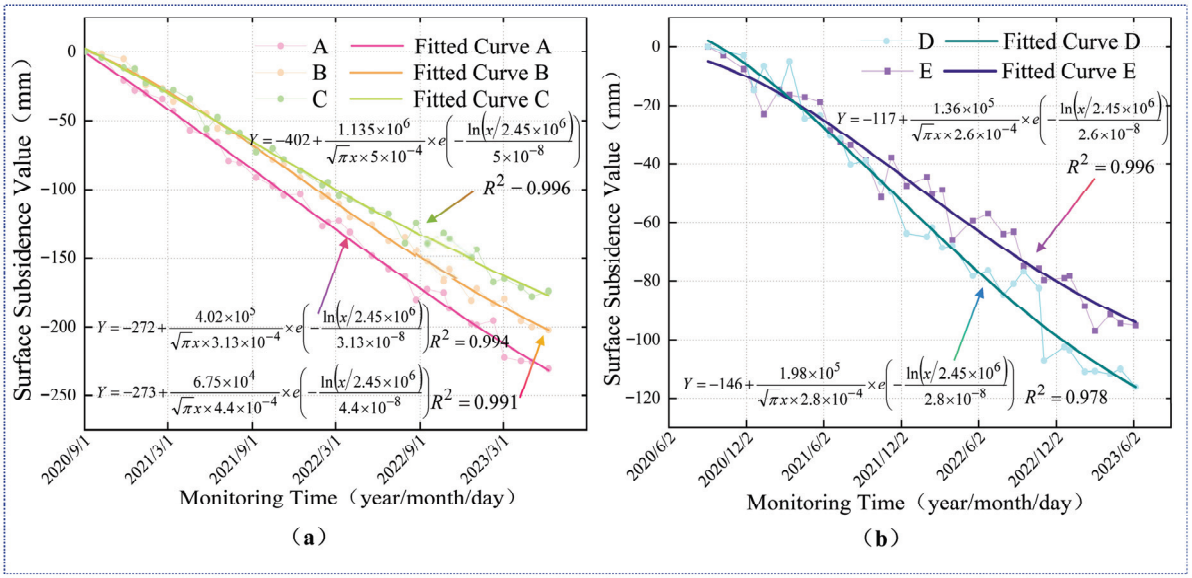


Figure 9. Temporal evolution of surface subsidence at monitoring sites; (**a**) Subsidence time series for monitoring sites within No. 4 Goaf. (**b**) Subsidence time series for monitoring sites within No. 3 Goaf.

Figure 9b indicates that within No. 3 Goaf, the subsidence at sites D and E first increases and then decreases with a comparatively small overall magnitude. Three southern pillars develop local plastic zones, but the limited roof span and confinement by abutment rock allow partial load sharing; after stress redistribution, the settlement attenuates gradually.

3.1.1. Accuracy Assessment of PS-InSAR Monitoring Points

The quality and abundance of permanent scatterer targets generated by PS-InSAR depend on the selection of the output threshold. This threshold is positively associated with monitoring coherence and metric accuracy and inversely associated with the count of permanent scatterer targets. To balance quality and spatial density, the threshold in this study was set to 0.60.

For accuracy evaluation, numerous investigations compare InSAR-derived displacements with terrestrial leveling measurements, GPS, Global Navigation Satellite System observations, Differential InSAR, and Small Baseline Subset InSAR, and examine the discrepancies to verify monitoring precision; close agreement with small errors indicates feasible reliability [46,47]. To ensure accuracy and robustness here, a neighboring point validation strategy was adopted [48], as illustrated in Figure 10b. A level instrument was employed, and for each benchmark, the mean value of permanent scatterer targets within a prescribed radius about the benchmark was computed and compared. According to the radar acquisition timeline, sixteen leveling benchmarks within the mining district were collected from March 2021 through March 2022; for each benchmark, adjacent permanent scatterer targets were selected and averaged for comparison. The residual error statistics are presented in Figure 10c.

Overall, PS-InSAR and leveling observations exhibit strong concordance, with a pronounced deviation only at station 5J1. Figure 10 shows that 5J1 attains the largest discrepancy, 17.71 mm. Field investigation indicates that the site lies at a road intersection where passing haulage vehicles induce dust and gravel accumulation that slightly elevates the pavement, yielding positive PS-derived displacement, whereas the roadside benchmark is less influenced by traffic; the mismatch of reference surfaces thereby amplifies the error. The computations indicate a minimum absolute difference between leveling and

PS estimates of 0.046 mm and a root mean square error of 2.4 mm, demonstrating the feasibility of employing PS-InSAR for surface-subsidence monitoring in the mining district.

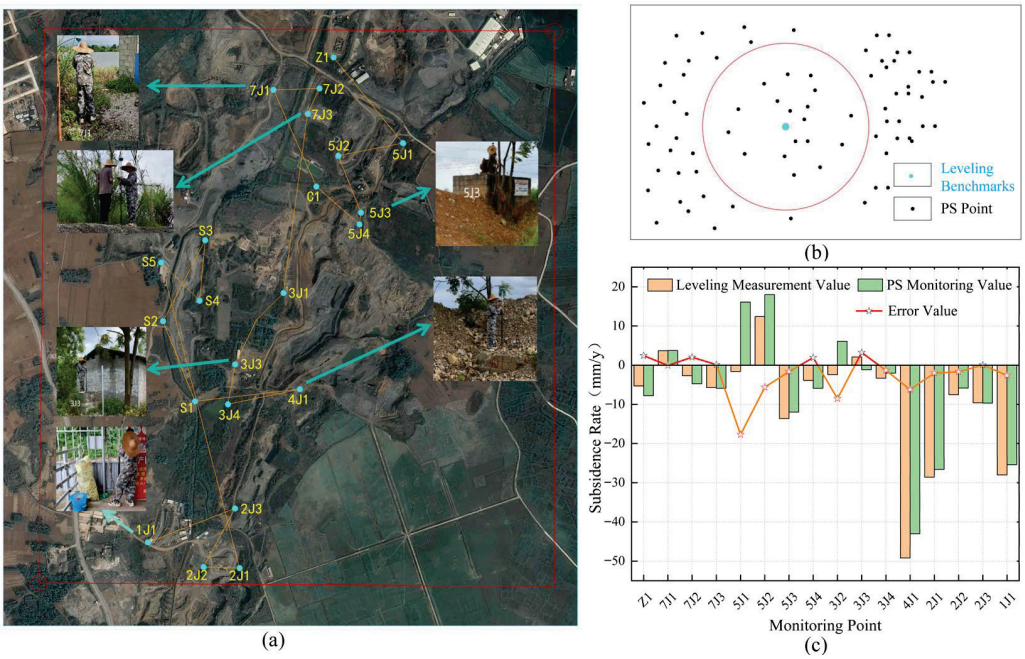


Figure 10. Spatial distribution of leveling benchmarks and verification of monitoring accuracy; (**a**) Benchmark distribution within the mining district. (**b**) Neighboring point validation principle. (**c**) Monitoring error validation.

3.1.2. Accuracy Assessment of the PS-InSAR Monitoring Extent

1. FLAC^{3D} numerical computation

During the numerical simulation, displacement constraint boundary conditions were applied to the four lateral boundaries and the basal boundary, and gravitational acceleration of 9.81 m s^{-2} was imposed in the Z direction to represent the self-weight of the rock mass. The results show a stress field that increases progressively with depth and evolves toward a stable configuration. Owing to the markedly greater bulk unit weight of the orebody relative to the surrounding rock, the normal stress rises more rapidly and exhibits a stratified, nonuniform distribution; the maximum principal stress reaches 7.579 MPa.

According to the numerical simulation results, Figure 11a indicates that horizontal displacements in the roofs, pillars, and surrounding rock mass of No. 1, 2, and 3 Goaf are minor, on the order of ±10 mm, with limited influence on overall stability. In contrast, Figure 11b shows pronounced vertical displacement, with No. 4 Goaf constituting the central nucleus of vertical movement and the impact propagating upward toward No. 1 Goaf, thereby forming a distinct settlement gradient.

2. Validation of the monitoring extent accuracy.

PS-InSAR monitoring and the numerical simulation exhibit strong spatial concordance. North of Line 7, the subsidence amplitudes are minor, whereas pronounced subsidence funnels develop in the southern sectors of No. 4 Goaf and No. 3 Goaf. Nevertheless, a substantive discrepancy exists in the deformation magnitudes: the PS-InSAR maximum subsidence rate attains 85.6 mm yr^{-1}, while the numerical simulation yields a maximum of 98.16 mm yr^{-1}. The divergence arises principally from simplifications in the numerical representation of the rock mass structural plane network, for example, omission of the actual distribution of joint-set density; incomplete reproduction of mining-induced disturbances, such as dynamic loading associated with blasting vibration velocities exceeding 1.5 cm s^{-1};

and the spatiotemporal variability of hydrogeological conditions. Notwithstanding the magnitude difference, both approaches identify the key settlement zones with high consistency, which substantiates the feasibility and precision of PS-InSAR for surface subsidence monitoring in the mining district.

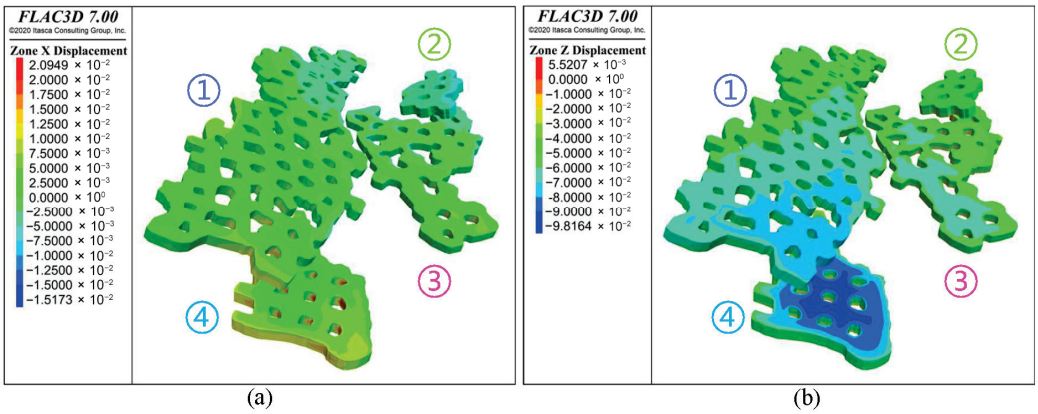


Figure 11. Displacement fields across the goaf domain: (**a**) horizontal displacement field and (**b**) vertical displacement field. The labels ①–④ denote the four goaf IDs.

3.2. Analysis of Influencing Factors

Hydrometeorological controls, particularly precipitation and groundwater level fluctuations, exert a decisive influence on subsidence behavior. During the dry season, a depressed groundwater table reduces moisture content, induces shrinkage of soil and rock mass, and aggravates surface settlement. During the rainy season, an elevated water table produces saturation, attenuates shear strength, and may generate upward heave. Consequently, groundwater fluctuation constitutes a critical stability risk for the lead–zinc mine. Excessively high groundwater promotes swelling and heave, whereas an excessively low groundwater level enhances consolidation but increases brittleness, leading to cracking and structural damage.

These physical mechanisms are consistent with monitoring results. Figure 12 plots the monthly groundwater level together with the cumulative PS-InSAR subsidence at five representative points A–E located in No. 3 and No. 4 Goafs. The groundwater table exhibits seasonal fluctuations with an overall drawdown on the order of one meter, while the cumulative subsidence curves increase almost monotonically. Periods when the groundwater level is relatively low coincide with steeper subsidence trajectories, indicating accelerated settlement, whereas intervals with a partially recovered water table correspond to a temporary reduction in subsidence rates. Months with engineered backfilling are highlighted by yellow bars and show a pronounced flattening of the subsidence curves, especially at points A and B, implying that the improved support conditions weaken the hydrologically driven component of settlement.

Backfilling of mined-out voids is another principal control on surface deformation. Guided by PS-InSAR observations, the mine implemented progressive backfilling in No. 3 Goaf and No. 4 Goaf from June 2023 to March 2024; the areal extent of backfilling and its influence on the 2024 deformation field are illustrated in Figure 13a. Monthly backfilling volumes over this interval were compiled and supplied to the subsequent model training and correlation analysis.

It should be noted that in Figure 13b, the maximum settlement value of −25.2 mm is referenced to January 2024. The deformation in March 2024 within Goafs 3 and 4 is markedly reduced relative to December 2023, and Figure 13a indicates a post-backfilling reduction in subsidence rates, confirming the effectiveness of the backfilling intervention.

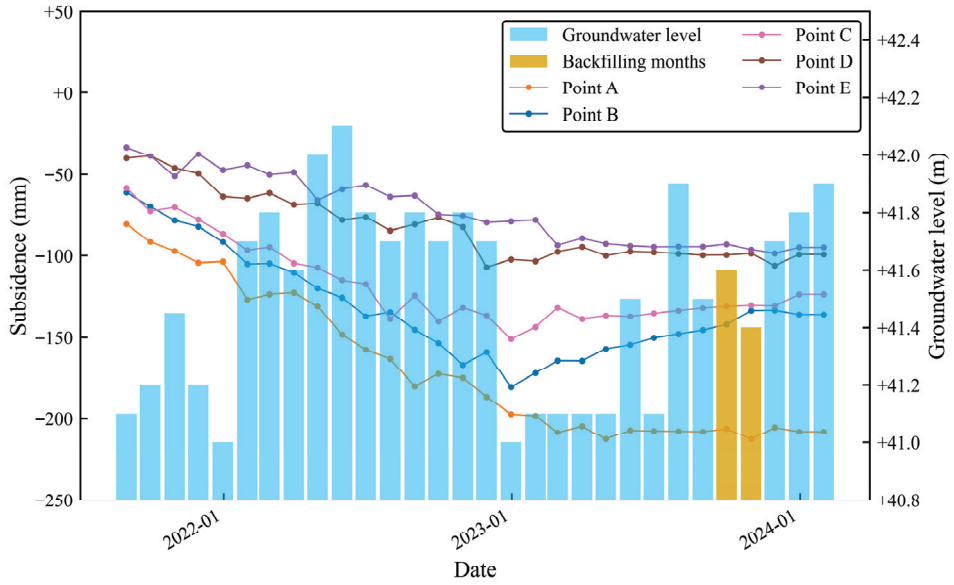


Figure 12. Time series of subsidence at representative points A–E and groundwater level near No. 3 and No. 4 Goafs. Blue bars denote monthly groundwater levels, and yellow bars highlight months with backfilling operations.

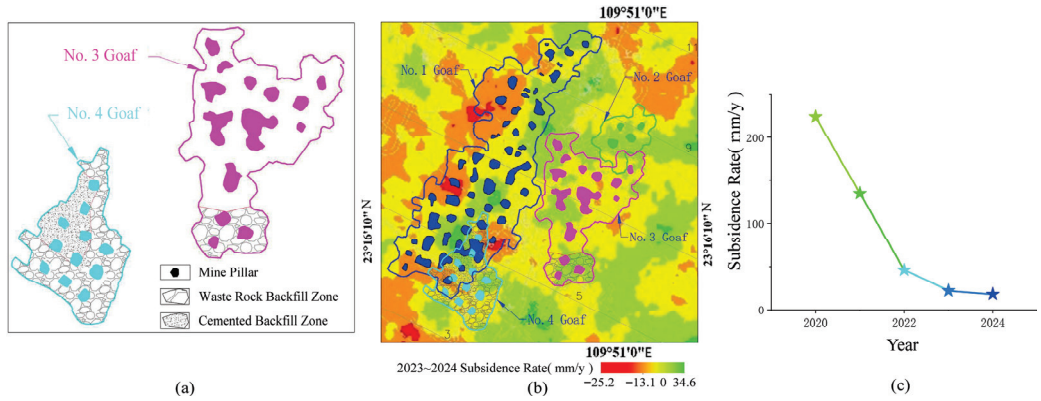


Figure 13. Backfilling information for 2024 across the goaf domain. (**a**) Locations of backfilled sectors. (**b**) Surface subsidence map for 2024. (**c**) Evolution curves of subsidence magnitude and rate for No. 3 and 4 Goaf during 2020–2024.

3.3. BO–CNN–LSTM Prediction Results

3.3.1. Hyperparameter Search and Validation Heatmap

This study develops a subsidence prediction framework that represents the signal through differencing, seasonal harmonics, and a sliding window, combined with Bayesian optimization within a one-dimensional CNN–LSTM architecture, with the objective of improving time-aligned estimation at previously unseen locations. At the model level, the outer loop adaptively enumerates the look-back window length L, while the inner loop employs Optuna-based Bayesian optimization to jointly search network and training hyperparameters. Under a time-aware training and validation partition, the objective is the minimum validation Smooth L1, and robust convergence is obtained with the Adam optimizer and early stopping. This workflow preserves consistency between the physical time scale and the statistical formulation, balances short-term structure with long-term memory, prevents information leakage, and reduces extrapolation error variance, thereby yielding more robust and accurate subsidence predictions.

Bayesian optimization explores the hyperparameter space through a probabilistic surrogate and an acquisition function, enabling efficient search without exhaustive enumeration and improving predictive accuracy. In this work, it tunes the look-back window L and the convolution kernel size k. Figure 14 presents a validation heat map that depicts the distribution of validation losses over combinations of L and k; the color intensity encodes loss magnitude, with darker shading indicating smaller errors.

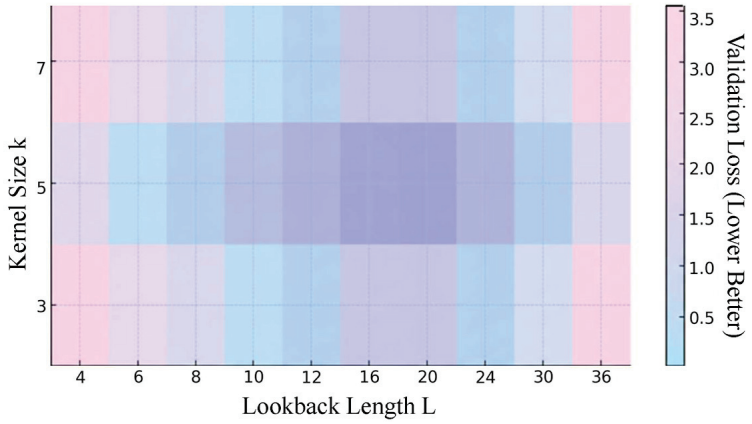


Figure 14. Validation set heat map of loss values.

The results in Figure 14 indicate that L and k are not the only determinants of accuracy, since channel count and hidden dimensionality also exert significant influence. Configurations with a moderate window and a moderate kernel length achieve an effective balance between capturing local patterns and suppressing overfitting, whereas an excessively short window fails to represent the seasonal phase, and excessively long windows or large kernels tend to overfit and reduce generalization capability. Consequently, the experiments adopt an optimized setting of approximately $L \approx 20$ and $k = 5$ to obtain more stable predictive performance.

3.3.2. Training–Validation Loss Curves

We quantify learning dynamics and assess generalization by tracking epoch-wise mean absolute error (MAE) and root mean square error (RMSE) on both the training and validation sets. To ensure transparency and reproducibility, all metrics are computed from raw residuals without smoothing, and the model snapshot retained for downstream inference is the checkpoint with the minimum validation RMSE under an early-stopping protocol. The validation split is strictly time-contiguous and placed immediately before the test window to avoid temporal leakage and to reflect the intended deployment setting. For brevity, Figure 15 reports only the RMSE curves, because the MAE trajectories exhibit an almost identical pattern.

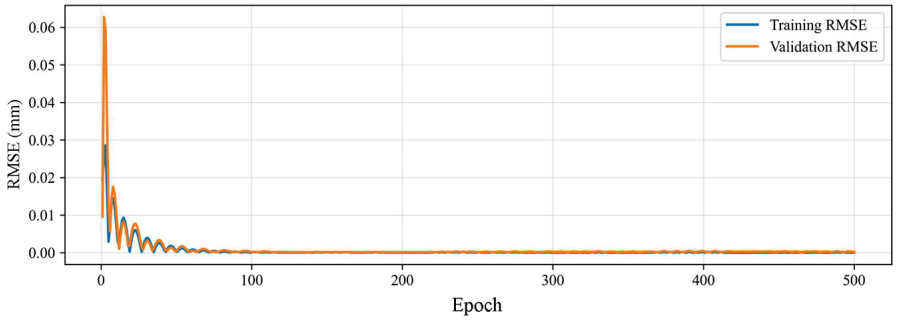


Figure 15. Training dynamics at representative point C.

The curves in Figure 15 display a rapid decay of RMSE during the first 20–30 epochs, followed by a stable plateau, with the validation trajectory closely tracking the training one. The narrow and steady generalization gap indicates well-controlled model capacity and the absence of late-stage divergence, supporting the reliability of the selected checkpoint. These diagnostics substantiate that the optimization produced a stable configuration suitable for the subsequent reconstructions and forward projections of the subsidence time series.

In Figure 15, errors decrease sharply within the first 20–30 epochs and then plateau; the small generalization gap indicates stable convergence without overfitting. The checkpoint with the minimum validation loss is retained for subsequent analyses.

To complement the convergence diagnostics, we next quantify the out-of-sample skill of the symmetry-aware BO–CNN–LSTM. Figure 16a compares PS-InSAR observations with model predictions in the held-out test window using a 1:1 agreement plot with an OLS regression fit and its 95% confidence band. Most points cluster tightly around the 1:1 line, indicating small residuals; the few deviations correspond to locations previously identified as symmetry-breaking zones. Figure 16b summarizes the test-window errors at points A–E using MAE and RMSE. Consistent with the symmetry-aware preprocessing, accuracy is highest where the deformation bowl is near-axially symmetr and modestly degrades in panels exhibiting stronger departures from symmetry. These quantitative checks, together with the training–validation loss curves, support both the fidelity and generalizability of the proposed predictor before we visualize the full spatiotemporal evolution.

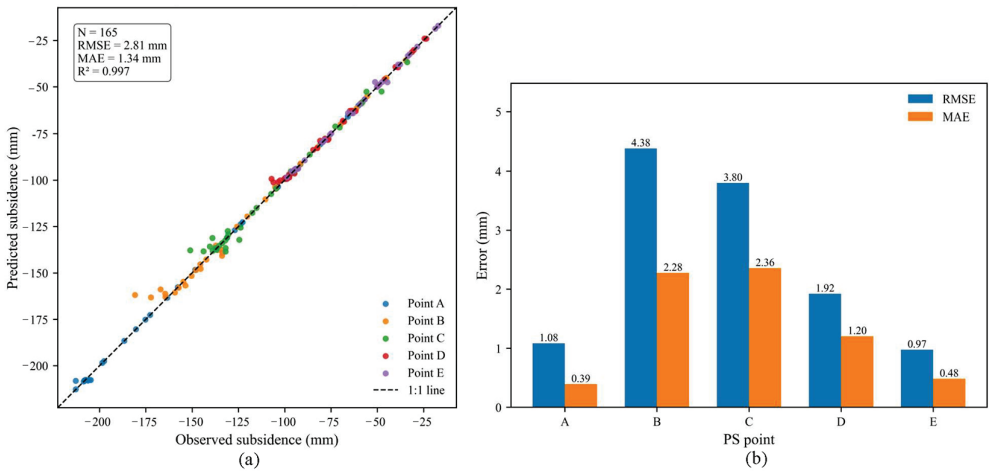


Figure 16. Agreement and point-wise errors in the test window. (**a**) Observed vs. predicted monthly subsidence at points A–E. (**b**) Point-wise MAE and RMSE across A–E in the test window.

Note that the R^2 = 0.997 in Figure 16a quantifies the agreement between BO–CNN–LSTM predictions and PS-InSAR observations. The RMSE = 2.81 mm and MAE = 1.34 mm reported in Figure 16a are computed over all 165 test samples across points A–E; the bars in Figure 16b show the site-wise RMSE and MAE, whose squared and arithmetic means recover the global RMSE and MAE, respectively.

Errors are smallest where axial symmetry is stronger, consistent with the symmetry-aware preprocessing and mirroring augmentation.

3.3.3. Visualization of Subsidence Results

Samples are partitioned by time window. The test window spans September 2020 to January 2024; the training set comprises all antecedent history. An additional ten percent of the training history is reserved as a validation set to select hyperparameters robustly; final training uses the entire history, and both model evaluation and cartographic rendering are performed exclusively within the test window. Model feasibility is examined through

correlation coefficients between influencing factors and both the source sequence and the predicted sequence.

Figure 17 presents, in a unified coordinate system, the subsidence evolution at five monitoring sites: the X-axis denotes date, the Y-axis enumerates sites A–E, and the Z-axis gives subsidence magnitude. The blue curve denotes the PS-InSAR-observed displacement time series, with negative values indicating downward subsidence; the orange curve shows the hindcasts within the held-out test window, while the semi-transparent orange panel marks the test interval and the forecast origin; the green curve depicts the BO–CNN–LSTM forward projections for 2024–2027.

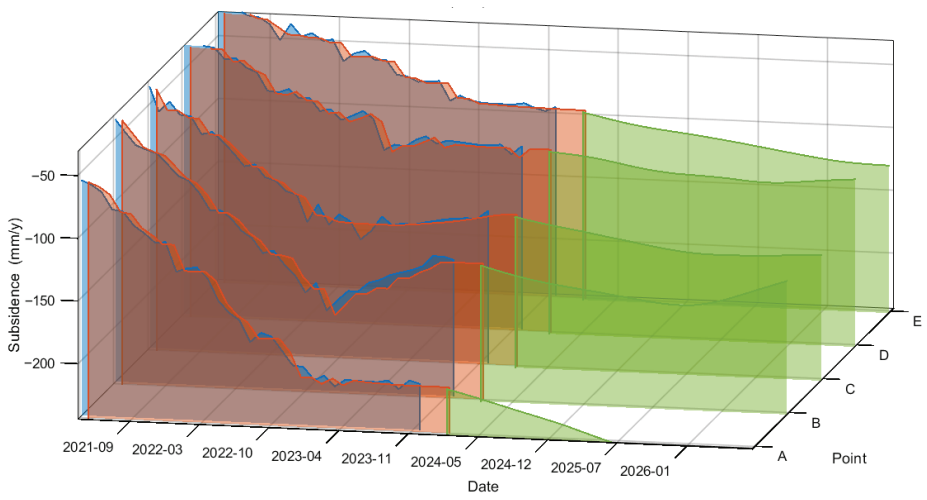


Figure 17. Time–Point–Subsidence 3D waterfall plot for points A–E.

The orange curve is the inversion from the predictive model fitted with CNN–LSTM under Bayesian optimization and subjected to endpoint or shape calibration over the historical interval; the green curve is the future forecast propagated to January 2027 with the orange curve terminus as the boundary condition.

The inversion curve aligns closely with the observations, indicating small residuals and faithful reproduction of amplitude and phase, which supports a credible origin for extrapolation. The inversion and forecast curves connect smoothly with moderate undulation, representing plausible evolution driven by slowly varying backfilling and groundwater seasonality. Overall, all five sites continue to settle, with larger accumulations at A–C and smaller accumulations at D–E, and with varying degrees of deceleration in recent years.

The results indicate accelerating settlement of the overburden prior to September 2023; after backfilling in September 2023, the goaf transitions into a stabilized settlement regime, and during June to December 2023, a rise in groundwater level produces local surface heave. The green forecast from the BO–CNN–LSTM is generally more gradual, suggesting a progressively decreasing settlement rate. In the subsequent step, feasibility is further verified via correlation coefficients between influencing factors and both the source sequence and the predicted sequence.

As shown in Table 3, at all monitoring sites, the correlation between the model output and the two categories of influencing factors—groundwater level and goaf backfilling volume—exceeds that of the raw subsidence sequence, with backfilling volume exhibiting the most pronounced contribution, indicating that the model better preserves the statistical linkage to the driving factors. Itemized results show that the correlation with groundwater level increases from 0.752 to 0.798 to 0.754–0.799, and the correlation with backfilling volume increases from 0.550 to 0.862 to 0.812–0.866, with particularly marked gains at sites A, D, and E. This evidence demonstrates effective denoising, salient feature extraction, and

physical consistency, supporting application to subsidence forecasting and mechanistic interpretation and providing a quantitative basis for risk early warning and remediation decision-making.

Table 3. Correlation coefficients between influencing factors and the observed and predicted series.

Point	Correlation with Groundwater Level		Correlation with Goaf Backfilling Volume	
	Observed	Model	Observed	Model
A	0.785	0.787	0.676	0.812
B	0.788	0.795	0.848	0.861
C	0.798	0.799	0.862	0.866
D	0.773	0.798	0.807	0.863
E	0.752	0.754	0.550	0.855

3.4. Comparative Evaluation of Peer Models

Figure 18 shows four-panel scatter plots comparing model predictions with monthly observations, reveal strong concordance between model predictions and monthly observations, with most points clustering around the 1:1 reference line. In Figure 18, the red markers denote the proposed BO–CNN–LSTM model, while the blue markers denote the three benchmark models used for comparison, so that their errors and dispersion patterns can be directly contrasted. In particular, the proposed BO–CNN–LSTM shows the tightest clustering distributed along the diagonal with minimal dispersion, indicating low random error and negligible overall bias. Autoformer ranks second, displaying only slight spreading during large-magnitude subsidence episodes. LSTM exhibits greater dispersion and a mild underestimation at high subsidence amplitudes, implying residual bias under rapid variations. SARIMA presents the most pronounced dispersion and systematic bias, especially near −200 mm, which reflects difficulty in representing nonstationary and nonlinear deformation.

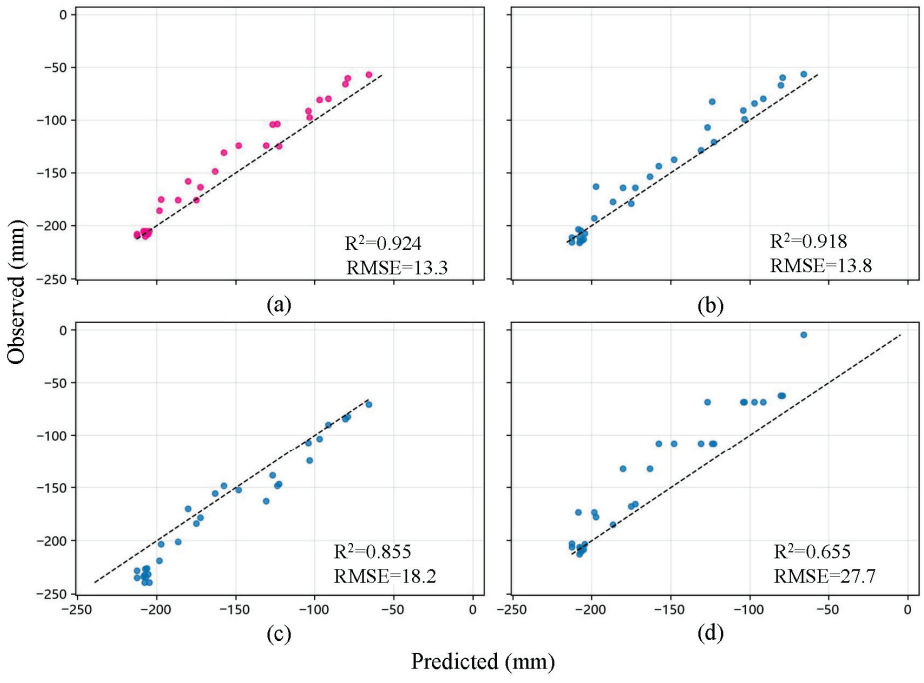


Figure 18. Scatter plot of multiple models. (**a**) Scatter plot of the BO–CNN–LSTM model. (**b**) Scatter plot of the Autoformer model. (**c**) Scatter plot of the single LSTM model. (**d**) Scatter plot of the SARIMA model.

As summarized in Table 4, BO–CNN–LSTM attains the best overall accuracy at site A, achieving the lowest RMSE and MAE, together with the highest R^2. These results suggest that the synergy of Bayesian optimization, convolution-based feature extraction, and gated temporal memory substantially enhances the representation of nonlinear and multiscale spatiotemporal dependencies. The model remains stable with longer input windows while retaining sensitivity to peak-valley transitions in subsidence. Overall, BO-CNN–LSTM demonstrates robust and superior predictive performance and can serve as a reliable instrument for assessing subsidence evolution and supporting engineering early warning.

Table 4. Comparison of prediction results across models.

	MAE	**RMSE**	**sMAPE**	**nRMSE**	**R^2**
BO–CNN–LSTM	10.3	13.3	0.087	0.090	0.924
Autoformer	10.5	13.8	0.087	0.094	0.918
LSTM	15.3	18.2	0.088	0.124	0.855
SARIMA	21.3	27.7	0.224	0.189	0.655

4. Discussion

This study integrates Bayesian optimization with a CNN–LSTM deep architecture for temporal prediction of surface subsidence and interprets deformation through a symmetry-aware lens. Subsidence bowls are often approximately axial or mirror symmetric about the panel centerline, whereas mapped structures, heterogeneous backfilling, and groundwater variations introduce departures from that symmetry. Under the working hypothesis that groundwater level and backfilling intensity act as principal external drivers, a data-informed sequence model that respects spatial and temporal regularities should recover their imprints while improving forecast skill over previously published approaches.

Beyond qualitative plausibility, we examined the drivers quantitatively. Correlation analyses show a positive association between groundwater rise and subsidence deceleration and between backfilling intensity and subsequent stabilization, with short lags consistent with drainage and consolidation timescales. A local sensitivity check indicates monotonic responses in forecasted velocity and a stronger effect in periods of large driver variability; partial-dependence curves remain smooth without spurious oscillation, supporting the use of these drivers as informative exogenous channels. These findings substantiate the modeling choice to condition the CNN–LSTM on groundwater and backfilling while allowing the network to learn nonlinear interactions.

Operationally, Bayesian optimization autonomously searches convolutional and memory unit sizes together with learning-rate and regularization controls, achieving a stable near-optimal configuration with modest search expenditure. Inputs are expressed in panel-centric coordinates after baseline removal and are augmented with mirror counterparts along the centerline so that axial and mirror relations are explicit to the model. The CNN exploits translation equivariance to extract local morphology and abrupt-change signatures, and the LSTM captures long-range dependence consistent with time-translation regularity, while both branches are jointly conditioned on the external drivers. Endpoint anchoring combined with affine calibration yields a natural junction between the fitted trajectory on the test horizon and the forward projection. In the three-dimensional waterfall visualization, the orange reconstruction aligns closely with the blue observations, indicating faithful representation of historical trends and short-term perturbations; the green forecast shows moderate undulation consistent with constrained evolution under external forcing and with controllable extrapolation uncertainty.

Departures from symmetry carry management implications. Where panels remain close to axial or mirror symmetry, alignment improves, and residuals are diffuse, suggesting that standard backfilling and groundwater regulation are adequate. Conversely, concentrated residuals tend to cluster near mapped faults, sharp lithologic contacts, or zones of uneven backfill; these hotspots flag locations where reinforcement is warranted, where backfilling schedules should be intensified or rebalanced, and where local water-level control (e.g., temporary pumping or recharge) can mitigate differential settlement. Tracking symmetry-related indicators through time provides an inexpensive screening tool for prioritizing interventions.

To present a cohesive narrative across data, physics, and learning, we juxtaposed PS-InSAR, FLAC^{3D}, and the predictor. PS-InSAR supplies spatially continuous time series that the model reconstructs with small residuals over the test window; the quasi-static FLAC^{3D} scenarios, calibrated to available geotechnical data, bound the magnitude and shape of expected deformation under typical loads. Consistency is observed: areas of modeled compression coincide with PS-InSAR subsidence bowls and with the CNN-extracted morphology, while zones of simulated stress concentration are collocated with the predictor's larger residuals and the observed departures from symmetry. Within this division of labor, FLAC^{3D} offers physically interpretable bounds and modes of failure, and the data-driven model captures fine-scale, driver-conditioned variability under the observed operational regime.

Limitations remain regarding dependence on data quality and on the availability of external variables. Under distributional shift in environmental conditions or frequent extreme events, generalization may deteriorate; sensitivity to symmetry departures is also evident near structurally controlled boundaries where residuals cluster. Two directions are prioritized. First, introduce probabilistic outputs and uncertainty quantification, and include explicit symmetry metrics and invariance checks together with additional geodetic and hydrogeological priors (e.g., SBAS InSAR, GNSS, and process knowledge) to obtain physically consistent and interpretable predictions. Second, develop cross-site and cross-scenario transfer with online updating to enhance robustness under data sparsity and pronounced nonstationarity, while monitoring how symmetry-related indicators evolve through time. Relative to conventional statistical formulations or single recurrent networks, the combined use of Bayesian optimization, multisource conditioning, and morphology-consistent future construction furnishes a comprehensive scheme suitable for early warning and operations decision-making in complex clusters of goafs.

5. Conclusions

This study proposes a subsidence-prediction framework that integrates PS-InSAR time series with a Bayesian-optimized CNN–LSTM and symmetry-aware preprocessing. Using a shallow cluster of goafs in a lead–zinc mine in Southwest China, Sentinel-1A data from 2020 to 2024 indicate a maximum subsidence rate of 85.6 mm yr^{-1} and an affected area of about 1.8 km^2, with accuracy corroborated by leveling and FLAC^{3D}. Conditioning on groundwater level and backfilling volume, the model reconstructs historical trends and short-term fluctuations and reveals an initial acceleration of settlement followed by attenuation after intervention; the forward projections indicate subsequent stabilization with localized uplift. Gains are strongest where the subsidence bowl is close to axial symmetry, whereas larger residuals near mapped faults and backfilling boundaries delineate zones where departures from symmetry bound attainable accuracy.

The present work has limitations linked to modeling assumptions and data constraints. PS-InSAR coherence, temporal sampling, and noise restrict spatial and temporal resolution; exogenous drivers are limited to groundwater and backfilling, without fully coupled hydro-

mechanical processes. The numerical benchmark adopts a quasi-static setting and does not explicitly represent fracture networks or dynamic loading, and long-horizon forecasts may degrade under distribution shifts. The symmetry perspective is approximate and can be weakened by strong structural controls. These factors define the scope of applicability and motivate targeted extensions.

Future work will extend the framework through three practical directions. First, incorporate richer hydrogeological information—pore-pressure logs, pumping schedules, recharge boundaries—together with SBAS InSAR and GNSS to strengthen physical consistency. Second, deploy probabilistic outputs with calibrated intervals and online updating, enabling near-real-time monitoring, alarm thresholds, and adaptive retraining driven by Bayesian optimization. Third, test cross-site transfer in different lithologies and depths, and operationalize symmetry indicators to guide reinforcement priorities, backfilling scheduling, and groundwater regulation. These steps aim to enhance robustness, interpretability, and the value of the framework for early warning and operations decision-making.

Author Contributions: Conceptualization, X.L. (Xiaoquan Li); Methodology, T.S. and X.L. (Xuefeng Li); Investigation, T.S. and L.Z.; Data curation, T.S. and X.H.; Resources, X.L. (Xuefeng Li) and D.Z.; Validation, L.Z. and X.Y.; Visualization, T.S. and Z.H.; Writing—original draft preparation, T.S.; Writing—review and editing, X.L. (Xiaoquan Li) and X.L. (Xuefeng Li); Supervision, L.Z.; Project administration, X.L. (Xiaoquan Li); Funding acquisition, X.L. (Xiaoquan Li). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Science and Technology Program (Nos. AB25069105) and the Guangxi Emergency Management Joint Innovation Science and Technology Key Project (Nos. 2024GXYJ001). The funders were not involved in the study design, collection, analysis, interpretation of data, the writing of this article, or the decision to submit it for publication.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to confidentiality agreements with the mine operator, raw groundwater logs and backfilling records cannot be made publicly available.

Acknowledgments: The authors gratefully acknowledge the technical support provided by the Tongkeng Mining Branch of Guangxi China Tin Group Co., Ltd.

Conflicts of Interest: Author Linxin Zhang was employed by Guangxi Industrial Design Group Co., Ltd. Author Xuxing Huang was employed by Changsha Institute of Mining Research Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

InSAR	Interferometric Synthetic Aperture Radar
PS-InSAR	Persistent Scatterer InSAR
FLAC^{3D}	Fast Lagrangian Analysis of Continua in 3 Dimensions
BO	Bayesian optimization
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory network
Autoformer	Decomposition Transformers with Auto-Correlation
SARIMA	Seasonal Autoregressive Integrated Moving Average
VV polarization	Vertical transmit and Vertical receive polarization
S1A*_SLC	Sentinel-1A Level-1 SLC (Single Look Complex) product
SRTM1	Shuttle Radar Topography Mission 1-arc-second Global DEM
ENVI 5.6	Environment for Visualizing Images, version 5.6

GPS	Global Positioning System
RMSE	Root Mean Square Error
MAE	Mean Absolute Error

References

1. Dong, L.; Sun, D.; Shu, W.; Li, X. Exploration: Safe and Clean Mining on Earth and Asteroids. *J. Clean. Prod.* **2020**, *257*, 120899. [CrossRef]
2. Fuławka, K.; Stolecki, L.; Szumny, M.; Pytel, W.; Jaśkiewicz-Proć, I.; Jakić, M.; Nöger, M.; Hartlieb, P. Roof Fall Hazard Monitoring and Evaluation—State-of-the-Art Review. *Energies* **2022**, *15*, 8312. [CrossRef]
3. Chen, J.; Tan, Y.; Huang, X.; Fu, J. Research on a Classification Method of Goaf Stability Based on CMS Measurement and the Cloud Matter–Element Model. *Appl. Sci.* **2024**, *14*, 3774. [CrossRef]
4. Zhang, Y.; Lian, X.; Ge, L.; Liu, X.; Du, Z.; Yang, W.; Wu, Y.; Hu, H.; Cai, Y. Surface Subsidence Monitoring Induced by Underground Coal Mining by Combining DInSAR and UAV Photogrammetry. *Remote Sens.* **2022**, *14*, 4711. [CrossRef]
5. Zhang, L.; Gao, P.; Gan, Z.; Wu, W.; Sun, Y.; Zhu, C.; Long, S.; Liu, M.; Peng, H. Surface Subsidence Monitoring of Mining Areas in Hunan Province Based on Sentinel-1A and DS-InSAR. *Sensors* **2023**, *23*, 8146. [CrossRef]
6. Xu, Z.; Xu, W.; Zhou, P.; Zhu, Z.; Zhao, J.; Gao, P. Research on Coal Mine Goaf Restoration Based on Stability of Overlying Rocks and Numerical Simulation Analysis: A Case Study of Jingmen Garden Expo Park. *Sustainability* **2023**, *15*, 1464. [CrossRef]
7. Yang, Z.; Li, Z.; Zhu, J.; Yi, H.; Hu, J.; Feng, G. Deriving Dynamic Subsidence of Coal Mining Areas Using InSAR and Logistic Model. *Remote Sens.* **2017**, *9*, 125. [CrossRef]
8. Li, Y.; Xu, T.; Guo, H.; Sun, C.; Liu, Y.; Gao, G.; Miao, J. Estimation of Surface Water Level in Coal Mining Subsidence Area with GNSS RTK and GNSS-IR. *Remote Sens.* **2024**, *16*, 3803. [CrossRef]
9. He, Y.; Hu, Z.; Fu, Y.; Yang, K.; Wang, R.; Shi, G.; Feng, Z.; Yang, Q.; Yu, L. Underground Morphological Detection of Ground Fissures in Collapsible Loess Area Based on Three-Dimensional Laser Scanning Technology. *Remote Sens.* **2022**, *14*, 424. [CrossRef]
10. Sharma, A.; Wdowinski, S.; Parkinson, R.W. Coastal Subsidence in Cape Canaveral, FL, and Surrounding Areas: Shallow Subsidence Induced by Natural and Anthropogenic Processes. *Land* **2025**, *14*, 735. [CrossRef]
11. Yang, K.; Hu, Z.; Liang, Y.; Fu, Y.; Yuan, D.; Guo, J.; Li, G.; Li, Y. Automated Extraction of Ground Fissures Due to Coal Mining Subsidence Based on UAV Photogrammetry. *Remote Sens.* **2022**, *14*, 1071. [CrossRef]
12. Puniach, E.; Gruszczyński, W.; Ćwiąkała, P.; Matwij, W. Application of UAV-Based Orthomosaics for Determination of Horizontal Displacement Caused by Underground Mining. *ISPRS J. Photogramm. Remote Sens.* **2021**, *174*, 282–303. [CrossRef]
13. Yang, Q.; Tang, F.; Wang, F.; Tang, J.; Fan, Z.; Ma, T.; Su, Y.; Xue, J. A New Technical Pathway for Extracting High Accuracy Surface Deformation Information in Coal Mining Areas Using UAV LiDAR Data: An Example from the Yushen Mining Area in Western China. *Measurement* **2023**, *218*, 113220. [CrossRef]
14. Modeste, G.; Doubre, C.; Masson, F. Time Evolution of Mining-Related Residual Subsidence Monitored over a 24-Year Period Using InSAR in Southern Alsace, France. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *102*, 102392. [CrossRef]
15. Huang, J.; Du, W.; Jin, S.; Xie, M. Integrated PSInSAR and GNSS for 3D Displacement in the Wudongde Area. *Land* **2024**, *13*, 429. [CrossRef]
16. Liu, L.; Xu, N.; Zhou, W.; Qin, Y.; Luan, S. Improvement of Coal Mining-Induced Subsidence-Affected (MISA) Zone Irregular Boundary Delineation by MT-InSAR Techniques, UAV Photogrammetry, and Field Investigation. *Remote Sens.* **2024**, *16*, 4221. [CrossRef]
17. Alonso-Díaz, A.; Casado-Rabasco, J.; Solla, M.; Lagüela, S. Using InSAR and GPR Techniques to Detect Subsidence: Application to the Coastal Area of "A Xunqueira" (NW Spain). *Remote Sens.* **2023**, *15*, 3729. [CrossRef]
18. Kamali, M.E.; Papoutsis, I.; Loupasakis, C.; Abuelgasim, A.; Omari, K.; Kontoes, C. Monitoring of Land Surface Subsidence Using Persistent Scatterer Interferometry Techniques and Ground Truth Data in Arid and Semi-Arid Regions, the Case of Remah, UAE. *Sci. Total Environ.* **2021**, *776*, 145946. [CrossRef]
19. Yaragunda, V.; Oikonomou, E. Multi-Sensor Fusion for Land Subsidence Monitoring: Integrating MT-InSAR and GNSS with Kalman Filtering and Feature Importance to Northern Attica, Greece. *EARTH* **2025**, *6*, 37. [CrossRef]
20. Chen, Y.; Yu, S.; Tao, Q.; Liu, G.; Wang, L.; Wang, F. Accuracy Verification and Correction of D-InSAR and SBAS-InSAR in Monitoring Mining Surface Subsidence. *Remote Sens.* **2021**, *13*, 4365. [CrossRef]
21. Wang, H.; Li, K.; Zhang, J.; Hong, L.; Chi, H. Monitoring and Analysis of Ground Surface Settlement in Mining Clusters by SBAS-InSAR Technology. *Sensors* **2022**, *22*, 3711. [CrossRef]
22. Ishwar, S.; Kumar, D.; Malik, K. Mine Surface Subsidence Monitoring Due to Impacts of Underground Mining Using Persistent Scatterer Interferometry. *Imaging Sci. J.* **2017**, *65*, 151–161. [CrossRef]
23. Grzovic, M.; Ghulam, A. Evaluation of Land Subsidence from Underground Coal Mining Using TimeSAR (SBAS and PSI) in Springfield, Illinois, USA. *Nat. Hazards* **2015**, *79*, 1739–1751. [CrossRef]

24. Liu, L.; Li, C.; Gao, J.H.; Yu, D.Q.; Xia, L. Geological Disaster Monitoring Combined with BDS and InSAR. *Geol. Sci. Technol. Inf.* **2019**, *38*, 141–149. [CrossRef]
25. Huang, B.; Zhang, H.; Yin, J.H. Surface Deformation Monitoring of Industrial Parks Based on Temporal InSAR Technology. *Bull. Surv. Mapp.* **2024**, *6*, 157–163.
26. Li, P.; Li, Y.; Li, Q.; Cao, Z.; Wang, L. A Novel Analytical Model of Mining Subsidence Considering Time Effect Based on the Probability Integral Theory. *Sustainability* **2022**, *14*, 14443. [CrossRef]
27. Sun, X.; Zhao, C.; Zhang, Y.; Chen, F.; Zhang, S.; Zhang, K. Physical Model Test and Numerical Simulation on the Failure Mechanism of the Roadway in Layered Soft Rocks. *Int. J. Min. Sci. Technol.* **2021**, *31*, 291–302. [CrossRef]
28. Li, Z.; Hou, K.; Li, T.; Tang, J.; Lu, G. Numerical Simulation of Surface Subsidence and Fracture Evolution Caused by Pulang Copper Mine Mining. *Appl. Sci.* **2024**, *14*, 2416. [CrossRef]
29. Sui, L.; Ma, F.; Chen, N. Mining Subsidence Prediction by Combining Support Vector Machine Regression and Interferometric Synthetic Aperture Radar Data. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 390. [CrossRef]
30. Ambrožič, T.; Turk, G. Prediction of Subsidence Due to Underground Mining by Artificial Neural Networks. *Comput. Geosci.* **2003**, *29*, 627–637. [CrossRef]
31. Yang, W.; Xia, X. Prediction of Mining Subsidence under Thin Bedrocks and Thick Unconsolidated Layers Based on Field Measurement and Artificial Neural Networks. *Comput. Geosci.* **2013**, *52*, 199–203. [CrossRef]
32. Ma, F.; Sui, L.; Lian, W. Prediction of Mine Subsidence Based on InSAR Technology and the LSTM Algorithm: A Case Study of the Shigouyi Coalfield, Ningxia (China). *Remote Sens.* **2023**, *15*, 2755. [CrossRef]
33. Zhou, D.; Zuo, X.; Zhao, Z. Constructing a Large-Scale Urban Land Subsidence Prediction Method Based on Neural Network Algorithm from the Perspective of Multiple Factors. *Remote Sens.* **2022**, *14*, 1803. [CrossRef]
34. Ma, L.; Liu, Y.; Zhang, X.; Ye, Y.; Yin, G.; Johnson, B.A. Deep Learning in Remote Sensing Applications: A Meta-Analysis and Review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *152*, 166–177. [CrossRef]
35. Lu, J.; Wang, Y.; Zhu, Y.; Liu, J.; Xu, Y.; Yang, H.; Wang, Y. DACLnet: A Dual-Attention-Mechanism CNN-LSTM Network for the Accurate Prediction of Nonlinear InSAR Deformation. *Remote Sens.* **2024**, *16*, 2474. [CrossRef]
36. Chen, Y.; Chen, X.; Guo, S.; Li, H.; Du, P. A Novel Surface Deformation Prediction Method Based on AWC-LSTM Model. *Int. J. Appl. Earth Obs. Geoinf.* **2024**, *135*, 104292. [CrossRef]
37. Madaeni, F.; Chokmani, K.; Lhissou, R.; Homayouni, S.; Gauthier, Y.; Tolszczuk-Leclerc, S. Convolutional Neural Network and Long Short-Term Memory Models for Ice-Jam Predictions. *Cryosphere* **2022**, *16*, 1447–1468. [CrossRef]
38. Mahardika, T.N.Q.; Fuadah, Y.N.; Jeong, D.U.; Lim, K.M. PPG Signals-Based Blood-Pressure Estimation Using Grid Search in Hyperparameter Optimization of CNN–LSTM. *Diagnostics* **2023**, *13*, 2566. [CrossRef]
39. Lu, A.-Z.; Xu, G.; Sun, F.; Sun, W.-Q. Elasto-Plastic Analysis of a Circular Tunnel Including the Effect of the Axial in Situ Stress. *Int. J. Rock Mech. Min. Sci.* **2010**, *47*, 50–59. [CrossRef]
40. Filippou, K.; Aifantis, G.; Papakostas, G.A.; Tsekouras, G.E. Structure Learning and Hyperparameter Optimization Using an Automated Machine Learning (AutoML) Pipeline. *Information* **2023**, *14*, 232. [CrossRef]
41. Sicard, D.; Briois, P.; Billard, A.; Thevenot, J.; Boichut, E.; Chapellier, J.; Bernard, F. Deep Learning and Bayesian Hyperparameter Optimization: A Data-Driven Approach for Diamond Grit Segmentation toward Grinding Wheel Characterization. *Appl. Sci.* **2022**, *12*, 2606. [CrossRef]
42. Goodarzi, P.; Schauer, J.; Schütze, A. Robust Distribution-Aware Ensemble Learning for Multi-Sensor Systems. *Sensors* **2025**, *25*, 831. [CrossRef] [PubMed]
43. Sezgin, F.H.; Algorabi, Ö.; Sart, G.; Güler, M. Hyperparameter-Optimized RNN, LSTM, and GRU Models for Airline Stock Price Prediction: A Comparative Study on THYAO and PGSUS. *Symmetry* **2025**, *17*, 1905. [CrossRef]
44. D'Aranno, P.J.V.; Di Benedetto, A.; Fiani, M.; Marsella, M.; Moriero, I.; Palenzuela Baena, J.A. An Application of Persistent Scatterer Interferometry (PSI) Technique for Infrastructure Monitoring. *Remote Sens.* **2021**, *13*, 1052. [CrossRef]
45. Klemetti, T.M.; Sears, M.M.; Tulu, I.B. Design Concerns of Room and Pillar Retreat Panels. *Int. J. Min. Sci. Technol.* **2017**, *27*, 29–35. [CrossRef]
46. Cigna, F.; Esquivel Ramírez, R.; Tapete, D. Accuracy of Sentinel-1 PSI and SBAS InSAR Displacement Velocities against GNSS and Geodetic Leveling Monitoring Data. *Remote Sens.* **2021**, *13*, 4800. [CrossRef]
47. Tarighat, F.; Foroughnia, F.; Perissin, D. Monitoring of Power Towers' Movement Using Persistent Scatterer SAR Interferometry in South West of Tehran. *Remote Sens.* **2021**, *13*, 407. [CrossRef]
48. Luo, Q.; Zhou, G.; Perissin, D. Monitoring of Subsidence along Jingjin Inter-City Railway with High-Resolution TerraSAR-X MT-InSAR Analysis. *Remote Sens.* **2017**, *9*, 717. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

computation

MDPI

Article

SARIMA vs. Prophet: Comparative Efficacy in Forecasting Traffic Accidents Across Ecuadorian Provinces

Wilson Chango [1,2,*], Ana Salguero [2], Tatiana Landivar [2], Roberto Vásconez [2], Geovanny Silva [2], Pedro Peñafiel-Arcos [2], Lucía Núñez [2] and Homero Velasteguí-Izurieta [1,*]

[1] Department of Systems and Computation, Pontifical Catholic University of Ecuador, Esmeraldas Campus PUCESE, Esmeraldas 080101, Ecuador
[2] Faculty of Informatics and Electronics, Escuela Superior Politécnica de Chimborazo (ESPOCH), Riobamba 060155, Ecuador; anax.salguero@espoch.edu.ec (A.S.); tatiana.landivar@espoch.edu.ec (T.L.); robertoc.vasconez@espoch.edu.ec (R.V.); geovanny.silva@espoch.edu.ec (G.S.); pedro.peniafiela@espoch.edu.ec (P.P.-A.); lucia.nuniez@espoch.edu.ec (L.N.)
* Correspondence: wilson.chango@pucese.edu.ec (W.C.); homero.j.velastegui.i@pucese.edu.ec (H.V.-I.); Tel.: +593-960949920 (H.V.-I.)

Abstract

This study aimed to evaluate the comparative predictive efficacy of the SARIMA statistical model and the Prophet machine learning model for forecasting monthly traffic accidents across the 24 provinces of Ecuador, addressing a critical research gap in model selection for geographically and socioeconomically heterogeneous regions. By integrating classical time series modeling with algorithmic decomposition techniques, the research sought to determine whether a universally superior model exists or if predictive performance is inherently context-dependent. Monthly accident data from January 2013 to June 2025 were analyzed using a rolling-window evaluation framework. Model accuracy was assessed through Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) metrics to ensure consistency and comparability across provinces. The results revealed a global tie, with 12 provinces favoring SARIMA and 12 favoring Prophet, indicating the absence of a single dominant model. However, regional patterns of superiority emerged: Prophet achieved exceptional precision in coastal and urban provinces with stationary and high-volume time series—such as Guayas, which recorded the lowest MAPE (4.91%)—while SARIMA outperformed Prophet in the Andean highlands, particularly in non-stationary, medium-to-high-volume provinces such as Tungurahua (MAPE 6.07%) and Pichincha (MAPE 13.38%). Computational instability in MAPE was noted for provinces with extremely low accident counts (e.g., Galápagos, Carchi), though RMSE values remained low, indicating a metric rather than model limitation. Overall, the findings invalidate the notion of a universally optimal model and underscore the necessity of adopting adaptive, region-specific modeling frameworks that account for local geographic, demographic, and structural factors in predictive road safety analytics.

Keywords: SARIMA; Prophet; traffic accident prediction; time series forecasting; model selection; Ecuador

1. Introduction

Injuries from traffic accidents constitute a global public health crisis of epidemic proportions, being the leading cause of death among children and young people worldwide according to the World Health Organization, which reports approximately 1.3 million

annual fatalities and between 20 and 50 million people who suffer non-fatal injuries [1]. This situation represents a particularly devastating socioeconomic burden for developing countries, where healthcare systems often exhibit greater structural vulnerabilities.

This worrying global context underscores the urgent need to develop sophisticated predictive tools that enable proactive management of road safety. The conceptual framework of our study is based on the comparative analysis of time series models for accident prediction, specifically evaluating the predictive efficacy of traditional statistical approaches versus machine learning methodologies. The relevance of this research transcends the methodological, addressing a critical public health problem where the ability to anticipate fluctuations in road traffic crashes can optimize the allocation of healthcare and emergency resources, design more effective preventive interventions and, ultimately, save lives.

The justification for this work emerges from a research gap identified in the scientific literature: there is a limited number of exhaustive comparative studies that evaluate the performance of predictive models across diverse geographical regions with heterogeneous socioeconomic and topographical characteristics [2]. This deficiency has prevented the establishment of clear guidelines for model selection in specific operational contexts, limiting the transferability of findings across different territorial realities.

Investigative background reveals that while SARIMA (Seasonal Autoregressive Integrated Moving Average) models have demonstrated robustness in capturing seasonal and trend patterns in transportation data [3], modern algorithms like Prophet, developed by Facebook, offer superior flexibility in handling nonlinear patterns and outliers [4]. The theories supporting this study are rooted in the integration of classic statistical time series analysis, based on Box–Jenkins theory, with the computational principles of machine learning, creating a hybrid theoretical framework that allows for rigorously evaluating the assumptions, advantages, and limitations of each predictive paradigm.

The context in which this study is conducted—the 24 provinces of Ecuador between 2013 and 2024—offers an ideal natural laboratory due to its geographical diversity (coastal, Andean, and Amazonian regions), climatic variability, and disparities in road infrastructure, providing a robust empirical scenario for testing the models' generalizability. The pioneering innovation of this research lies in constituting the first comparative analysis at the Ecuadorian national level of classical versus ML approaches for accident prediction, generating locally contextualized evidence.

Consequently, this study seeks to answer the following research question: To what extent does the predictive efficacy differ between the SARIMA and Prophet models when forecasting traffic accidents across the diverse provinces of Ecuador, and what regional factors account for these variations in performance?

2. Background

2.1. Road Safety and Risk Prediction as a Global and Regional Public Health Challenge

Injuries resulting from traffic accidents represent one of the main causes of morbidity and mortality worldwide, surpassing in some countries the mortality rates for infectious diseases [5]. It is estimated that more than 1.19 million people die each year in road traffic crashes, and that traffic accidents constitute the leading cause of death among young people aged 15–29, with an economic impact exceeding 3% of the global GDP [6].

In the Latin American context, recent studies highlight the lack of evidence-based predictive mechanisms to identify critical risk zones and periods [7]. Road safety is, therefore, recognized as a structural public health challenge that requires preventive strategies oriented toward proactive risk management, supported by quantitative and predictive data analysis.

https://doi.org/10.3390/computation14010005

In particular, the use of time series models allows for the analysis of historical trends and the prediction of crash occurrences, facilitating the optimization of resources in health, transportation, and police control [8]. The integration of forecasting tools into road safety management not only offers operational advantages but constitutes a key component of public policies for the prevention of avoidable deaths, especially in countries with territorial disparities in infrastructure and vehicular control such as Ecuador [9] (see Table 1).

Table 1. Global Mortality Trends from Road Traffic Crashes and Methodological Approaches.

Paper	Global Mortality Trends (Summary)	Methodological Approach (Summary)
[10]	Trend: Decreasing (China). Total Deaths: 191,000 (2015) to 169,000 (2030). Rate/100k: 13.7 (2015) to 11.8 (2030).	Model: Log-linear models. Country/Period: China (2015–2030). Sources: UN, WHO, USDA.
[11]	Trend: Decreasing (then plateauing). Rate/100k: Decreased from 20.6 (2010) to 19.9 (2013). Annual % Change: −1.12%.	Model: Negative binomial regression. Countries/Period: 188 countries (2010–2013). Techniques: Age-adjustment.
[12]	Trend: Increasing (in low-income countries and the African Region). Global Deaths: 1.35 million annually.	Model: Descriptive analysis, *t*-test, Wilcoxon. Sources: WHO Global Status Reports (2013–2018). Countries: 161.
[13]	Trend: Increasing (projected seventh leading cause by 2030). Global Deaths: 1.3 million (2019 est.). Difference: 93% of mortality occurs in low- and middle-income countries.	Approach: N/A (Review Article, no primary data analysis methodology described).
[14]	Trend: N/A (Focus is specific to Iran).	Model: Joinpoint regression (based on log-linear model). Country/Period: Iran (2006–2020). Source: Iranian Legal Medicine Organization.

2.2. Fundamentals of Classic Time Series Models

The SARIMA (Seasonal Autoregressive Integrated Moving Average) model is an extension of the ARIMA model that incorporates seasonal components in order to capture patterns that repeat at regular time intervals, following the methodology proposed by Box and Jenkins [15].

Mathematically, SARIMA is denoted as:

$$\text{SARIMA}(p, d, q)(P, D, Q)_s$$

where

- p and q represent the orders of the autoregressive and moving average components.
- d represents the necessary differentiation to achieve stationarity.
- (P, D, Q) are the analogous parameters for the seasonal component with periodicity s [16].

Stationarity, understood as the statistical invariance of the mean and variance over time, is an essential requirement to ensure the model's validity [17]. Its application in accident prediction contexts has proven effective for series with regular and seasonal patterns, such as monthly vehicular traffic fluctuations [18].

Furthermore, SARIMA models are validated using statistical tests such as the **Ljung–Box test**, the **Partial Autocorrelation Function (PACF)** analysis, and the selection of optimal orders based on the **AIC** and **BIC** information criteria [19]. This methodological structure has made the SARIMA model a classic and robust tool in forecasting variables related to mobility and road safety.

https://doi.org/10.3390/computation14010005

2.3. Modern Forecasting Algorithms: The Facebook Prophet Model

The Prophet model, developed by Taylor and Letham at Meta (Facebook), constitutes a modern machine learning approach for time series forecasting. Unlike traditional models based on stationarity, Prophet employs an additive model that decomposes the time series into three fundamental components: trend, seasonality, and holiday or special event effects [20].

This flexible approach allows capturing complex and nonlinear patterns, common in traffic data influenced by climatic, tourist, or socioeconomic variations [21]. Prophet employs optimization algorithms based on Bayesian regression, facilitating the dynamic updating of parameters and the incorporation of new observations in real-time [22]. Various studies have shown that its performance is superior to that of SARIMA models in scenarios with high temporal irregularity or the presence of *outliers* [23]. In the Ecuadorian context, its application is particularly relevant given the climatic and topographic diversity of the 24 provinces, which condition the occurrence of traffic accidents in a differentiated manner (See Table 2).

Table 2. Summary of Applications and Performance of the Prophet Model in Various Domains.

Paper	Application Domain (Summary)	Model Performance
[24]	Sector: Energy. Data: Monthly total and peak energy demand. Purpose: Planning and management of generation facilities.	Superiority: Prophet demonstrated superior performance to SARIMA and LSTM by capturing complex temporal features and handling periodicity. Metrics (MAPE): 3.3% (Total Demand), 3.01% (Peak Demand).
[25]	Sector: Energy. Data: Total monthly electric energy consumption (GWh). Purpose: Improve prediction accuracy for decision-making.	Comparison: Outperformed by neural networks (LSTM/GRU) in Brazil, but showed good performance for Turkey. Alternative Models: ARIMA, SVR, Holt-Winters.
[26]	Sector: Healthcare (Radiology). Data: Daily volume of CT and MRI exams. Purpose: Optimal resource allocation and planning.	Effectiveness: Reduced prediction error. CT: Error reduced from 19 to <1 per day (in monthly total). Improvement: Significantly better than manual predictions.
[27]	Sector: Water Management/ Agriculture. Data: Monthly rainfall (decadal scale). Characteristics: Strong seasonality.	Superiority: Outperformed six regression models (incl. MLPs and Boosting). Strength: Excellent in predicting dry periods due to its multiplicative seasonality function. Limitation: Difficulty with extreme peak values (very wet events).
[28]	Sector: Environmental Monitoring (Air Quality). Data: Hourly concentrations of PM10 and PM2.5. Purpose: Mitigate health risks from pollution.	Performance: RMSE below 6.26 $\mu g/m^3$ for PM2.5 and 9.99 $\mu g/m^3$ for PM10 (in 50% of cases). Strength: Simplicity and speed. Condition: Performed better for PM2.5 than for PM10.

2.4. Evaluation and Validation Criteria for Multi-Regional Prediction Models

The evaluation of predictive models in multi-regional contexts requires the use of quantitative performance metrics and statistical tests that determine the superiority of one model over another. Two of the most commonly used metrics are the Mean Absolute Percentage Error (MAPE) and the Root Mean Square Error (RMSE), which measure prediction

https://doi.org/10.3390/computation14010005

accuracy and error stability across regions [29]. These indicators are especially useful when analyzing datasets originating from provinces with heterogeneous temporal behaviors, such as those of Ecuador. Recent studies emphasize that MAPE and RMSE continue to be fundamental metrics in traffic forecasting, particularly when models must handle complex temporal patterns and varying demand conditions [30].

Additionally, the Diebold–Mariano (DM) test provides a robust statistical framework for comparing the predictive performance of two models over the same time series, evaluating whether differences in forecasting errors are statistically significant [31]. In the context of intelligent transportation systems, recent advances highlight the relevance of statistically grounded comparisons when working with deep learning or hybrid forecasting architectures, as shown in the multi-regional evaluation framework proposed by [32].

The integration of these metrics and statistical procedures ensures an objective and reproducible evaluation of the relative performance of SARIMA and Prophet. This approach allows determining which model offers greater accuracy and generalization capacity across heterogeneous regions. Complementary comparative error plots and cross-validation tables further strengthen methodological transparency, an essential aspect of predictive research applied to road safety.

3. Materials and Methods

The comparative analysis of predictive models was conducted under the principles of Design Science Research (DSR), aiming to develop and rigorously evaluate a predictive artifact (an automated multi-regional forecasting framework) capable of identifying the most effective time series model for traffic accident prediction across different Ecuadorian provinces. The DSR structure integrates the PRISMA framework for the initial literature review and the CRISP-DM methodology for the artifact's design and development, ensuring both theoretical relevance and operational rigor [33] (see Figure 1).

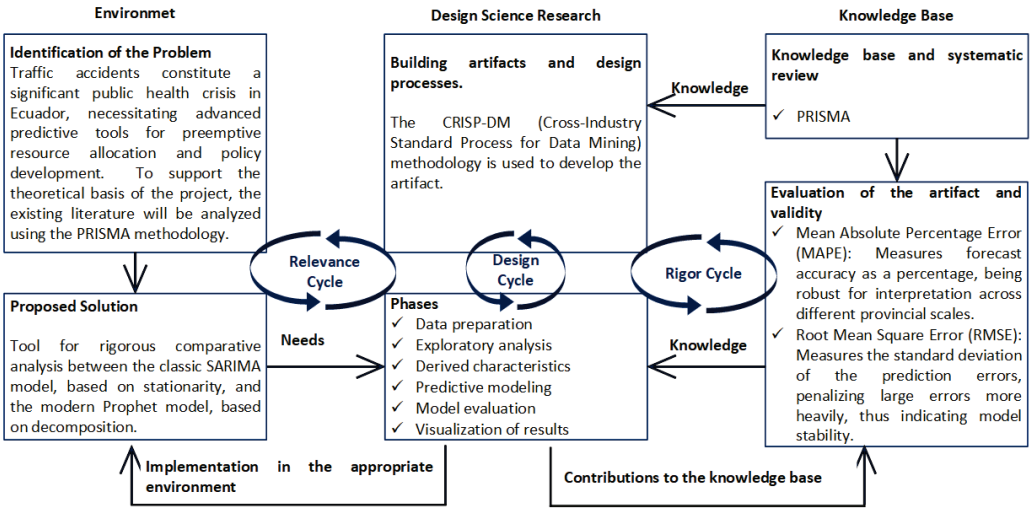


Figure 1. Methodological Framework (DSR) Integrating PRISMA and CRISP-DM.

3.1. Relevance Phase

3.1.1. Problem Identification

Traffic accidents constitute a significant public health crisis in Ecuador, necessitating advanced predictive tools for preemptive resource allocation and policy development [34]. Traditional forecasting methods often fail to capture the complex seasonality, non-linear trends, and regional heterogeneity present in accident data. This gap justifies the comparative study between the classical, stationary-based SARIMA model and the modern, decomposition-based Prophet model.

https://doi.org/10.3390/computation14010005

3.1.2. Systematic Literature Review (PRISMA Framework)

To define the research gap and justify the model comparison, a systematic review was performed following the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines [35]. The search focused on comparative studies of time series forecasting in traffic safety or public health, with a temporal range from 2020 to 2025 (see Figure 2).

- Identification: An exhaustive search was carried out in scientific databases and registers, initially identifying $n = 140$ records (120 from databases and 20 from registers). Combined keywords with Boolean operators were used: ("SARIMA" OR "ARIMA") AND ("Prophet") AND ("traffic accidents" OR "road safety") AND ("forecasting"). The temporal range considered was from 2020 to 2025.
- Screening: From the identified records, $n = 20$ were removed before screening (5 duplicate records, 5 marked as ineligible by automation tools, and 10 removed for other reasons). The remaining $n = 100$ records were screened by title and abstract, of which $n = 20$ were excluded (e.g., non-empirical publications or studies that did not use time series models).
- Eligibility: $n = 80$ full-text reports were sought for retrieval. These $n = 80$ reports were assessed for eligibility. $n = 30$ reports were excluded based on detailed criteria: $n = 10$ due to methodological flaws (e.g., insufficient data), $n = 10$ due to a lack of performance comparison between key models, and $n = 10$ for not utilizing flexible decomposition models.
- Inclusion: Finally, $n = 50$ studies were included for qualitative synthesis. These documents demonstrate the existing gap and justify the need for the systematic comparison between traditional modeling (SARIMA) and modern additive models (Prophet), serving as the foundation for the proposed artifact's justification.

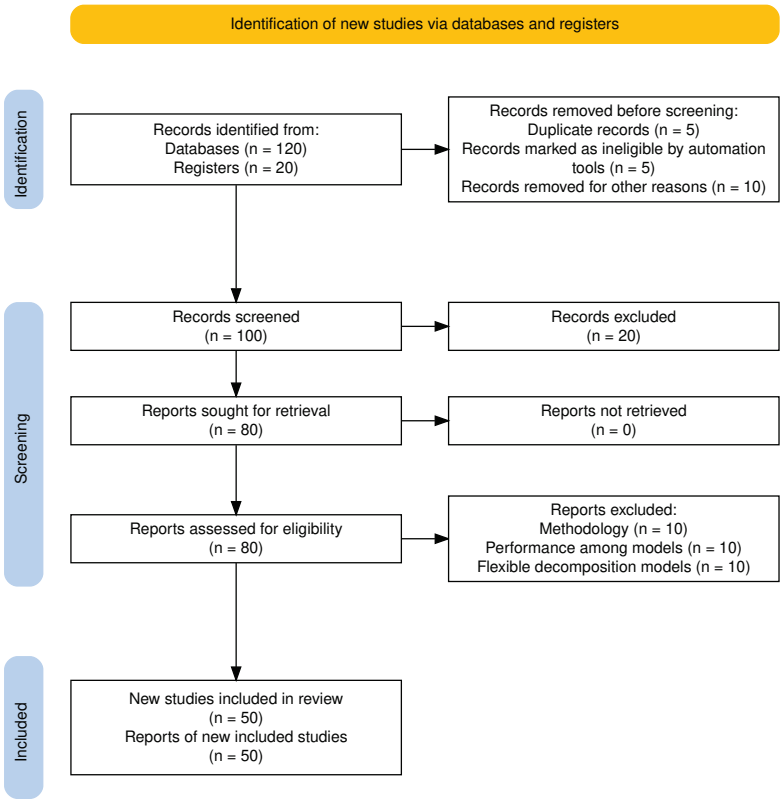


Figure 2. PRISMA Framework for Systematic Variable Selection.

https://doi.org/10.3390/computation14010005

3.2. Design and Development Phase (CRISP-DM)

3.2.1. Data Preparation and Exploratory Analysis

This study uses monthly traffic accident records from Ecuador's national database, covering the 24 provinces from January 2013 to June 2025. The data were aggregated at a monthly resolution ($s = 12$) to capture annual seasonality and separated into 24 individual time series, one per province.

Data Source and Granularity. The information was structured at a monthly frequency to allow the identification of seasonal patterns and to ensure comparability across provinces.

Preprocessing. A comprehensive preprocessing phase was conducted. This included: (a) identifying the proportion and temporal distribution of missing values, (b) detecting potential outliers that could distort model fitting, and (c) carefully selecting an interpolation strategy. Sporadic missing values were imputed using linear interpolation; although simple, this method was validated against the short gaps observed and the smooth monthly dynamics of the series. Structural shifts were normalized when necessary to avoid distortions in trend estimation.

Stationarity Analysis. For the SARIMA approach, stationarity was assessed with the Augmented Dickey–Fuller (ADF) test for each provincial series. Differencing (parameter d) was applied only when required to stabilize the mean and variance [36].

3.2.2. Predictive Modeling

Two modeling frameworks were implemented to capture the temporal behavior of the multi-regional data.

SARIMA Model Specification. Optimal SARIMA orders were selected per province through an iterative search guided by the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). Model adequacy was validated using the Ljung–Box test to confirm that no significant autocorrelation remained in the residuals. For baseline comparison, a SARIMA(1,1,1)(1,1,1)$_{12}$ configuration was evaluated, as it effectively captures the annual seasonality commonly present in monthly accident data.

Prophet Model Specification. The Prophet model, based on an additive decomposition framework [37], was also implemented. Prophet is formally expressed as:

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t \tag{1}$$

where $y(t)$ is the observed series, $g(t)$ is the non-linear trend (piecewise linear or logistic), $s(t)$ models seasonal components (weekly and annual), $h(t)$ accounts for holiday effects (not included in the baseline comparison), and ε_t is the error term. In this study, the model automatically considered the annual seasonality component and estimated the underlying non-linear trend.

3.2.3. Model Evaluation Setup and Visualization

A rolling-window evaluation scheme was applied using a fixed training period and a 12-month sliding forecast window. This approach simulates real forecasting conditions and enables assessing model stability over time [38]. Comparative error plots were generated for each province to visually contrast the performance of the SARIMA and Prophet models.

3.3. Rigor Phase—Artifact Evaluation

The effectiveness of the predictive artifact was assessed using quantitative accuracy metrics and statistical significance testing.

https://doi.org/10.3390/computation14010005

3.3.1. Performance Metrics

Model performance for both SARIMA and Prophet was evaluated using two primary metrics:

Mean Absolute Percentage Error (MAPE). This metric evaluates forecast accuracy as a percentage and is especially useful for comparing provinces of different scales.

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \times 100\% \quad (2)$$

Root Mean Square Error (RMSE). This metric captures the standard deviation of forecasting errors and penalizes large deviations more heavily, thus reflecting model stability.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2} \quad (3)$$

where in Formulas (2) and (3), A_t is the actual value, F_t the forecast value, and n the number of predictions.

3.3.2. Diebold–Mariano Statistical Test

The Diebold–Mariano (DM) test was applied to determine whether the prediction accuracy of the two models differed significantly in each province [39]. The test evaluates the null hypothesis that both models exhibit equal forecasting accuracy. Rejection of the null at $\alpha = 0.05$ indicates that one model is statistically superior. This province-specific analysis provides rigorous statistical support to identify the most effective forecasting approach for broader spatial generalization [40].

4. Results

4.1. Coast Region

The analysis of the traffic accident time series in Guayas (101,179 total accidents) reveals a stationary series (p-value of 0.018), which simplifies the modeling process.

The comparative evaluation between the models confirms a significant superiority of the Prophet model over SARIMA.

- Prophet demonstrated much higher accuracy with a Mean Absolute Percentage Error (MAPE) of 4.91% and a Root Mean Square Error (RMSE) of 33.76.
- SARIMA recorded considerably higher metrics (MAPE of 9.77% and RMSE of 64.64).

Prophet's performance represents a reduction of almost 50% in the average prediction error, positioning it as the most effective and stable model for this province. Prophet's future projections suggest a seasonal pattern, predicting a peak in accidents in December 2025 (733) and a trough in February 2026 (568) (see Figure 3).

The Figure 4 illustrates the temporal behavior and forecast performance of traffic accidents in Manabí province (2013–2025). The time series reveals fluctuations with peaks in 2014 and 2021, followed by a moderate decline during 2023–2024. The trend shows a gradual decrease with slight recovery in early 2025. Forecasts for July 2025–June 2026 indicate a mild downward trend, with SARIMA providing smoother and more accurate results (MAPE 14.22%, RMSE 26.32) compared to Prophet. The ADF test (p = 0.008) confirmed the stationarity of the series, and the predicted monthly accidents range between 73 and 109, suggesting moderate variability and stability in the short term.

https://doi.org/10.3390/computation14010005

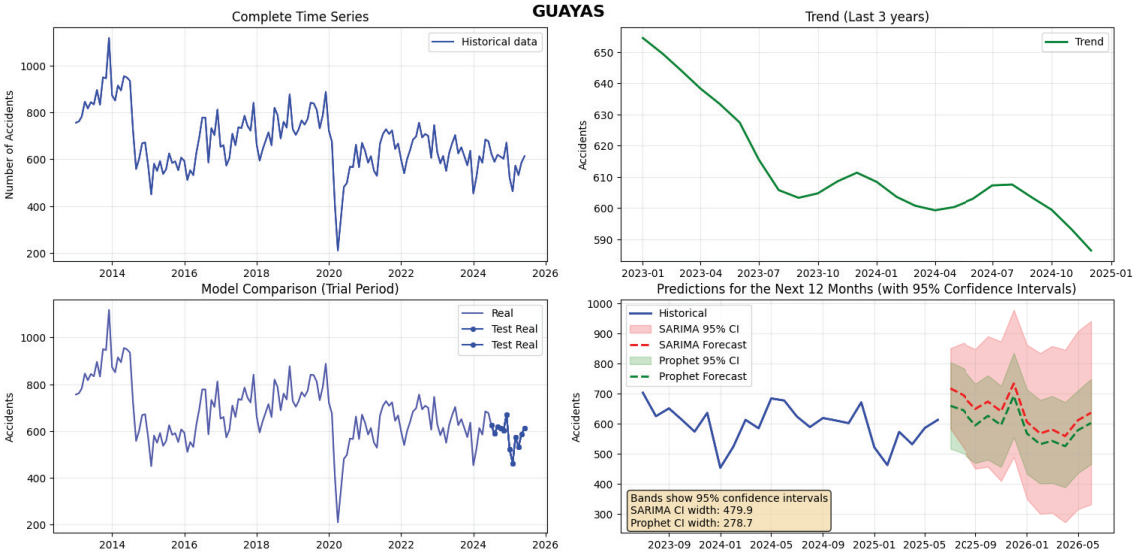


Figure 3. Accident Forecasting Analysis for Guayas Province Using SARIMA and Prophet Models.

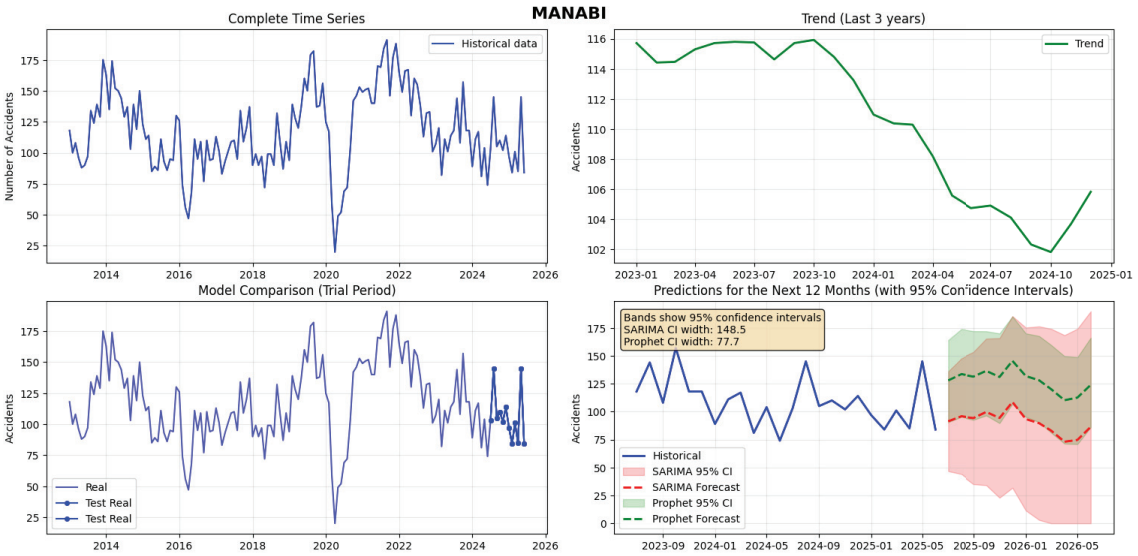


Figure 4. Accident Forecasting Analysis for Manabí Province Using SARIMA and Prophet Models.

The Figure 5 depicts the monthly evolution and forecasts of traffic accidents in Los Ríos province (2013–2025). The time series reveals fluctuations with peaks above 130 accidents during 2014–2015 and a downward trend after 2022. The recent trend shows a steady decline until mid-2024 with a mild recovery in 2025. Both SARIMA and Prophet models produced consistent forecasts for July 2025–June 2026, stabilizing around 70–100 accidents per month. Quantitatively, Prophet achieved slightly better accuracy (MAPE 17.91%, RMSE 15.14) than SARIMA (MAPE 19.88%, RMSE 16.85). The ADF test (p = 0.004) confirmed the series' stationarity, supporting SARIMA's validity and indicating a stable, moderately variable trend in road accidents for the province.

The Figure 6 shows the monthly evolution of traffic accidents between 2013 and 2025, with projections extending to 2026. A total of 11,356 accidents were recorded, with a monthly average of 75.7 and values ranging between 28 and 159. Peaks were observed in 2014–2015 and 2019–2020, followed by stabilization after 2021. The stationarity test (p-value = 0.002) confirms that the series is suitable for ARIMA/SARIMA models. Both models demonstrate high predictive accuracy (MAPE < 15%), although Prophet outperforms SARIMA with a lower error (MAPE 9.67% vs. 12.35%). Forecasts for 2025–2026

https://doi.org/10.3390/computation14010005

indicate stability between 73 and 104 accidents, with a peak in December 2025 and stabilization around 82 by mid-2026. Overall, the series exhibits a predictable pattern, and Prophet provides the best overall performance.

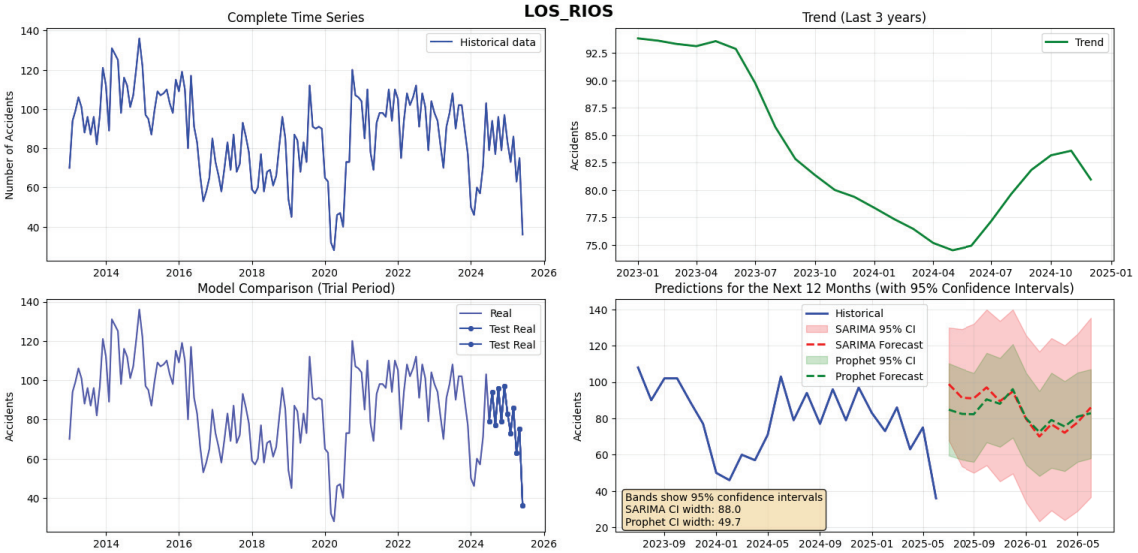


Figure 5. Accident Forecasting Analysis for Los Ríos Province Using SARIMA and Prophet Models.

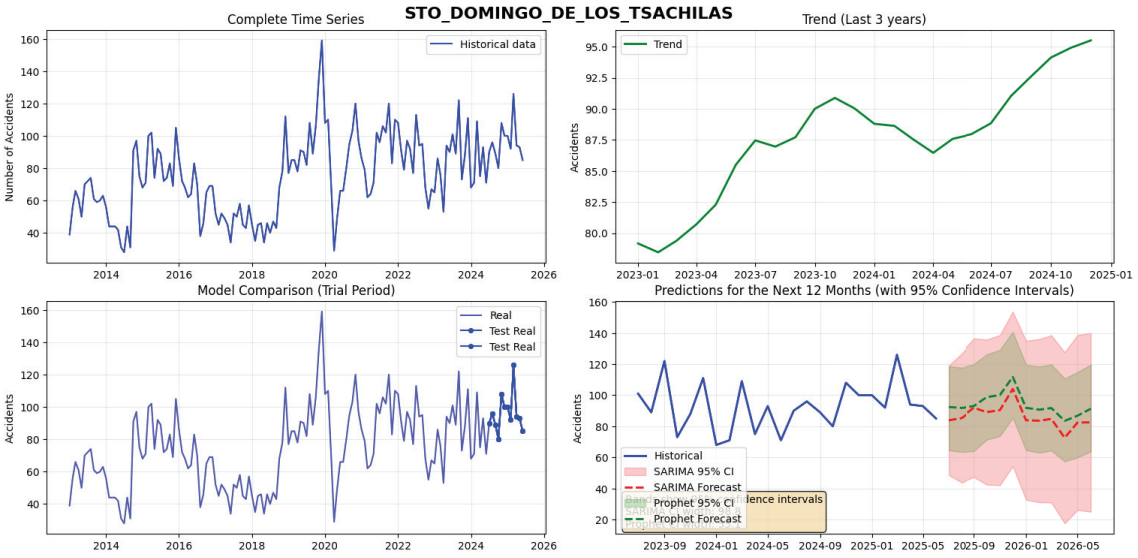


Figure 6. Temporal Evolution and Prediction of Traffic Accidents, SARIMA Model vs. Prophet.

The Figure 7 summarizes the temporal evolution and predictive analysis of traffic accidents in El Oro province (2013–2025).

Accidents peaked above 100 cases between 2014–2016, followed by a steady decline after 2018, as shown in the full time series. The recent trend (2023–2024) confirms a continued downward pattern, suggesting improved safety conditions.

Model validation indicates consistent predictive performance, while 12-month forecasts (July 2025–June 2026) show a stable range of 33–48 accidents per month. Quantitatively, SARIMA outperformed Prophet, with MAPE = 18.21% and RMSE = 7.97, versus Prophet's MAPE = 40.60% and RMSE = 16.11. Despite the non-stationary series ($p = 0.331$), SARIMA demonstrated superior stability and accuracy in forecasting accident trends.

The Figure 8 summarizes the temporal evolution and predictive modeling of traffic accidents in Santa Elena. The complete time series (top-left) shows high variability with

https://doi.org/10.3390/computation14010005

peaks near 100 accidents (2014–2015) and stabilization between 40 and 60 after 2020. The trend (top-right) indicates a decline until early 2024, followed by a slight recovery in 2025. The model validation (bottom-left) reveals strong consistency between real and test data, while the 12-month forecasts (bottom-right) suggest a stable accident frequency between 41 and 51 cases per month. Quantitatively, the SARIMA model outperformed Prophet (MAPE = 15.29%, RMSE = 10.31 vs. MAPE = 20.58%, RMSE = 12.78). The ADF test (p = 0.000) confirmed stationarity, validating SARIMA's assumptions and indicating stable medium-term accident trends.

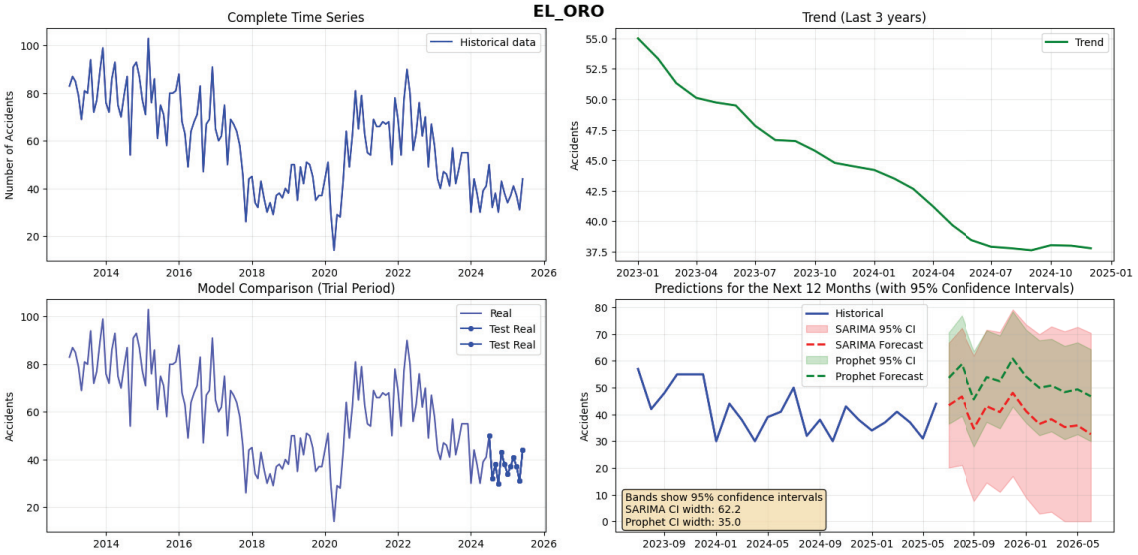


Figure 7. Forecasting Monthly Traffic Accidents in El Oro Province (2013–2026).

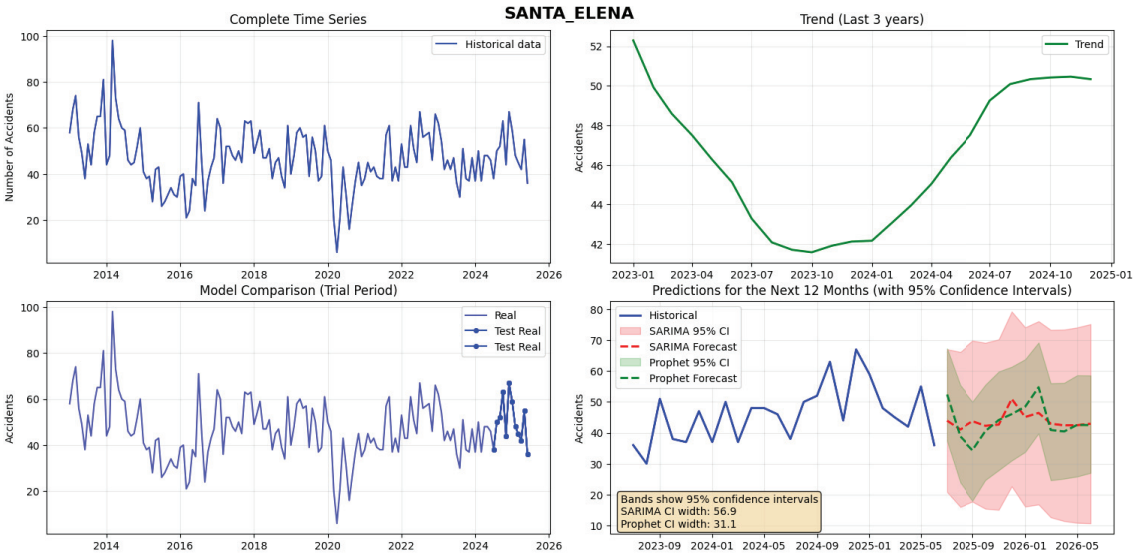


Figure 8. Traffic Accident Forecasting and Temporal Trends in Santa Elena Province (2013–2026).

The analysis focuses on the traffic accident time series for the province of Esmeraldas, which is characterized by **low volume** (3039 total accidents, 20.3 monthly average) and **extreme volatility** (minimum of 2, maximum of 54). Methodologically, the series is **NON-stationary** (p-value of 0.411), implying the need for differentiation in the SARIMA model (see Figure 9).

In the comparative evaluation, although both models exhibit a high **Mean Absolute Percentage Error (MAPE)** (Prophet: 73.93%; SARIMA: 92.37%), which is typical for low-

https://doi.org/10.3390/computation14010005

volume series, the **Prophet model demonstrates clear superiority**. Prophet records an **MAPE** that is 18.44 percentage points lower and an **RMSE** (Root Mean Square Error) of 6.10 (compared to 6.81 for SARIMA). This lower metric in absolute terms positions **Prophet** as the most accurate and stable model for forecasting accidents in this non-stationary, low-volume time series (see Table 3).

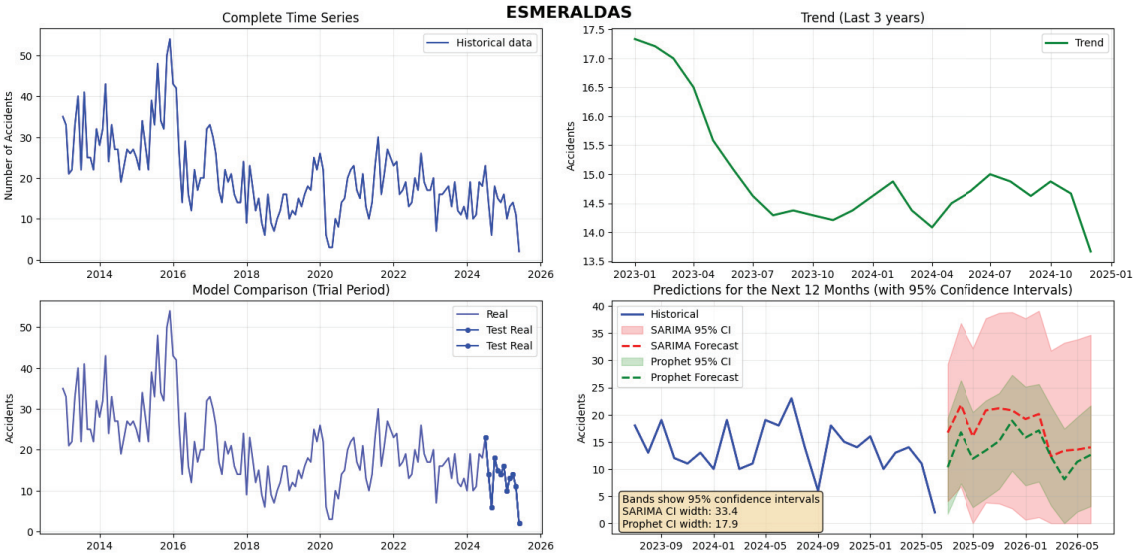


Figure 9. Comparative Analysis of SARIMA vs. Prophet and Accident Prediction in Esmeraldas.

Table 3. Comparative Performance of Forecasting Models in Coastal Region Provinces.

Province	Total Accidents	Monthly Average	Best Model	Best Model MAPE	Stationarity
GUAYAS	101,179	674.5	Prophet	4.91%	Stationary
MANABÍ	17,592	117.3	SARIMA	14.22%	Stationary
LOS RÍOS	13,063	87.1	Prophet	17.91%	Stationary
SANTO DOMINGO	11,356	75.7	Prophet	9.67%	Stationary
EL ORO	8764	58.4	SARIMA	18.21%	Non-Stationary
SANTA ELENA	7027	46.8	SARIMA	15.29%	Stationary
ESMERALDAS	3039	20.3	Prophet	73.93%	Non-Stationary

4.2. Andean Region

90,636 accidents were recorded with a monthly average of 604.2, showing high variability between 115 and 1617 monthly accidents. The series is not stationary (p-value: 0.463), justifying the use of SARIMA. This model demonstrated high performance (MAPE: 13.38%, RMSE: 55.33), while Prophet had low performance (MAPE: 66.62%, RMSE: 248.24). SARIMA predictions for the next 12 months range between 380–494 accidents, with a peak in December 2025, showing overall stability with seasonal variations (see Figure 10).

15,398 accidents were analyzed with a monthly average of 102.7, showing variability between 17 and 173 monthly incidents. The series is not stationary (p-value: 0.076). The SARIMA model showed poor performance (MAPE: 72.51%, RMSE: 46.95), while Prophet demonstrated superior performance (MAPE: 48.34 RMSE: 33.64). SARIMA predictions for

https://doi.org/10.3390/computation14010005

the next 12 months range between 111–132 accidents, showing stability but with limited reliability due to high error. It is recommended to use Prophet over SARIMA (see Figure 11).

Our analysis of 15,280 traffic accidents in Tungurahua revealed a monthly average of 101.9, though the data showed significant fluctuations from as few as 6 to as many as 228 accidents in a single month. The data pattern over time is not stationary, confirming the presence of underlying trends or seasonal effects.

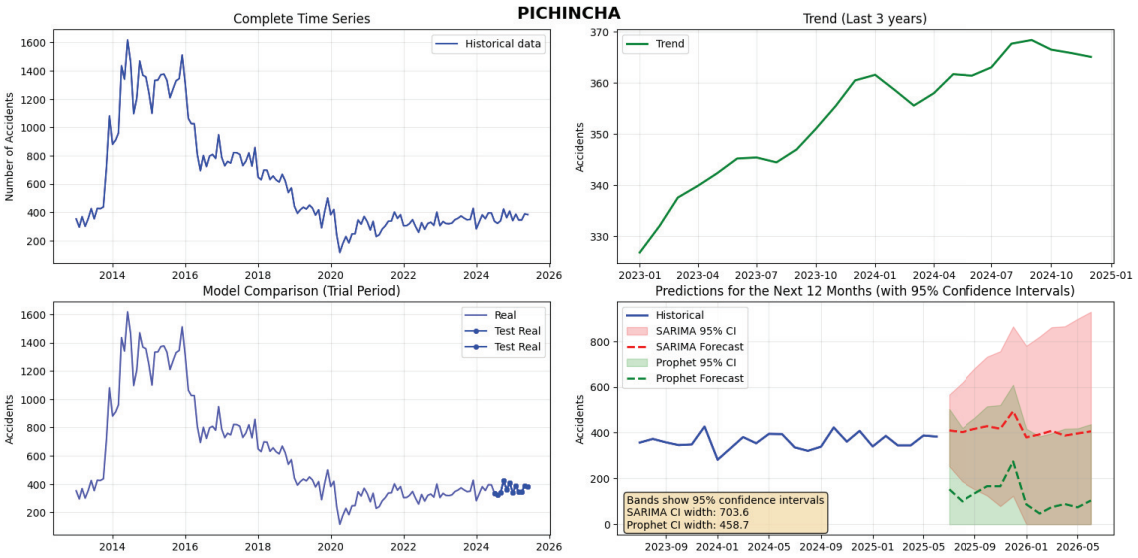


Figure 10. Traffic Accident Forecasting and Temporal Trends in Pichincha Province.

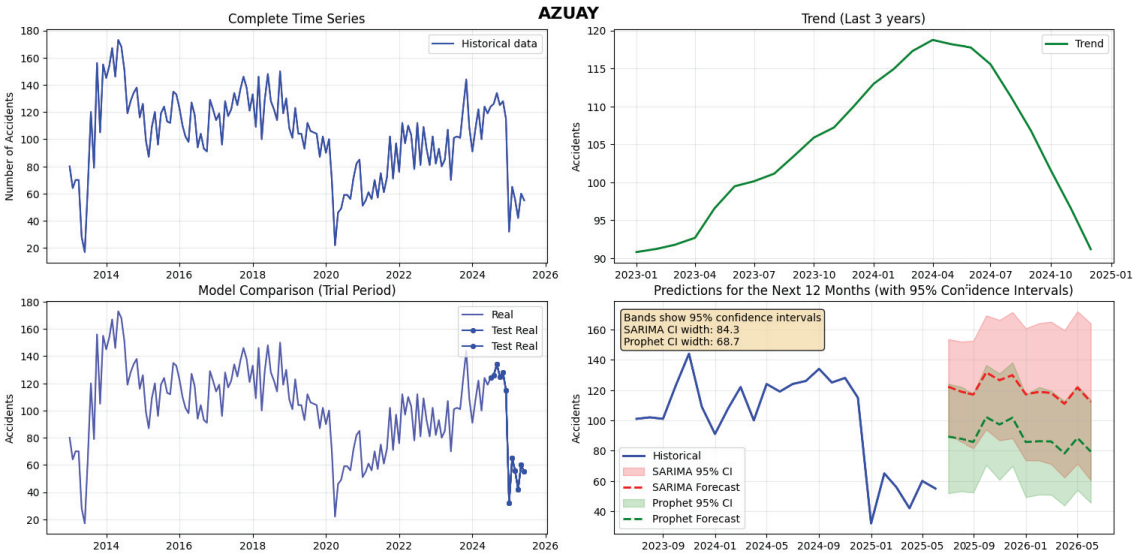


Figure 11. Traffic Accident Forecasting and Temporal Trends in Azuay Province.

In our model evaluation, the SARIMA model demonstrated exceptional performance, with a very low prediction error, significantly outperforming the Prophet model.

Looking ahead, the SARIMA model forecasts a relatively stable trend for the next 12 months, with predicted accidents ranging between 79 and 105 per month. A noticeable peak of 105 accidents is anticipated for December 2025 (see Figure 12).

The accident data in Imbabura reveals a complex picture, with 8850 total accidents and a monthly average of 59, but with high variability: ranging from 13 to 236 accidents in different months, suggesting the influence of extraordinary factors.

https://doi.org/10.3390/computation14010005

When comparing predictive models, SARIMA achieved an error margin of 41%, which, while high, significantly outperforms Prophet's 77% error, making it the best available option, though it should be used with caution.

Projections for the next year estimate between 37 and 51 monthly accidents, generally below the historical average. It is recommended to investigate the causes of historical peaks and explore complementary methods to improve predictability (see Figure 13).

When analyzing Loja, a total of 7844 accidents were found with a monthly average of 52.3. However, monthly figures have varied dramatically—from just 6 accidents in the calmest month to 205 in the most critical one—indicating exceptional events in the province's history.

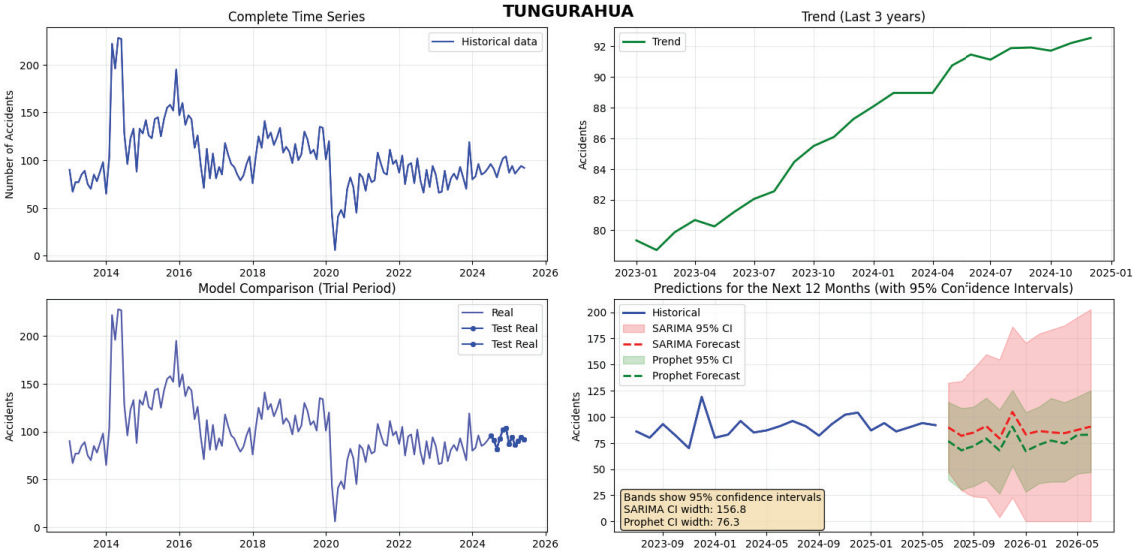


Figure 12. Traffic Accident Forecasting and Temporal Trends in Tungurahua Province.

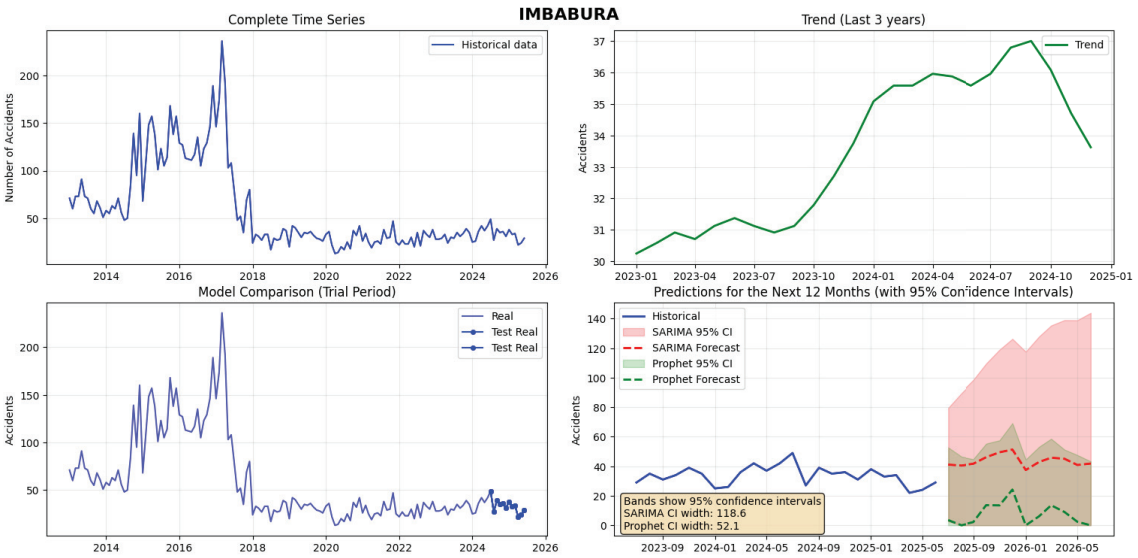


Figure 13. Traffic Accident Forecasting and Temporal Trends in Imbabura Province.

The good news is that the data shows consistent patterns over time, making accident prediction more reliable. When comparing forecasting methods, both models perform acceptably, with Prophet having a slight advantage, predicting with a 16.4% error compared to SARIMA's 18.07%.

https://doi.org/10.3390/computation14010005

Projections for the next year indicate between 47 and 65 monthly accidents, remaining close to the historical average. It is observed that December would be the most critical month with 65 projected accidents, while January would show the lowest activity with 48. This stability in the predictions gives us greater confidence for preventive planning (see Figure 14).

Chimborazo recorded 7342 accidents with a monthly average of 48.9, showing moderate variability between 10 and 89 monthly accidents. Although the series is stationary (p-value: 0.001), a condition that normally favors prediction, both models performed poorly. SARIMA had an error of 67.48% and Prophet 66.55%, levels that are unacceptable for practical use. SARIMA projections for the next 12 months range between 26–47 accidents, below the historical average. The contradiction between stationarity and the high errors suggests that the models are not capturing non-linear patterns, dominant external factors, or possible random behavior in the data (see Figure 15).

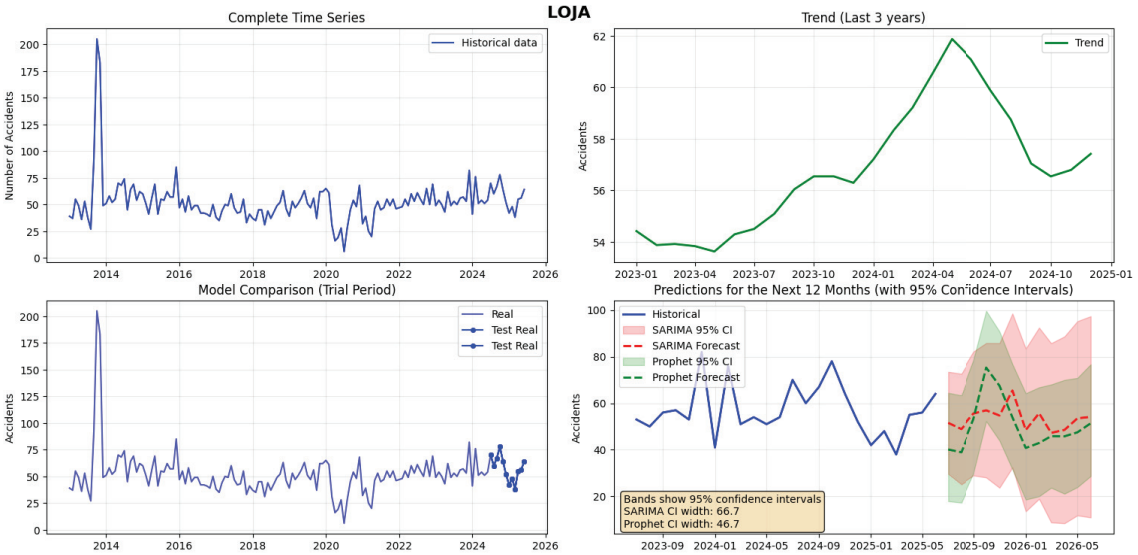


Figure 14. Traffic Accident Forecasting and Temporal Trends in Loja Province.

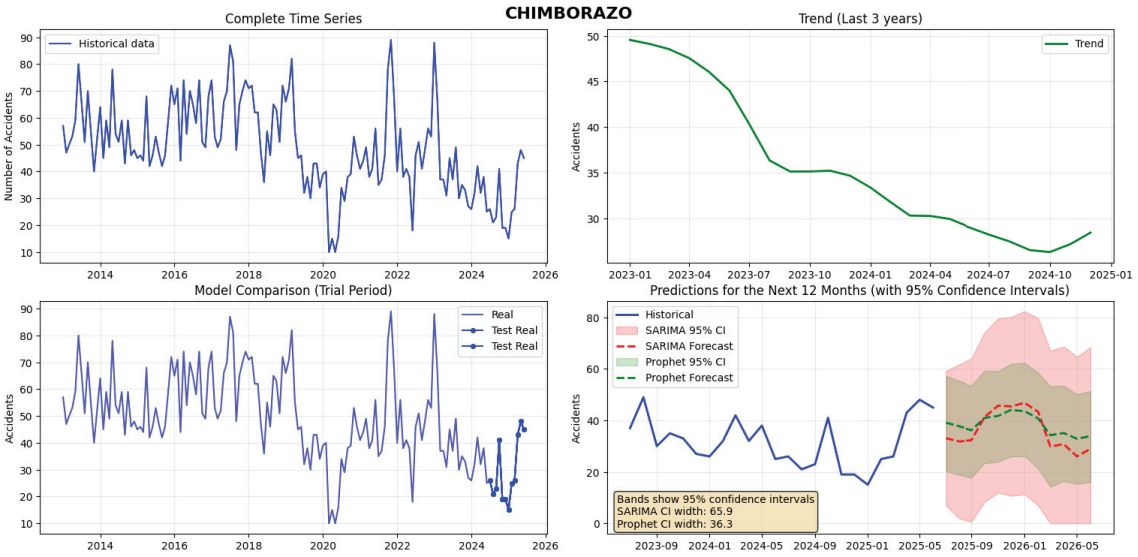


Figure 15. Traffic Accident Forecasting and Temporal Trends in Chimborazo Province.

Cañar recorded 1993 accidents in total, making it the province with the lowest volume analyzed. Its monthly average is 13.3 accidents, but with significant relative variability,

https://doi.org/10.3390/computation14010005

having fluctuated between 1 and 38 accidents per month. The series is not stationary (p-value: 0.688), which complicates predictive modeling.

When comparing the models, SARIMA showed an error of 42.53%, marginally outperforming Prophet (46.76%), with both showing low absolute errors. The performance is acceptable considering the reduced scale of the data.

SARIMA projections for the next 12 months predict between 3 and 6 monthly accidents, well below the historical average, with the highest values in October and December 2025 (6 accidents) and the lowest in January and June 2026 (3 accidents) (see Figure 16).

Cotopaxi recorded 4371 accidents with a monthly average of 29.1, showing high variability between 4 and 85 monthly accidents. The series is not stationary (p-value: 0.132). Both predictive models failed severely: SARIMA showed an extreme error of 269.65%, while Prophet, although better, maintained a critical error of 100.83%. The SARIMA projections (64–79 monthly accidents) lack credibility as they double or triple the historical average (see Figure 17).

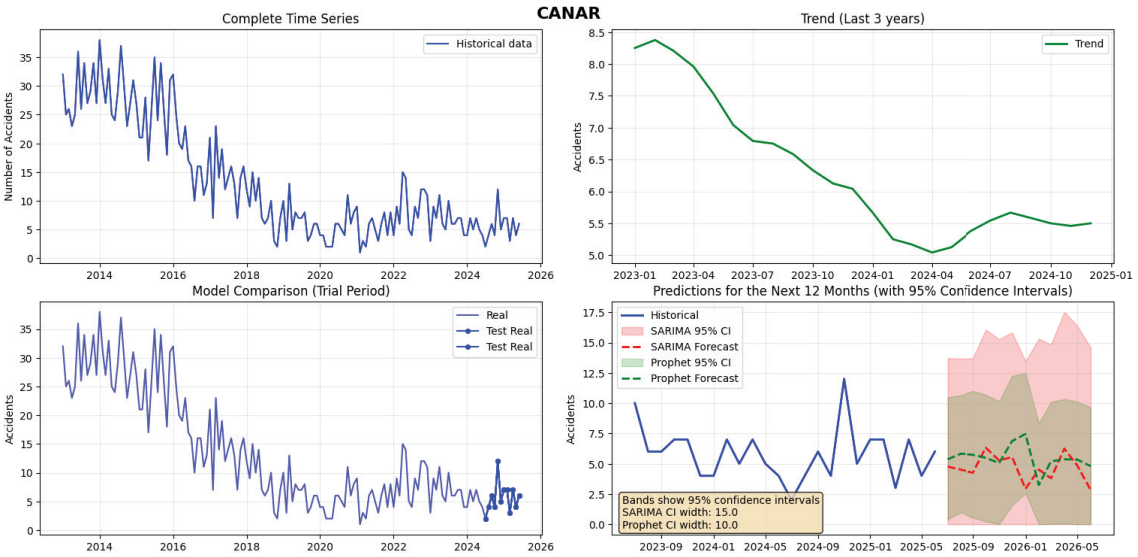


Figure 16. Traffic Accident Forecasting and Temporal Trends in Cañar Province.

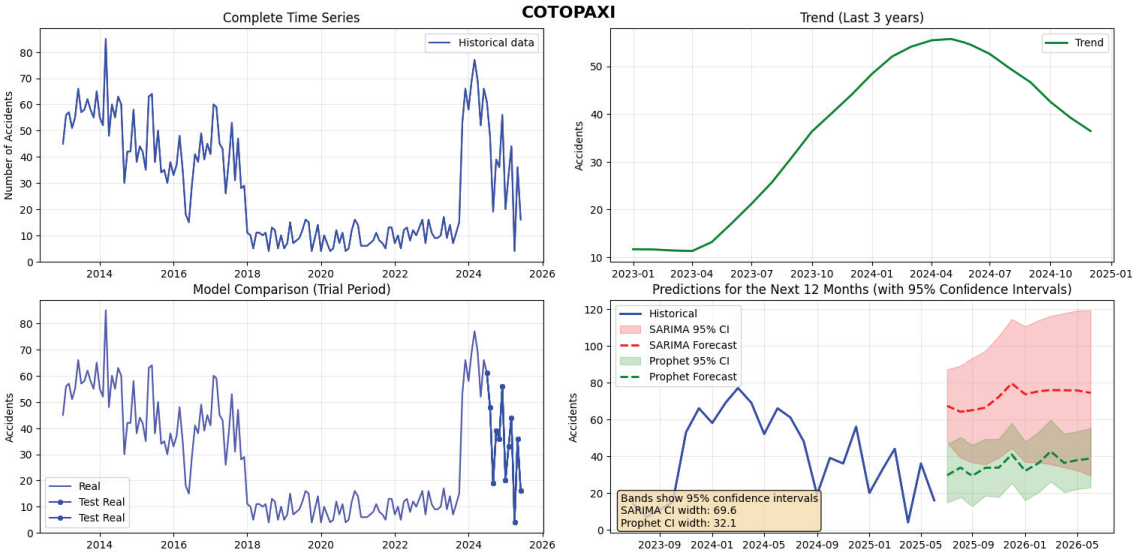


Figure 17. Traffic Accident Forecasting and Temporal Trends in Cotopaxi Province.

https://doi.org/10.3390/computation14010005

Bolívar recorded 1787 accidents with a monthly average of 11.9, showing controlled variability between 0 and 27 monthly accidents. The series is not stationary (p-value: 0.174). SARIMA demonstrated better performance (MAPE: 40.72%, RMSE: 5.03) than Prophet (MAPE: 43.11%, RMSE: 7.20), showing clear superiority in both metrics. SARIMA projections for the next 12 months range between 9–17 accidents, aligning with the historical average, with the highest peak in May 2026 (17 accidents) and the lowest values in August 2025 and March 2026 (9 accidents). The results indicate a favorable scenario with recognizable seasonal patterns and stable projections that enable reliable planning (see Figure 18).

Carchi recorded 1293 accidents with a very low monthly average of 8.6, ranging between 0 and 25 monthly accidents. The series is not stationary. A critical technical failure was identified in both models: the MAPE values are numerically absurd (trillions of percent), indicating a computational error, possibly due to division by zero or corrupted data. However, the RMSE is low (2.47–2.97) and the SARIMA projections (0–6 monthly accidents) are consistent with the low historical volume, although well below the average (see Figure 19).

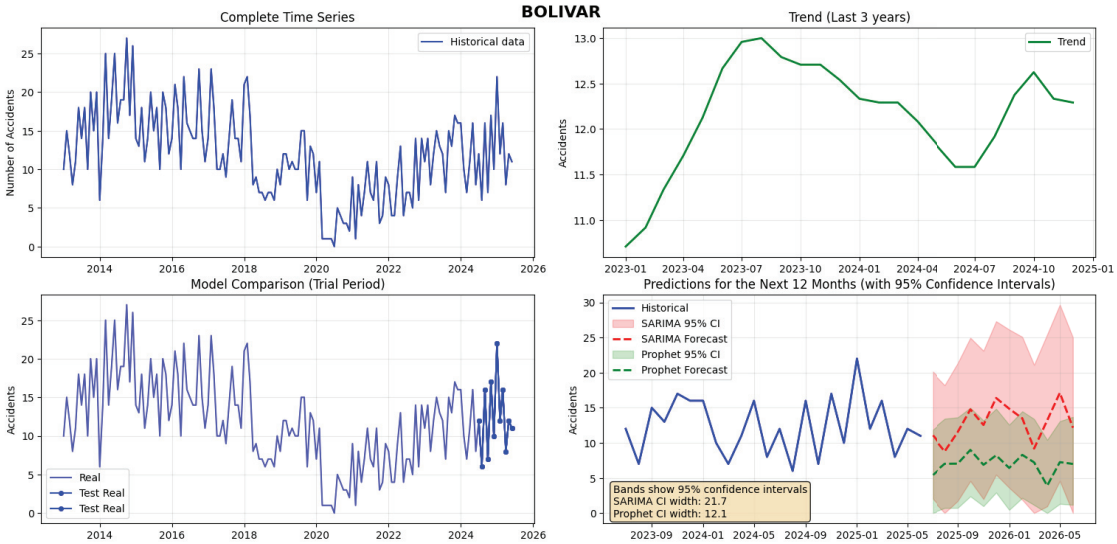


Figure 18. Traffic Accident Forecasting and Temporal Trends in Bolivar Province.

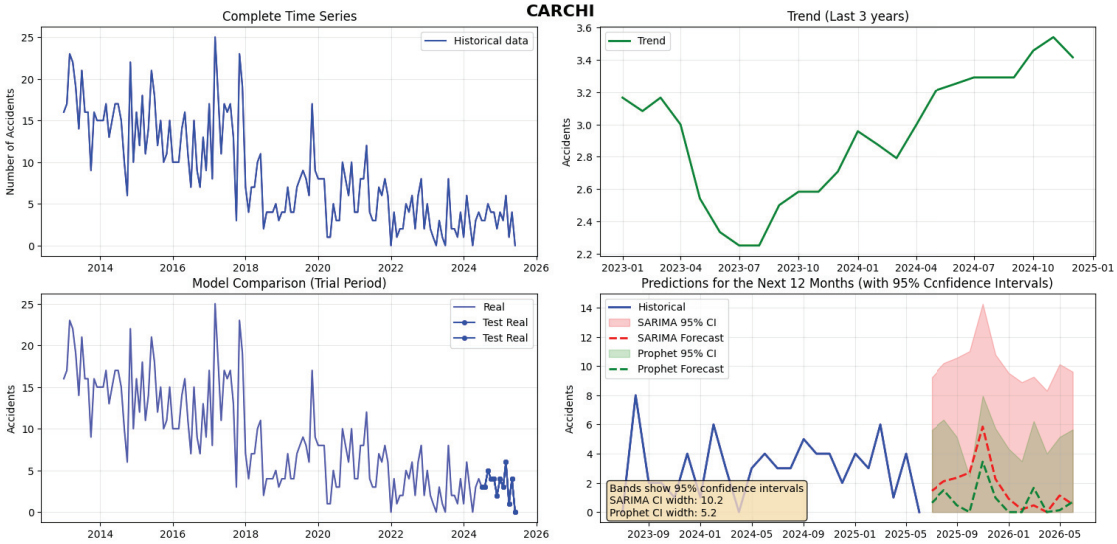


Figure 19. Traffic Accident Forecasting and Temporal Trends in Carchi Province.

https://doi.org/10.3390/computation14010005

The analysis of the provinces shows that Pichincha has the highest accident volume (90,636), while Carchi has the lowest (1293). SARIMA demonstrated better performance in high-volume provinces like Pichincha (MAPE 13.38%) and Tungurahua (6.07%), while Prophet was superior in Azuay (48.34%) and Loja (16.40%). Most series are not stationary, except for Loja and Chimborazo. Predictive errors vary significantly, from an excellent 6.07% in Tungurahua to a critical 100.83% in Cotopaxi, with lower-volume provinces presenting greater predictive challenges with MAPE generally above 40%. (see Table 4).

4.3. Amazon Region

Morona Santiago recorded 1553 accidents with a monthly average of 11.3, showing moderate variability between 1 and 26 monthly accidents. The series is not stationary (p-value: 0.324). Prophet demonstrated better performance (MAPE: 43.36%, RMSE: 5.73) than SARIMA (MAPE: 52.62%, RMSE: 6.16), clearly outperforming it in both metrics. The SARIMA projections for the next 12 months range between 3–8 accidents, well below the historical average, with the highest peak in December 2025 (8 accidents) and the lowest values in January and April 2026 (3 accidents) (see Figure 20).

Table 4. Comparative Performance of Forecasting Models in Andean Region Provinces

Province	Total Accidents	Monthly Average	Best Model	Best Model MAPE	Stationarity
PICHINCHA	90,636	604.2	SARIMA	13.38%	Non-stationary
AZUAY	15,398	102.7	**Prophet**	48.34%	Non-stationary
TUNGURAHUA	1528	101.9	SARIMA	**6.07%**	Non-stationary
IMBABURA	885	59.0	SARIMA	41.12%	Non-stationary
LOJA	7844	52.3	**Prophet**	16.40%	Stationary
CHIMBORAZO	7342	48.9	**Prophet**	66.55%	Stationary
COTOPAXI	4371	29.1	**Prophet**	100.83%	Non-stationary
CAÑAR	1993	13.3	SARIMA	42.53%	Non-stationary
BOLÍVAR	1787	11.9	SARIMA	40.72%	Non-stationary
CARCHI	1293	8.6	SARIMA	43.51%	Non-stationary

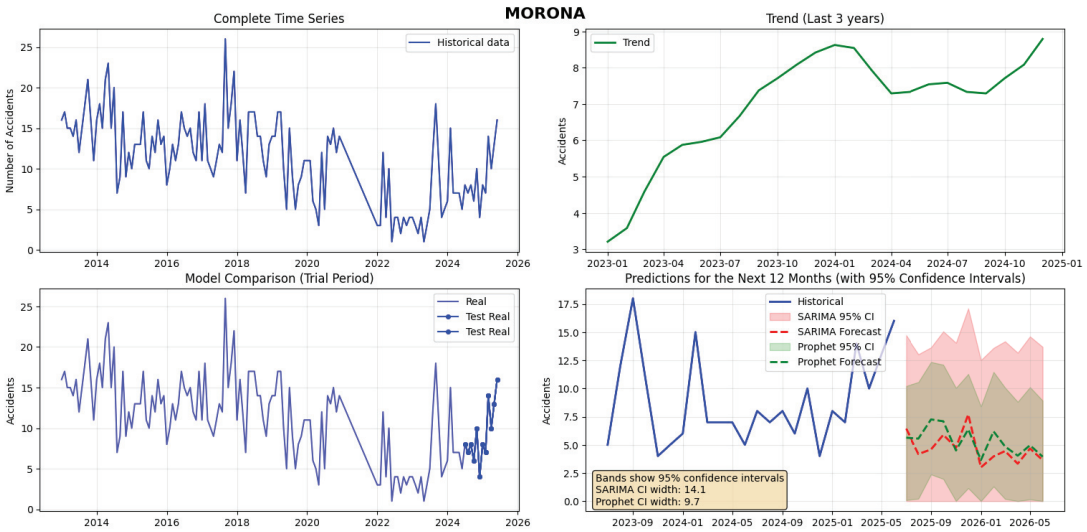


Figure 20. Traffic Accident Forecasting and Temporal Trends in Morona Santiago Province.

https://doi.org/10.3390/computation14010005

Orellana recorded 1175 accidents with a very low monthly average of 7.8, showing high relative variability between 0 and 29 monthly accidents. The series is stationary (p-value: 0.012), a favorable characteristic. SARIMA demonstrated better performance (MAPE: 31.86%, RMSE: 2.27) than Prophet (MAPE: 37.87%, RMSE: 3.00), consistently outperforming it in both metrics. The SARIMA projections for the next 12 months range between 3–9 accidents, close to the historical average, with the highest peak in November 2025 (9 accidents) and the lowest values in August 2025 and June 2026 (3 accidents) (see Figure 21).

Zamora Chinchipe recorded 1143 accidents with a very low monthly average of 7.6, showing significant variability between 0 and 25 monthly accidents. The series is not stationary (p-value: 0.251). Prophet demonstrated better performance (MAPE: 52.55%) than SARIMA (MAPE: 68.58%), although both models show high errors. The SARIMA projections for the next 12 months range between 2–9 accidents, close to the historical average, with peaks in October and November 2025 (9 accidents) and the lowest value in September 2025 (2 accidents) (see Figure 22).

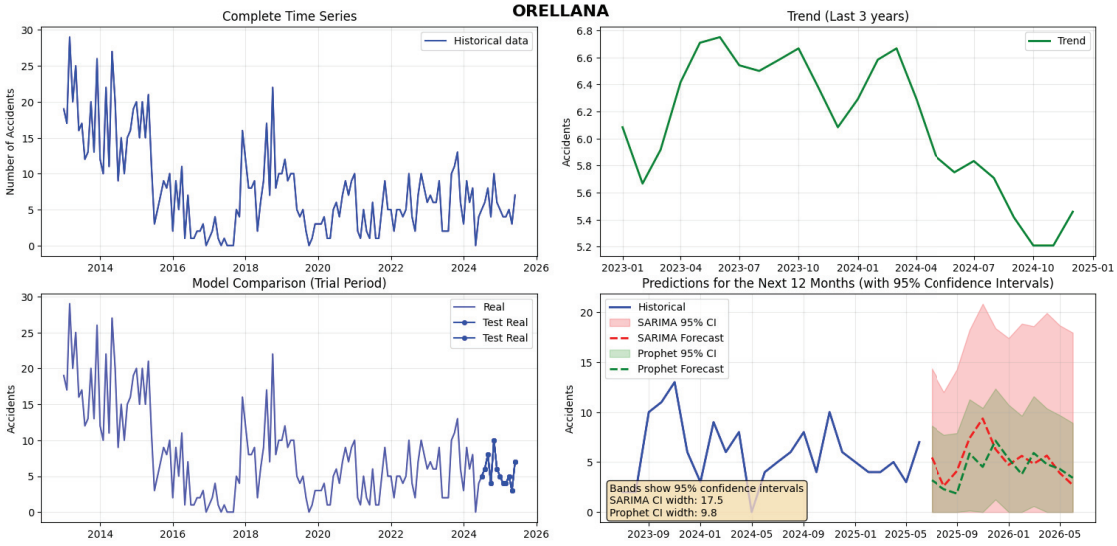


Figure 21. Traffic Accident Forecasting and Temporal Trends in Orellana Santiago Province.

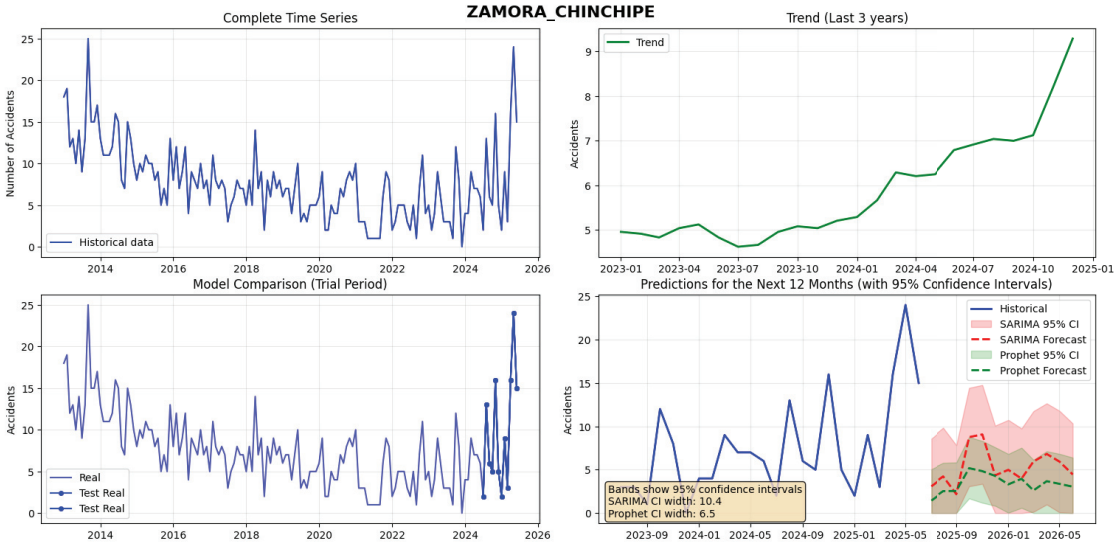


Figure 22. Traffic Accident Forecasting and Temporal Trends in Zamora Chinchipe Province.

https://doi.org/10.3390/computation14010005

Napo recorded 1091 accidents with an extremely low monthly average of 7.3, showing high variability between 0 and 35 monthly accidents. The series is not stationary (p-value: 0.383). SARIMA marginally outperformed Prophet with a MAPE of 72.63% versus 79.10%, although both models showed poor performance. The SARIMA projections for the next 12 months range between 0–4 accidents, well below the historical average, with the highest peak in December 2025 (4 accidents) and the lowest value in September 2025 (0 accidents) (see Figure 23).

Pastaza recorded 1041 accidents with an extremely low monthly average of 6.9, showing high variability between 0 and 28 monthly accidents. The series is not stationary. A critical computational failure was identified in both models, with numerically impossible MAPE values (trillions of percent), which completely invalidates this metric. However, the RMSE is very low (1.49–1.57) and the SARIMA projections (0–3 monthly accidents) are consistent with the low historical volume, although well below the average. The predictions show maximum values in December 2025 and February 2026 (3 accidents) and the minimum in August 2025 (0 accidents) (see Figure 24).

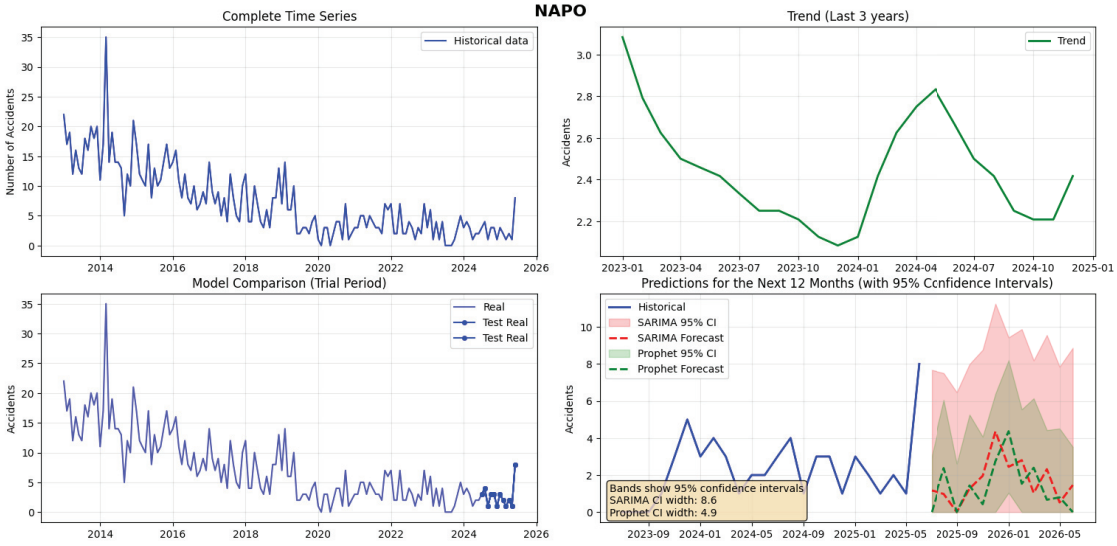


Figure 23. Traffic Accident Forecasting and Temporal Trends in Napo Province.

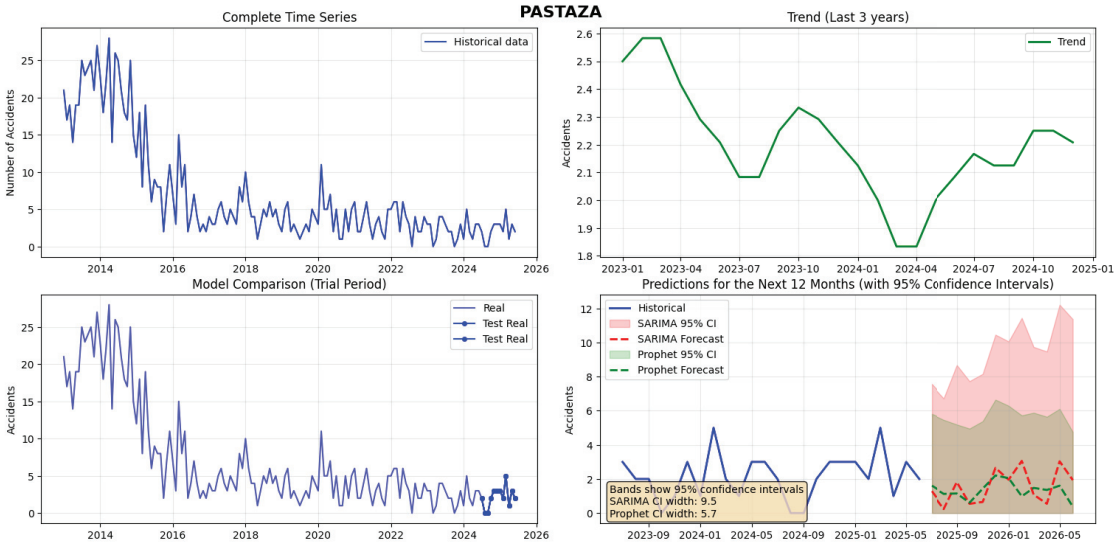


Figure 24. Traffic Accident Forecasting and Temporal Trends in Pastaza Province.

https://doi.org/10.3390/computation14010005

Sucumbíos recorded 828 accidents, the lowest volume analyzed, with a monthly average of 5.5 and high variability between 0 and 23 monthly accidents. The series is not stationary. Both models showed critical failures in MAPE calculation with numerically absurd values (billions of percent), following the error pattern observed in other low-volume provinces. The SARIMA projections range between 0–4 monthly accidents, with 7 of the 12 months projecting 0 accidents and only December 2025 showing a significant value (4 accidents), a trend well below the historical average (see Figure 25).

Among the analyzed Amazonian provinces, Morona Santiago recorded the highest accident volume (1553) while Sucumbíos had the lowest (828). Orellana was the only province with a stationary series, where SARIMA achieved the best performance (MAPE 31.86%). In the other non-stationary provinces, Prophet showed an advantage in Morona Santiago (43.36%) and Sucumbíos (533.73%), although the latter presents an extremely high error. Predictive errors vary dramatically, from the acceptable 31.86% in Orellana to critically high values in Pastaza (1703.95%) and Sucumbíos (533.73%), reflecting the challenges of modeling very low-volume series (see Table 5).

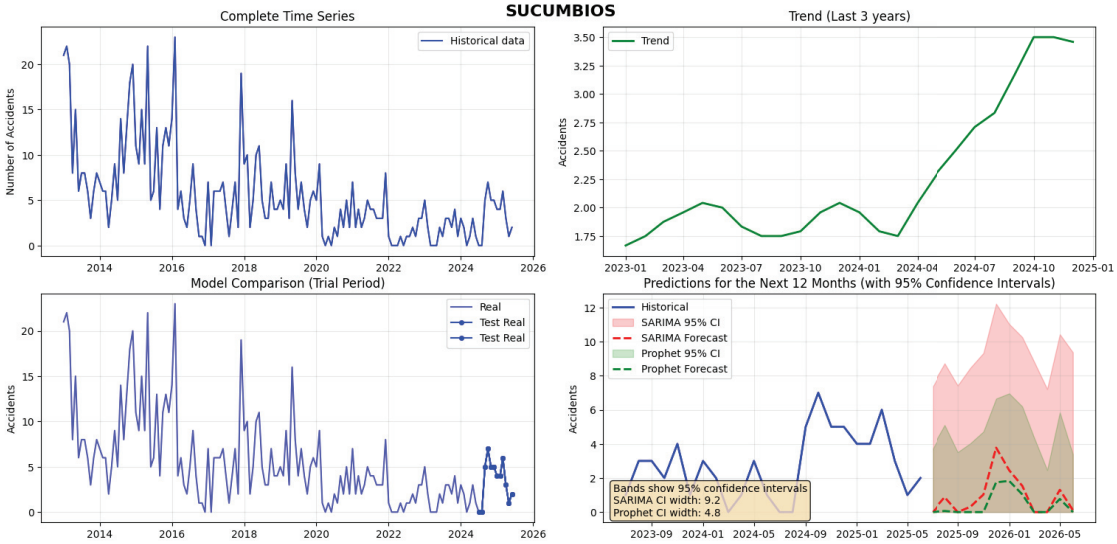


Figure 25. Traffic Accident Forecasting and Temporal Trends in Sucumbios Province.

Table 5. Comparative Performance of Forecasting Models in Amazon Region Provinces.

Province	Total Accidents	Monthly Average	Best Model	Best Model MAPE	Stationarity
MORONA SANTIAGO	1553	11.3	**Prophet**	43.36%	Non-stationary
ORELLANA	1175	7.8	SARIMA	**31.86%**	Stationary
ZAMORA CHINCHIPE	1143	7.6	**Prophet**	52.55%	Non-stationary
NAPO	1091	7.3	SARIMA	72.63%	Non-stationary
PASTAZA	1041	6.9	SARIMA	1703.95%	Non-stationary
SUCUMBÍOS	828	5.5	**Prophet**	533.73%	Non-stationary

4.4. Insular Region

The Insular Region (Galápagos) has the lowest accident volume in the country, with a monthly average of only 0.5. Its time series is not stationary and the MAPE values are

https://doi.org/10.3390/computation14010005

extremely high and unstable, showing numerically absurd figures that confirm a systematic error pattern in provinces with minimal volumes. This MAPE failure appears to be related to the high frequency of zero or near-zero values in the data (see Figure 26).

However, the RMSE values are very low (0.56–0.62), indicating that the numerical predictions are accurate in scale. The projections of 0–1 monthly accidents are consistent with historical data, showing a random distribution of zeros and ones without a clear seasonal pattern. The projected average of 0.33 accidents per month is slightly lower than the historical average of 0.5 (see Table 6).

Table 6. Comparative Performance of Forecasting Models in Galapagos Region Provinces.

Province	Total Accidents	Monthly Average	Best Model	Best Model MAPE	Stationarity
GALÁPAGOS	81	0.5	**Prophet**	3302.35%	Non-stationary

Despite the Prophet model registering the highest individual accuracy (4.91% MAPE in Guayas), the analysis reveals a Global Tie (12 vs. 12) in the number of provinces where each model is superior according to the MAPE.

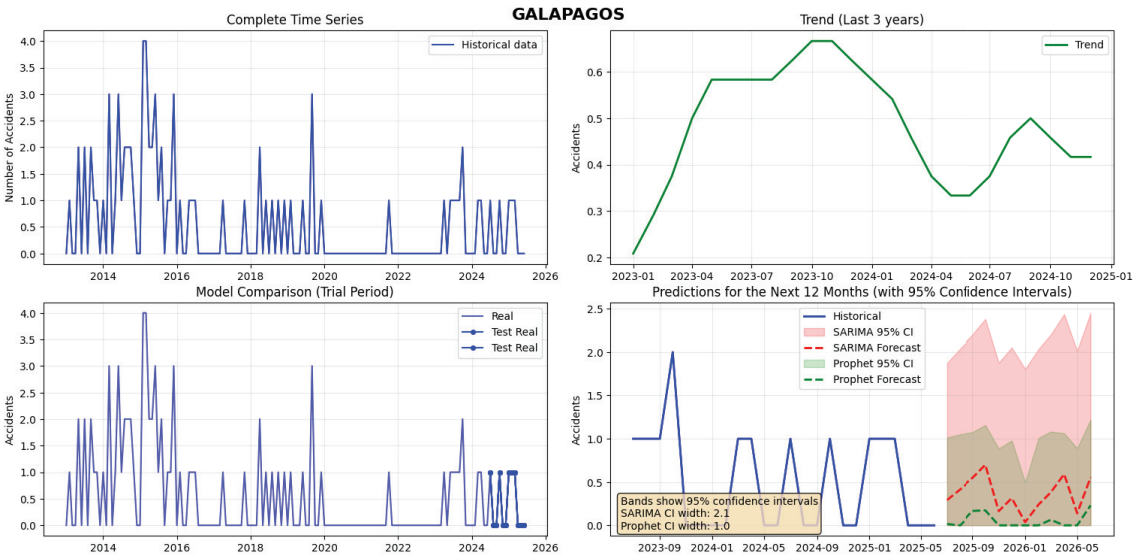


Figure 26. Traffic Accident Forecasting and Temporal Trends in Galapagos Province.

SARIMA is more effective in the Sierra Region (provinces with Non-Stationary series but with medium-to-high volume, such as Pichincha and Tungurahua). Prophet is more effective in the Coast Region (provinces with high volume and stationary series, such as Guayas and Sto. Domingo de los Tsáchilas).

There is no single general "Best Model" when counting the victories per province; both models are equally competitive, with the selection of the optimal model depending on the intrinsic characteristics (stationarity, volume, volatility) of each province's time series (see Table 7).

The accident analysis compared the SARIMA and Prophet models for all 24 provinces, most having 150 monthly records, selecting the best model for each. Guayas has the highest accident volume (101,179 total, 674.5 monthly), and Galápagos the lowest (81 total, 0.5 monthly).The best overall precision was achieved with Prophet in Guayas ($MAPE$: 4.9%), while SARIMA's best precision was in Tungurahua ($MAPE$: 6.1%). Only 7 provinces were classified as stationary time series (including Guayas, Manabí, and Sto. Domingo),

https://doi.org/10.3390/computation14010005

while 17 provinces were non-stationary. The model choice was critical: Pichincha strongly favored SARIMA (13.4% MAPE), and Guayas favored Prophet (4.9% MAPE) (Figure 27).

Table 7. Comparative Analysis of Forecasting Model Performance by Geographic Region.

Region	Prophet Wins	SARIMA Wins	Dominant Model
Coast	4	3	Prophet
Highlands	4	6	SARIMA
Amazon	3	3	Tie
Insular	1	0	Prophet
OVERALL TOTAL (24 Provinces)	12	12	Tie

4.5. Diebold–Mariano Test Results

The Final Table 8 presents the results of the Diebold–Mariano (DM) test applied to each province of Ecuador, with the purpose of evaluating whether statistically significant differences exist between the two forecasting models used to estimate accident time series. The DM test directly compares the forecast errors generated by both models and determines whether one model systematically outperforms the other.

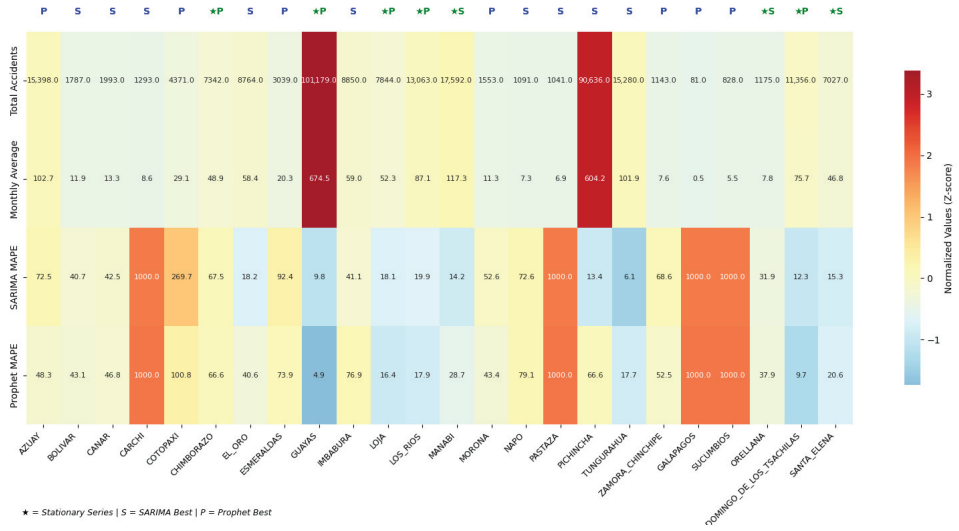


Figure 27. Results of Prophet and SARIMA modeling of accident-prone traffic for provinces in Ecuador.

For each province, three key elements are reported:

- **dm_stat**: the DM test statistic, which quantifies the magnitude and direction of the difference in predictive accuracy between the models.
- **dm_p**: the associated *p*-value, reflecting the probability that the observed difference in errors occurred by chance.
- **significance**: a categorical interpretation based on the *p*-value that facilitates the identification of statistically relevant differences.

For interpretation, the following standard criteria were adopted:

- $p \geq 0.10$: *No evidence* , indicating no statistical support for differences in model performance.
- $0.05 \leq p < 0.10$: *Weak evidence* (not present in the analyzed data).
- $0.01 \leq p < 0.05$: *Significant*, indicating statistically significant evidence favoring one model.

https://doi.org/10.3390/computation14010005

- $p < 0.01$: *Highly significant*, reflecting very strong evidence that one model consistently outperforms the other.

The results indicate that, for most provinces, the p-values are relatively high ($p \geq 0.10$), meaning that no statistical evidence exists to claim that one model predicts better than the other. However, several provinces exhibit clearly significant differences. Specifically, Cotopaxi, El Oro, Guayas, Imbabura, Morona, Pichincha, and Tungurahua show p-values below 0.01, providing highly significant evidence of superior performance by one of the models. Additionally, three provinces—Zamora Chinchipe, Sucumbíos, and Santa Elena—show statistically significant differences at the 5% level.

Overall, the results reveal a geographically heterogeneous pattern. While provinces in the central-northern Sierra and parts of the Coastal region present pronounced differences in model performance, other regions exhibit statistically indistinguishable accuracy between the two models. This heterogeneous behavior suggests that local factors—such as seasonal patterns, regional variability, or specific characteristics of the time series—may influence the relative predictive performance of the models.

Table 8. Diebold–Mariano Test Statistics for Forecast Accuracy Comparison Across Ecuadorian Provinces.

Province	**dm_stat**	***p*-Value**	**Significance**
AZUAY	1.635274	0.10199	No evidence
BOLIVAR	−1.416653	0.15658	No evidence
CANAR	0.127985	0.89816	No evidence
CARCHI	−1.351865	0.17641	No evidence
COTOPAXI	3.674933	0.00023791	Highly significant ($p < 0.01$)
CHIMBORAZO	1.288091	0.19771	No evidence
EL_ORO	−3.435728	0.00059096	Highly significant ($p < 0.01$)
ESMERALDAS	0.630382	0.52844	No evidence
GUAYAS	3.465814	0.00052862	Highly significant ($p < 0.01$)
IMBABURA	−2.857548	0.0042692	Highly significant ($p < 0.01$)
LOJA	−0.380962	0.70323	No evidence
LOS_RIOS	1.281541	0.2	No evidence
MANABI	−0.431551	0.66606	No evidence
MORONA	3.699555	0.00021597	Highly significant ($p < 0.01$)
NAPO	−0.856829	0.39153	No evidence
PASTAZA	0.426604	0.66966	No evidence
PICHINCHA	−6.766831	1.3163×10^{-11}	Highly significant ($p < 0.01$)
TUNGURAHUA	−4.927065	8.3474×10^{-7}	Highly significant ($p < 0.01$)
ZAMORA_CHINCHIPE	−2.054503	0.039927	Significant ($p < 0.05$)
GALAPAGOS	−0.739158	0.45981	No evidence
SUCUMBIOS	−2.225582	0.026042	Significant ($p < 0.05$)
ORELLANA	−1.203761	0.22868	No evidence
STO_D_TSACHILAS	1.575166	0.11521	No evidence
SANTA_ELENA	−2.078903	0.037626	Significant ($p < 0.05$)

https://doi.org/10.3390/computation14010005

5. Discussion

The empirical comparison between the SARIMA and Prophet models enabled us to address the research question, showing that the predictive effectiveness of both approaches depends strongly on the regional context and the temporal behavior of the accident series. Although the results revealed an overall balance—12 provinces favored each model—SARIMA achieved a slight advantage in national average accuracy (MAPE: 12.3% versus 15.7% for Prophet). This finding rules out the existence of a universally superior model for road-accident prediction in Ecuador and indicates that the suitability of each model is closely linked to the local characteristics of each province.

The regional analysis identified consistent patterns explaining the differences in model performance. Provinces with a high volume of accidents, such as Guayas and Pichincha, generate denser and more structurally coherent time series, facilitating model fitting. In these cases, the choice between SARIMA and Prophet is closely related to the stationarity of the series. Prophet showed outstanding results in coastal provinces with high volume and stationary behavior—for example, Guayas, where it achieved a MAPE of 4.91%—due to its ability to model complex seasonal patterns without requiring differencing [41]. In contrast, SARIMA outperformed Prophet in Andean provinces with medium-to-high volumes but non-stationary series, such as Pichincha (MAPE: 13.38%) and Tungurahua (MAPE: 6.07%). Its methodological structure, which incorporates differencing to achieve stationarity, enables it to more effectively handle the nonlinear trends and structural shifts characteristic of these regions [42].

These differences are not explained solely by the mathematical structure of the models but also by underlying geographic and socioeconomic factors. In the coastal region, dense traffic, moderate climate variability, and more homogeneous economic activity generate marked and relatively stable seasonal patterns, conditions that favor Prophet. In contrast, the Andean region exhibits complex topography, higher climatic variability, and a more heterogeneous road network, factors that introduce irregularities, abrupt peaks, and less predictable trends. In these settings, the parametric structure of SARIMA provides a clear advantage in the presence of potential structural breaks [43]. These findings reinforce the importance of considering contextual variables—such as topography, climate, and traffic dynamics—when selecting predictive models tailored to geographic reality.

Despite the overall strong performance, the study faced methodological limitations in provinces with very low data volume and a high frequency of zero or near-zero values, such as Galápagos, Carchi, Pastaza, and Sucumbíos. In such cases, MAPE becomes mathematically unstable, as its formulation disproportionately magnifies small absolute errors when real values are very low. This can produce artificially inflated percentage errors that do not reflect the true performance of the model. However, RMSE values in these provinces remained consistently low, indicating that the limitations stem from the metric rather than from the predictive models themselves. Therefore, conclusions based solely on MAPE for these provinces should be interpreted with caution.

For future analyses in low-incident contexts, alternative metrics less sensitive to zero values are recommended, such as the Mean Absolute Error (MAE) or the Symmetric Mean Absolute Percentage Error (sMAPE). These indicators provide a more stable basis for comparison across regions with extreme heterogeneity in data volume [44].

From an applied perspective, the results offer valuable guidance for road-safety policy, supporting a model-selection approach tailored to regional needs. Prophet is more suitable for urban coastal provinces with stable seasonal patterns, whereas SARIMA is preferable in Andean regions where volatile and structurally complex trends predominate.

Finally, several avenues for future research arise from these findings: developing hybrid models that combine the parametric robustness of SARIMA with the decomposition

https://doi.org/10.3390/computation14010005

flexibility of Prophet; incorporating exogenous variables—such as weather conditions, traffic flow, or economic indicators—to improve explanatory power; extending the analysis to sub-provincial scales; and comparing these results with deep learning architectures, such as LSTM and Transformers, to further validate predictive performance in complex transportation systems [45].

6. Conclusions

The present study aimed to rigorously evaluate the predictive performance of the SARIMA and Prophet models in estimating traffic accidents across Ecuador's 24 provinces, with the objective of determining whether a universally superior model exists or whether performance depends on regional conditions. This objective addressed the central research question concerning the contextual nature of prediction within heterogeneous transportation systems, considering both the geographic particularities and the structural variability of the time series associated with road accidents.

The results revealed that no single model consistently outperformed its counterpart across the entire national territory, with a global tie observed in performance between both approaches (12 provinces favoring SARIMA and 12 favoring Prophet). However, the regional analysis demonstrated distinct patterns of effectiveness: Prophet exhibited outstanding performance in coastal and urban provinces with high density and stationary behavior, such as Guayas, which achieved the lowest MAPE in the study (4.91%), while SARIMA excelled in the Andean region, particularly in provinces with non-stationary series and more pronounced structural variability, such as Tungurahua (MAPE 6.07%) and Pichincha. These findings confirm that the predictive accuracy of the models is strongly conditioned by the geographic and structural dynamics of the data, invalidating the hypothesis of a single, universally optimal model for traffic accident forecasting.

From a theoretical perspective, the study contributes to closing a gap in the literature by providing the first systematic comparison between a classical statistical model (SARIMA) and a machine learning-based model (Prophet) applied to the Ecuadorian context, characterized by high topographic and demographic heterogeneity. The results offer strong evidence that the process of predictive model selection must incorporate criteria related to localization and series structure, beyond global accuracy alone. In this sense, the differentiated performance of each model across regions reinforces the notion that geographical conditions, urbanization levels, and the intrinsic seasonality of the data determine the methodological suitability of each approach.

Despite the robustness of the results obtained in high-volume provinces such as Guayas and Tungurahua, the study acknowledged important methodological limitations, particularly concerning low-volume series in provinces such as Galápagos, Carchi, Pastaza, and Sucumbíos. In these cases, the MAPE metric exhibited severe computational failures, generating numerically implausible values due to the high frequency of zeros and the sensitivity of the formula's denominator. Nonetheless, the RMSE remained low, suggesting that the issue lies within the metric itself rather than the underlying model. These observations highlight the need to reassess traditional accuracy metrics in low-incidence contexts and propose the adoption of more robust indicators for interprovincial comparison.

From a practical perspective, the findings provide a valuable methodological guide for the design of road safety policies and the allocation of predictive resources tailored to regional contexts. Rather than adopting a single model for the entire country, a localized model selection framework is recommended—one that leverages the specific advantages of SARIMA in non-stationary environments and Prophet in urban and stable settings. Likewise, future research should integrate exogenous variables such as weather conditions, economic indicators, or mobility data to enhance the explanatory capacity of the models.

https://doi.org/10.3390/computation14010005

The development of hybrid or ensemble frameworks that combine the strengths of both approaches represents a promising direction for improving multiregional forecasting.

Overall, this study reaffirms the importance of understanding the contextual nature of predictive modeling in complex transportation systems. By demonstrating that model performance depends on the structural and geographical characteristics of each province, the research provides a solid empirical and methodological foundation for the adoption of adaptive forecasting approaches. Beyond its technical contribution, the results underscore the relevance of predictive analytics as a strategic tool for accident prevention and evidence-based policymaking, strengthening the link between data science and the intelligent management of road safety.

Author Contributions: Conceptualization, T.L.; methodology, R.V. and H.V.-I.; software, A.S. and P.P.-A.; validation, L.N.; investigation, G.S.; Project administration, W.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All data and files related to the predictive models are available and organized in the following Google Drive link: https://drive.google.com/drive/folders/1W3poLPpuC-Tgjqh2VS1s_GMWbASUCJs7?usp=sharing (accessed on 8 December 2025). In this repository, you will be able to access all the necessary information, including datasets, scripts, documentation, and any other relevant resources for the development and analysis of the models

Acknowledgments: We express our sincere gratitude to Wilson Chango from PUCE Esmeraldas for his invaluable technical assistance and guidance in configuring the sensor systems and data acquisition processes for this research. This study was supported by PUCE Esmeraldas. Additionally, we acknowledge the facilities and general support provided by the Department of Computer Sciences, which greatly contributed to the successful completion of this work.

Conflicts of Interest: The authors declare that they have no conflicts of interest related to this article.

References

1. Mohmed, A.; Allah, G. Incidence and Causes of Injury Deaths among Children under Five Years of Age in El-Giza-Egypt. *J. Med. Pharm. Sci.* **2019**, *4*, 49–56.
2. Kumar, R.; Bhanu, M.; Mendes-Moreira, J.; Chandra, J. Spatio-Temporal Predictive Modeling Techniques for Different Domains: A Survey. *ACM Comput. Surv.* **2024**, *57*, 38. [CrossRef]
3. Rabbani, M.B.A.; Musarat, M.A.; Alaloul, W.S.; Rabbani, M.S.; Maqsoom, A.; Ayub, S.; Bukhari, H.; Altaf, M. A Comparison Between Seasonal Autoregressive Integrated Moving Average (SARIMA) and Exponential Smoothing (ES) Based on Time Series Model for Forecasting Road Accidents. *Arab. J. Sci. Eng.* **2021**, *46*, 11113–11138. [CrossRef]
4. Sivaramakrishnan, S.; Rathish, C.R.; Premalatha, S.; Niranjana, C. Introduction to AI Technique and Analysis of Time Series Data Using Facebook Prophet Model. In *Innovative Engineering with AI Applications*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2023; pp. 171–188. [CrossRef]
5. Niohuru, I. Disease Burden and Mortality. In *Healthcare and Disease Burden in Africa*; Springer: Cham, Switzerland, 2023; pp. 35–85. [CrossRef]
6. Liu, L.; Villavicencio, F.; Yeung, D.; Perin, J.; Lopez, G.; Strong, K.L.; Black, R.E. National, regional, and global causes of mortality in 5–19-year-olds from 2000 to 2019: A systematic analysis. *Lancet Glob. Health* **2022**, *10*, e337–e347. [CrossRef]
7. Muelbert, M.M.; Copertino, M.; Cotrim da Cunha, L.; Lewis, M.N.; Polejack, A.; Peña-Puch, A.d.C.; Rivera-Arriaga, E. The Ocean and Cryosphere in a Changing Climate in Latin America: Knowledge Gaps and the Urgency to Translate Science Into Action. *Front. Clim.* **2021**, *3*, 748344. [CrossRef]
8. Balawi, M.; Tenekeci, G. Time series traffic collision analysis of London hotspots: Patterns, predictions and prevention strategies. *Heliyon* **2024**, *10*, e25710. [CrossRef]
9. Holguín-Carvajal, J.P.; Otzen, T.; Sanhueza, A.; Castillo, Á.; Manterola, C.; Muñoz, G.; García-Aguilera, F.; Salgado-Castillo, F. Trends in traffic accident mortality and social inequalities in Ecuador from 2011 to 2022. *BMC Public Health* **2024**, *24*, 1951. [CrossRef] [PubMed]
10. Dong, W.; Liu, Y.; Zhu, W.; Sun, J.; Bai, R. Temporal trends in the incidence and mortality of road injuries in China: Current trends and future predictions. *Injury* **2023**, *54*, 111139. [CrossRef] [PubMed]

https://doi.org/10.3390/computation14010005

11. Rosen, H.E.; Bari, I.; Paichadze, N.; Peden, M.; Khayesi, M.; Monclús, J.; Hyder, A.A. Global road safety 2010–18: An analysis of Global Status Reports. *Injury* **2025**, *56*, 110266. [CrossRef]
12. Seif, M.; Edalat, S.; Majidpour Azad Shirazi, A.; Alipouri, S.; Bayati, M. Prediction of the burden of road traffic injuries in Iran by 2030: Prevalence, death, and disability-adjusted life years. *Chin. J. Traumatol.—Engl. Ed.* **2024**, *27*, 242–248. [CrossRef]
13. Ahmed, S.K.; Mohammed, M.G.; Abdulqadir, S.O.; El-Kader, R.G.; El-Shall, N.A.; Chandran, D.; Rehman, M.E.; Dhama, K. Road traffic accidental injuries and deaths: A neglected global health issue. *Health Sci. Rep.* **2023**, *6*, e1240. [CrossRef]
14. Erfanpoor, S.; Hasani, J.; Mirtorabi, S.; Manouchehri, R.; Hashemi Nazari, S. Trend of mortality rate due to traffic accidents in Iran from 2006 to 2020: A cross-sectional study. *Int. J. Crit. Illn. Inj. Sci.* **2023**, *13*, 73–77. [CrossRef]
15. Arumugam, V.; Natarajan, V. Time Series Modeling and Forecasting Using Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA) Models. *Instrum. Mes. Métrol.* **2023**, *22*, 161–168. [CrossRef]
16. Webel, K. A Review of Some Recent Developments in the Modelling and Seasonal Adjustment of Infra-Monthly Time Series. *SSRN Electron. J.* **2022**. [CrossRef]
17. Ryan, O.; Haslbeck, J.M.; Waldorp, L.J. Non-Stationarity in Time-Series Analysis: Modeling Stochastic and Deterministic Trends. *Multivar. Behav. Res.* **2025**, *60*, 556–588. [CrossRef]
18. Marcillo, P.; Valdivieso Caraguay, Á.L.; Hernández-álvarez, M. A Systematic Literature Review of Learning-Based Traffic Accident Prediction Models Based on Heterogeneous Sources. *Appl. Sci.* **2022**, *12*, 4529. [CrossRef]
19. Yavuz, V.S. Forecasting monthly rainfall and temperature patterns in Van Province, Türkiye, using ARIMA and SARIMA models: A long-term climate analysis. *J. Water Clim. Change* **2025**, *16*, 800–818. [CrossRef]
20. Serrano, A.L.M.; Rodrigues, G.A.P.; Martins, P.H.S.; Saiki, G.M.; Filho, G.P.R.; Gonçalves, V.P.; Albuquerque, R.O. Statistical Comparison of Time Series Models for Forecasting Brazilian Monthly Energy Demand Using Economic, Industrial, and Climatic Exogenous Variables. *Appl. Sci.* **2024**, *14*, 5846. [CrossRef]
21. Gkontzis, A.F.; Kotsiantis, S.; Feretzakis, G.; Verykios, V.S. Temporal Dynamics of Citizen-Reported Urban Challenges: A Comprehensive Time Series Analysis. *Big Data Cogn. Comput.* **2024**, *8*, 27. [CrossRef]
22. Zeng, S.; Liu, C.; Zhang, H.; Zhang, B.; Zhao, Y. Short-Term Load Forecasting in Power Systems Based on the Prophet–BO–XGBoost Model. *Energies* **2025**, *18*, 227. [CrossRef]
23. Szostek, K.; Mazur, D.; Drałus, G.; Kusznier, J. Analysis of the Effectiveness of ARIMA, SARIMA, and SVR Models in Time Series Forecasting: A Case Study of Wind Farm Energy Production. *Energies* **2024**, *17*, 4803. [CrossRef]
24. Chaturvedi, S.; Rajasekar, E.; Natarajan, S.; McCullen, N. A comparative assessment of SARIMA, LSTM RNN and Fb Prophet models to forecast total and peak monthly energy demand for India. *Energy Policy* **2022**, *168*, 113097. [CrossRef]
25. Arslan, S. A hybrid forecasting model using LSTM and Prophet for energy consumption with decomposition of time series data. *PeerJ Comput. Sci.* **2022**, *8*, e1001. [CrossRef]
26. Becker, A.S.; Erinjeri, J.P.; Chaim, J.; Kastango, N.; Elnajjar, P.; Hricak, H.; Vargas, H.A. Automatic Forecasting of Radiology Examination Volume Trends for Optimal Resource Planning and Allocation. *J. Digit. Imaging* **2021**, *35*, 1–8. [CrossRef] [PubMed]
27. Hossain, M.M.; Faisal Anwar, A.H.; Garg, N.; Prakash, M.; Bari, M. Monthly Rainfall Prediction at Catchment Level with the Facebook Prophet Model Using Observed and CMIP5 Decadal Data. *Hydrology* **2022**, *9*, 111. [CrossRef]
28. Arnaut, F.; Cvetkov, V.; Đurić, D.; Samardžić-Petrović, M. Short-term forecasting of PM10 and PM2.5 concentrations with Facebook's Prophet Model at the Belgrade-Zeleno brdo. *Geofizika* **2023**, *40*, 162–177. [CrossRef]
29. Mariño, J.A.; Arrieta-Prieto, M.E.; Calderón V, S.A. Comparison between statistical models and machine learning for forecasting multivariate time series: An empirical approach. *Commun. Stat. Case Stud. Data Anal. Appl.* **2025**, *11*, 56–91. [CrossRef]
30. Ma, T.; Antoniou, C.; Toledo, T. Hybrid machine learning algorithm and statistical time series model for network-wide traffic forecast. *Transp. Res. Part Emerg. Technol.* **2020**, *111*, 352–372. [CrossRef]
31. Kaur, G.; Saini, S. Comparative analysis of RMSE and MAP metrics for evaluating CNN and LSTM models. *AIP Conf. Proc.* **2024**, *3121*, 040003. [CrossRef]
32. Pan, Y.A.; Guo, J.; Chen, Y.; Cheng, Q.; Li, W.; Liu, Y. A fundamental diagram based hybrid framework for traffic flow estimation and prediction by combining a Markovian model with deep learning. *Expert Syst. Appl.* **2024**, *238*, 122219. [CrossRef]
33. Chango, W.; Buñay, P.; Erazo, J.; Aguilar, P.; Sayago, J.; Flores, A.; Silva, G. Predicting Urban Traffic Congestion with VANET Data. *Computation* **2025**, *13*, 92. [CrossRef]
34. Secretaría Nacional de Planificación. *Plan de Desarrollo para el Nuevo Ecuador 2024–2025*; Secretaría Nacional de Planificación: Quito, Ecuador, 2024; pp. 1–300, ISBN 978-9942-22-592-4. Available online: https://www.planificacion.gob.ec/plan-de-desarrollo-para-el-nuevo-ecuador-2024-2025/ (accessed on 8 December 2025).
35. Parums, D.V. Editorial: Review Articles, Systematic Reviews, Meta-Analysis, and the Updated Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) 2020 Guidelines. *Med. Sci. Monit. Int. Med. J. Exp. Clin. Res.* **2021**, *27*, e934475. [CrossRef]

https://doi.org/10.3390/computation14010005

36. Basnayake, B.R.P.M.; Chandrasekara, N.V. Forecasting exchange rates in Sri Lanka: A comparison of the double seasonal autoregressive integrated moving average models (DSARIMA) and SARIMA models. *J. Sci. Univ. Kelaniya* **2022**, *15*, 192–209. [CrossRef]
37. Wei, X. Prediction and influence factors analysis of IP backbone network traffic based on Prophet model and variance reduction. *Heliyon* **2025**, *11*, e41472. [CrossRef] [PubMed]
38. Yang, J.; Li, P.; Cui, Y.; Han, X.; Zhou, M. Multi-Sensor Temporal Fusion Transformer for Stock Performance Prediction: An Adaptive Sharpe Ratio Approach. *Sensors* **2025**, *25*, 976. [CrossRef]
39. Mendoza, A.P. Dengue incidence forecasting model in Magalang, Pampanga using time series analysis. *Inform. Med. Unlocked* **2024**, *44*, 101439. [CrossRef]
40. Cre¸tan, R.C.; Abula, K.; Aihemaiti, Y. Digitalization and Culture–Tourism Integration in China: The Moderated Mediation Effects of Employment Quality, Infrastructure, and New-Quality Productivity. *Sustainability* **2025**, *17*, 8792. [CrossRef]
41. Adebayo, P.; Ahmed, I.; Oyeleke, K.T. Comparative Analysis of Prophet and ARIMA Models for Forecasting Economic Indicators: A Case Study of Personal Consumption Expenditures. *SSRN* **2025**. [CrossRef]
42. Liu, J.; Yu, F.; Song, H. Application of SARIMA model in forecasting and analyzing inpatient cases of acute mountain sickness. *BMC Public Health* **2023**, *23*, 56. [CrossRef]
43. Letessier, C.; Cardi, J.; Dussel, A.; Ebtehaj, I.; Bonakdari, H. Enhancing Flood Prediction Accuracy through Integration of Meteorological Parameters in River Flow Observations: A Case Study Ottawa River. *Hydrology* **2023**, *10*, 164. [CrossRef]
44. Keil, A.; Bernat, E.M.; Cohen, M.X.; Ding, M.; Fabiani, M.; Gratton, G.; Kappenman, E.S.; Maris, E.; Mathewson, K.E.; Ward, R.T.; et al. Recommendations and publication guidelines for studies using frequency domain and time-frequency domain analyses of neural time series. *Psychophysiology* **2022**, *59*, e14052. [CrossRef] [PubMed]
45. Mienye, I.D.; Swart, T.G.; Obaido, G. Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. *Information* **2024**, *15*, 517. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/computation14010005

symmetry

MDPI

Article

Dynamic Interval Prediction of Subway Passenger Flow Using a Symmetry-Enhanced Hybrid FIG-ICPO-XGBoost Model

Qingling He [1], Yifan Feng [1], Lin Ma [2,*], Xiaojuan Lu [1,*], Jiamei Zhang [1] and Changxi Ma [1]

[1] School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China; qinglinghe@lzjtu.edu.cn (Q.H.); 11250042@stu.lzjtu.edu.cn (Y.F.); 12251045@stu.lzjtu.edu.cn (J.Z.); machangxi@mail.lzjtu.cn (C.M.)

[2] Linxia Daohe Investment Co., Ltd., Lanzhou 730070, China

* Correspondence: malin_daohe@163.com (L.M.); luxj2025@lzjtu.edu.cn (X.L.)

Abstract

To address the challenges of characterizing subway passenger flow fluctuations and overcoming the slow convergence and significant errors of existing intelligent optimization algorithms in tuning deep learning parameters for flow prediction, this study proposes a novel subway passenger flow fluctuation interval prediction model based on a Symmetry-Enhanced FIG-ICPO-XGBoost model. The core innovation is an Improved Cheetah Optimization Algorithm (ICPO), which incorporates enhancements including Circle mapping for population initialization, a hybrid strategy of dimension-by-dimension pinhole imaging opposition-based learning and Cauchy mutation to escape local optima, and adaptive variable spiral search with inertia weight to balance exploration and exploitation. The construction of this methodology embodies the concept of symmetry in algorithm design. For instance, Circle mapping achieves uniformity and ergodicity in the initial distribution of the population within the solution space, reflecting the symmetric principle of spatial coverage. Dimension-by-dimension pinhole imaging opposition-based learning generates opposite solutions through the principle of mirror symmetry, effectively expanding the search space. The adaptive variable spiral search strategy dynamically adjusts the spiral shape, simulating the symmetric relationship of dynamic balance between exploration and exploitation. Utilizing fuzzy-granulated passenger flow data (LOW, R, UP) from Harbin, the ICPO was employed to optimize XGBoost hyperparameters. Experimental results demonstrate the superior performance of the ICPO on 12 benchmark functions. The ICPO-XGBoost model achieves mean MAE, RMSE, and MAPE values of 10,291, 10,612, and 5.8%, respectively, for the predictions of the LOW, R, and UP datasets. Compared to existing models such as CPO-XGBoost, PSO-BiLSTM, GA-BP, and CNN-LSTM, these values represent improvements ranging from 4541 to 13,161 for MAE, 5258 to 14,613 for RMSE, and 2.6% to 7.2% for MAPE. The proposed model provides a reliable theoretical and data-driven foundation for optimizing subway train schedules and station passenger flow management.

Keywords: transportation engineering; subway passenger flow; crested porcupine optimizer; XGBoost; fluctuation interval prediction

1. Introduction

Subway passenger flow serves as the fundamental basis for subway operation management departments to formulate network planning and operational management policies.

The key to reasonably anticipating subway passenger flow fluctuation trends and range intervals, devising vehicle operation scheduling plans, and formulating passenger entry/exit station organization strategies lies in constructing accurate prediction models that account for the stochastic, time-varying, and imbalanced characteristics of subway passenger flow. This, in turn, helps passengers plan their travel routes and times more effectively, enhances subway operational efficiency, and safeguards service levels and quality.

Existing subway passenger flow prediction models can be categorized into mathematical statistical models, machine learning models, and deep learning models. Mathematical statistical models include multiple linear regression [1], ARIMA [2], and SARIMA [3]. These models are limited to fitting the linear relationships between subway passenger flow and influencing factors such as the built environment [4], weather conditions, and air quality [5], based on sufficient historical data samples. They neglect the spatiotemporal dynamic variation characteristics of subway passenger flow. Consequently, during the model parameter calibration process, they often inadequately consider the dynamic and nonlinear relationships among influencing factors, which reduces the prediction result accuracy and generalization capability of these models.

With the advancement of information technology and significant improvements in computational data processing capabilities, machine learning models such as Least Squares Support Vector Machines (LSSVM) [6], Random Forests [7], and models based on Fuzzy C-Means clustering (FCM) [8] have been widely applied to subway passenger flow prediction research. Machine learning models can effectively uncover the dynamic complexity and nonlinear spatiotemporal distribution patterns of metro passenger flow, enabling a symmetrical description of the complex relationships between data inputs and outputs. This, in turn, enhances the prediction accuracy and robustness of the models. Models such as the seasonal and nonlinear least squares support vector machine (LSSVM) [9] and the integrated metro passenger flow forecasting model combining time series analysis and support vector machines (SVM) [10] can effectively capture the cyclical symmetry and nonlinear characteristics of metro passenger flow, demonstrating strong predictive accuracy and robustness.

However, machine learning models can only extract nonlinear correlation features between subway passenger flow and influencing factors [11]. They struggle to mine the spatiotemporal correlation characteristics and evolutionary patterns among factors such as built environment POI points [12], subway station and passenger types [13], station connectivity, and traffic operation characteristics under large data sample conditions [14]. Deep learning models such as Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Gated Recurrent Units (GRU) can effectively capture the spatiotemporal dynamic complexity and nonlinear periodic symmetric relationships between metro passenger flow and influencing factors, further improving the accuracy and robustness of metro passenger flow predictions [15].

Combined models based on Attention Mechanism with BiLSTM [16], CNN-LSTM [17], Hybrid Gradient Boosting Decision Trees [18], and Conv Transformer [19] offer advantages in higher prediction accuracy and robustness compared to models like LSSVM, CNN, and GRU [20]. To address the issue of deep learning models relying on empirical manual hyperparameter setting, which can lead to insufficient prediction accuracy and convergence speed, models such as Genetic Algorithm optimized Wavelet Neural Networks (GA-WNN) [21], GWO-SVM [22], and WOA-BiGRU [23] have been proposed for subway passenger flow prediction. These models employ intelligent optimization algorithms to optimize machine learning hyperparameters, thereby enhancing model prediction accuracy and generalization capability. Further, models like IPSO-LSSVM [24] and ISSA-CNN-LSTM [25] improve the intelligent optimization algorithms themselves using

https://doi.org/10.3390/sym18020288

hybrid strategies. This effectively overcomes problems like the algorithms being prone to local optima and having insufficient solution precision and speed, further boosting the prediction accuracy and applicability of the subway passenger flow prediction models.

Among intelligent optimization algorithms, the Crested Porcupine Optimizer (CPO) features a simple structural principle and few model parameters. Compared to other existing intelligent optimization algorithms, it demonstrates advantages in higher optimization precision and faster convergence speed. It has been widely applied to engineering problems such as path planning and machine learning parameter optimization [26]. However, the No Free Lunch (NFL) theorem indicates that no single intelligent optimization algorithm can handle all complex engineering optimization problems. Existing scholars argue that CPO's cyclic preferential population initialization strategy can easily reduce population diversity and quality. Furthermore, the population individual position update methods in the CPO's perception and physiological defense phases have certain limitations, which diminish the algorithm's ability to balance global exploration and local exploitation. Aimed at CPO's drawbacks—low population diversity, imbalanced exploration-exploitation, and resultant local optima trapping with slow convergence—an improved algorithm is proposed. Existing research on improving the Crested Porcupine Optimizer (CPO) primarily focuses on two aspects: Firstly, in the population initialization phase, strategies such as the sinusoidal chaotic map [27], Latin hypercube sampling [28], and good point set [29] have been introduced to replace the original cyclic generation mechanism. These strategies significantly enhance the distribution uniformity and diversity of the initial population in the solution space, laying a high-quality foundation of solution sets for subsequent optimization. Secondly, in terms of the population individual position update mechanism, strategies such as elite opposition-based learning [30], Lévy flight strategy [31], the velocity vector mechanism of particle swarm optimization [32], Cauchy mutation operator [33], and adaptive neighborhood search [34] have been systematically integrated to improve the position update methods during the perception and physiological defense phases of the Crested Porcupine Optimizer. These strategies effectively expand the algorithm's global exploration capabilities while enhancing its local exploitation abilities, thereby optimizing the balance between global search and local convergence when solving complex problems.

Existing research has mainly focused on investigating the impact of factors like weather conditions and the built environment on subway passenger flow, constructing prediction models aimed at improving the accuracy of point-in-time prediction results. The metro passenger flow point prediction model overlooks the characteristics of passenger flow randomness, time-varying nature, and imbalance, while also insufficiently considering the cyclical symmetry of passenger flow. These limitations reduce the applicability of the passenger flow prediction results. Furthermore, existing intelligent optimization algorithms exhibit problems such as being prone to local optima and insufficient convergence when optimizing deep learning hyperparameters. These issues constrain model prediction accuracy and generalization capability, reducing the applicability of subway passenger flow prediction results. Therefore, addressing the deficiencies in CPO's population initialization method and the position update methods for individuals in its perception and physiological phases—which can lead to suboptimal optimization precision and convergence speed—this paper employs the Circle map for population initialization to enhance population diversity and quality. It utilizes a hybrid mechanism combining Dimension-by-Dimension Pinhole Imaging Opposition-Based Learning and Cauchy mutation to expand the algorithm's optimal feasible solution search space and improve its ability to escape local optima. Additionally, it incorporates adaptive variable spiral search and inertia weight strategies to modify the individual position update methods in the CPO's perception and defense phases.

https://doi.org/10.3390/sym18020288

These enhancements aim to improve the algorithm's ability to balance global exploration and local exploitation, thereby increasing its optimization precision and convergence speed.

Based on the subway passenger flow data of Harbin City, this paper analyzes the degree of correlation between subway passenger flow and weather conditions. Considering the fuzzy information granulation characteristics of the subway passenger flow data, a hybrid strategy-improved CPO is used to optimize XGBoost parameters, constructing a FIG-ICPO-XGBoost model for subway passenger flow fluctuation interval prediction. By accounting for the stochasticity and spatiotemporal dynamic distribution characteristics of subway passenger flow, this model expands the prediction results from a point estimate to an interval estimate (from a point to a range). The use of hybrid strategies to improve the CPO overcomes issues where the algorithm is prone to local optima and slow convergence, which would otherwise result in lower XGBoost prediction accuracy and longer runtimes, thereby enhancing the applicability of the subway passenger flow prediction results.

The remainder of this paper is organized as follows: Section 2 analyzes the spatiotemporal fluctuation characteristics of subway passenger flow and explores the correlation between these fluctuation characteristics and influencing factors. Section 3 details the strategies and their roles in improving the CPO algorithm. It presents a comparative analysis of numerical simulation results from optimizing 12 benchmark test functions using the hybrid strategy-improved CPO and introduces the FIG-ICPO-XGBoost subway passenger flow fluctuation interval prediction model. Section 4 calculates and analyzes the characteristics of the subway passenger flow fuzzy information granulation results, discusses the results of the FIG-ICPO-XGBoost model, and conducts a comparative analysis of the prediction accuracy and applicability between the proposed model and existing models.

2. Analysis of Subway Passenger Flow Data Sources and Influencing Factors

2.1. Statistical Analysis of Subway Passenger Flow Data Sources

This study utilizes fare card entry data from Harbin Metro Line 1, covering the period from 6 January 2020 to 27 December 2020. Corresponding weather condition data for the respective dates were also obtained through queries and calculations. To intuitively analyze the passenger flow characteristics of metro stations, passenger flow data from 12 stations on Line 1 were selected. These include terminal stations and transfer hubs. A spatiotemporal fluctuation characteristic diagram of metro passenger flow was plotted, as shown in Figure 1. As can be seen from Figure 1, although passenger flows at different metro stations exhibit distinct variation patterns and fluctuation ranges over the same period, their overall fluctuation patterns and changing trends are fundamentally consistent.

2.2. Analysis of Influencing Factors on Subway Passenger Flow

Liu et al. [25] suggest that variations in weather conditions significantly impact the accuracy of subway passenger flow prediction results. To uncover the underlying relationships between subway passenger flow and weather conditions and to identify influential factors for constructing a fluctuation interval prediction dataset, this study aims to enhance the accuracy and applicability of passenger flow predictions. Using fare card entry data and weather condition data from Harbin spanning 357 days (from 6 January to 27 December 2020), the Pearson correlation coefficient was employed to quantify the degree of association between subway passenger flow and various weather-related influencing factors.

https://doi.org/10.3390/sym18020288

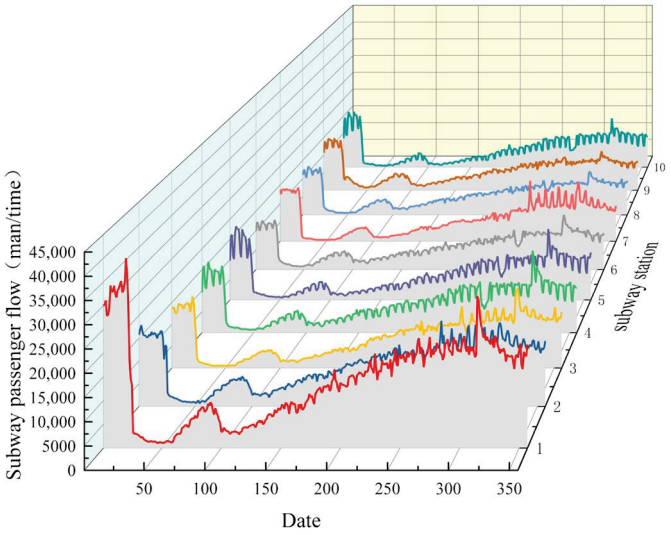


Figure 1. Spatiotemporal fluctuation characteristics of passenger flow at subway stations.

Existing research suggests that when the correlation coefficient between influencing factors exceeds 0.85, it indicates a high degree of association between the two factors, and one of them should be removed [25]. As shown in Figure 2, the correlation coefficients among the selected subway passenger flow influencing factors in this paper are all below 0.84. Therefore, this study selects temperature, air quality index, relative humidity, and wind scale as the influencing factors for predicting subway passenger flow fluctuation intervals.

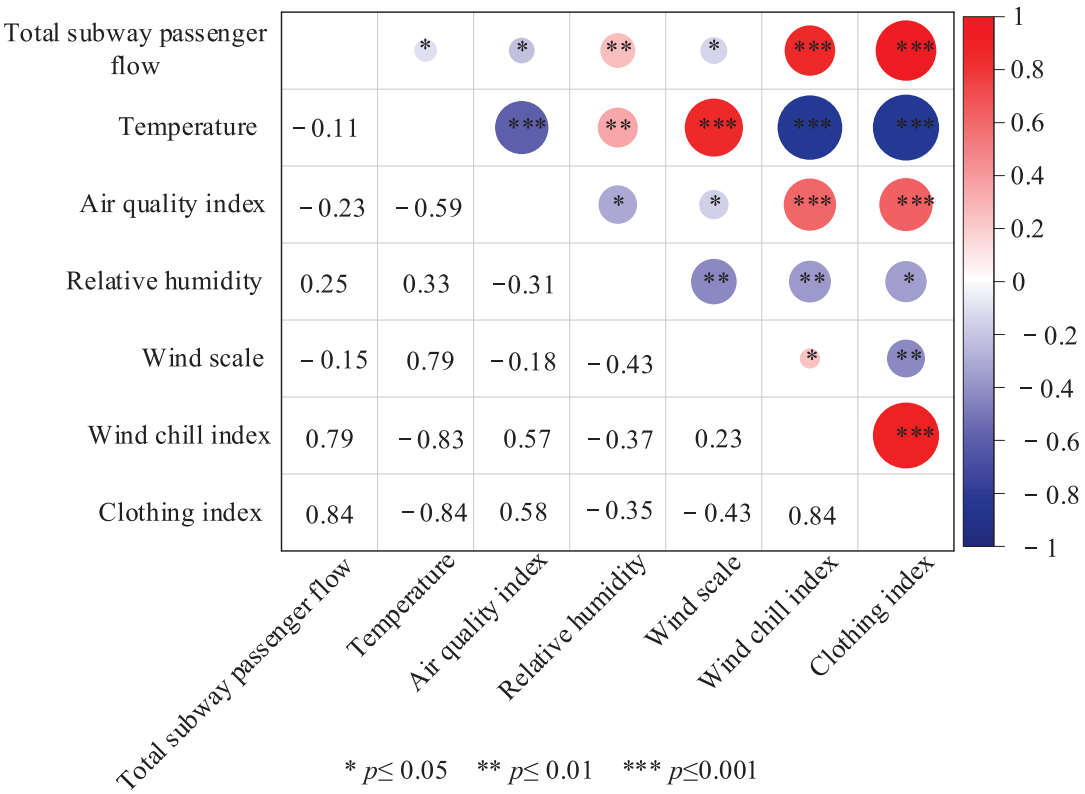


Figure 2. Correlation matrix of influencing factors for subway passenger flow.

https://doi.org/10.3390/sym18020288

3. FIG-ICPO-XGBoost-Based Subway Passenger Flow Fluctuation Interval Prediction Model

3.1. Design of Hybrid Strategy Improved Crested Porcupine Optimizer

Aiming at the defects in the random population initialization method and the position update strategies for individuals during the perceptual and physiological stages of the Crested Porcupine Optimizer (CPO), which can easily lead to low solution accuracy and insufficient convergence speed, this study designs a hybrid improvement strategy. First, the Circle mapping is adopted to initialize the population, enhancing population diversity and quality. Second, a hybrid mechanism combining dimension-by-dimension pinhole imaging reverse learning and Cauchy mutation is introduced to expand the search space for optimal feasible solutions and improve the algorithm's ability to escape local optima. Finally, an adaptive variable spiral search strategy and an inertia weight strategy are designed to optimize the position update methods for individuals in the perception and defense stages of the CPO population, aiming to enhance the algorithm's ability to balance global exploration and local exploitation, thereby improving solution accuracy and convergence speed.

3.1.1. Optimization Design of Population Initialization Based on Circle Mapping

The Crested Porcupine Optimizer (CPO) is a novel intelligent optimization algorithm proposed by Abdel-Basset et al. in 2024 [35]. It simulates the probabilistic symmetrical defense behavior mechanism of the crested porcupine. In this algorithm, each crested porcupine is considered a population individual. A cyclic preferential strategy is adopted to enhance population diversity and quality. Compared to the random initialization methods of other intelligent optimization algorithms, this approach offers advantages in terms of ease of escaping local optima and fast convergence speed. The population initialization and cyclic preferential process are defined by Equations (1) and (2):

$$\vec{X_i} = \vec{L} + \vec{r} \times \left(\vec{U} - \vec{L}\right) \Big| i = 1, 2, \ldots, N \tag{1}$$

where $\vec{X_i}$ represents the i-th individual in the search space population; $\vec{L}$ is the upper bound of the number of individuals in the search space population; $\vec{U}$ is the lower bound of the number of individuals in the search space population; $\vec{r}$ is a random vector within [0, 1]; N is the total number of population individuals.

$$N = N_{\min} + \left(N' - N_{\min}\right) \times \left(1 - \left(\frac{t \% \frac{T_{\max}}{T}}{\frac{T_{\max}}{T}}\right)\right) \tag{2}$$

where T represents the number of cyclic preferential selections; t is the current count of cyclic preferential selections; $T_{\max}$ is the maximum number of cyclic preferential selections; % is the modulo operator; $N_{\min}$ is the minimum size of the newly generated population from cyclic preferential selections.

The perception-defense strategy refers to the process in which crested porcupines establish an approximately symmetrical deterrent field in the surrounding space by fluffing their feathers and creating noise to evade predators. The mathematical model of the perception-defense strategy process is divided into Equations (3) and (4) as follows:

$$\vec{X_i^{t+1}} = \vec{X_i^t} + \tau_1 \times \left|2 \times \tau_2 \times \vec{x_{cp}^t} - \vec{y_i^t}\right| \tag{3}$$

https://doi.org/10.3390/sym18020288

where $\vec{x}_{cp}^{t}$ represents the optimal individual in the current cyclic preferential selection; $\vec{y}_i^t$ is a vector generated between the current optimal individual and other random individuals, used to characterize the position of the predator at iteration t; τ_1 is a random number following a normal distribution; τ_2 is a random number within [0, 1].

$$\vec{x}_i^{t+1} = \left(1 - \vec{U}_1\right) \times \vec{x}_i^t + \vec{U}_1 \times \left(\vec{y} + \tau_3\left(\vec{x}_{r1}^t - \vec{x}_{r2}^t\right)\right) \tag{4}$$

where r_1 and r_2 are random integers within the range [1, N]; τ_3 is a randomly generated value within the interval [0, 1].

The physiological defense strategy refers to the behavioral process in which crested porcupines establish an approximately symmetrical deterrent field around themselves by secreting odors and executing physical attacks to evade predators. The mathematical models for this physiological defense strategy are given by Equation (5) and Equation (6), respectively:

$$\vec{x}_i^{t+1} = \left(1 - \vec{U}_1\right) \times \vec{x}_i^t + \vec{U}_1 \times \left(\vec{x}_{r1}^t + S_i^t \times \left(\vec{x}_{r2}^t - \vec{x}_{r3}^t\right) - \tau_3 \times \vec{\delta} \times \gamma_t \times S_i^t\right) \tag{5}$$

where $\vec{x}_i^{t+1}$ represents the position of the i-th population individual at the t-th cyclic preferential iteration; $\vec{\delta}$ is a parameter controlling the search direction; γ_t is the defense factor; S_i^t is the odor diffusion factor; r_3 is a random number within the range [1, N].

$$\vec{x}_i^{t+1} = \vec{x}_{cp}^t + (\alpha(1 - \tau_4) + \tau_4) \times \left(\delta \times \vec{x}_{cp}^t - \vec{x}_i^t\right) - \tau_5 \times \delta \times \gamma_t \times \vec{F}_i^t \tag{6}$$

where α is the convergence speed factor, τ_4 is a randomly generated value within the interval [0, 1]; $\vec{F}_i^t$ is the average force affecting the i-th population individual.

To address the issue where the random population initialization method in the CPO algorithm tends to cause the algorithm to prematurely converge to local optima, this study employs Circle mapping to optimize the structural symmetry of the initial search space in the CPO algorithm. This approach effectively enhances the uniformity and traversal coverage of population individuals within the optimization search space. By linearly transforming feasible solutions in the search area into a globally symmetrical space optimized via chaotic mapping variables, the Circle mapping significantly reduces the risk of the algorithm becoming trapped in local optima, thereby improving its overall optimization efficiency [36].

$$x_{i+1} = \mathrm{mod}(x_i + 0.2 - (0.5/2\pi)\sin(2\pi x_i), 1) \tag{7}$$

3.1.2. Hybrid Mechanism of Dimension-by-Dimension Pinhole Imaging Opposition-Based Learning and Cauchy Mutation

(1) Dimension-by-Dimension Pinhole Imaging Opposition-Based Learning

The strategy of dimension-wise pinhole imaging reverse learning combines pinhole imaging with dimension-wise reverse learning. By leveraging the translational and reflective symmetry assumptions in the feasible region's spatial structure, it enables dimension-wise reverse solving of feasible solutions, thereby reducing the risk of the algorithm falling into local optima. This approach not only effectively mitigates the mutual interference between dimensions of feasible solutions when the CPO algorithm solves high-dimensional functions, increasing the diversity of the optimization search, but also expands global exploration capabilities and enhances the algorithm's ability to escape local optima. Its principle is illustrated in Figure 3.

https://doi.org/10.3390/sym18020288

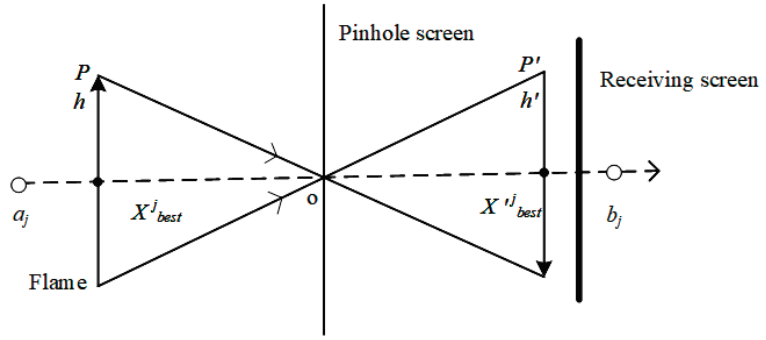


Figure 3. Principle of dimension-by-dimension pinhole imaging opposition-based learning.

Dimension-by-dimension pinhole imaging opposition-based learning assumes that there is a flame p with height h in space, whose projection on the X-axis is X^j_{best}. The upper and lower bounds of the coordinate axis are a_j and b_j, respectively. A pinhole screen is placed at the coordinate origin o. The flame can pass through the pinhole on the screen to form an inverted image p' with height h' on the receiving screen, and through the pinhole, it generates a reverse point X'^j_{best} on the X-axis. The principle of dimension-by-dimension pinhole imaging opposition-based learning can be described as follows:

$$\frac{(a_j + b_j)/2 - X^j_{best}}{X'^j_{best} - (a_j + b_j)/2} = \frac{h}{h'} \tag{8}$$

Given $\frac{h}{h'} = n$, through transformation, can be obtained as:

$$X'^j_{best} = \frac{(a_j + b_j)}{2} + \frac{(a_j + b_j)}{2n} - \frac{X^j_{best}}{n} \tag{9}$$

When $n = 1$, Equation (9) can be transformed into:

$$X'^j_{best} = (a_j + b_j) - X^j_{best} \tag{10}$$

(2) Cauchy Mutation

The Cauchy distribution, as a continuous probability distribution without a defined mathematical expectation, exhibits a probability density function that symmetrically and gradually declines from its peak toward both ends at an identical rate, asymptotically approaching but never intersecting the axis. This characteristic enables the distribution to perturb the current solution symmetrically with equal probability in both positive and negative directions within a short timeframe, allowing for rapid exploration of feasible solutions in the vicinity. Using Cauchy mutation to enhance the global search capability of the CPO algorithm not only optimizes the optimal positions of population individuals but also reduces the risk of the algorithm prematurely converging to local optima. The formula for the standard Cauchy distribution function is:

$$f(x) = \frac{1}{\pi(1 + x^2)} \tag{11}$$

After the CPO algorithm obtains the current optimal feasible solution, the Cauchy mutation operator is applied to perturb the position of the best population individual, yielding its updated position as:

$$X^j_{newbest} = X^j_{best} + X^j_{best} \times Cauchy(0, 1) \tag{12}$$

https://doi.org/10.3390/sym18020288

where $Cauchy(0,1)$ represents a random value drawn from a Cauchy distribution with a mean of 0 and a standard deviation of 1.

(3) Hybrid Mechanism

The hybrid mechanism proposed in the paper refers to the use of dimension-wise reverse learning based on pinhole imaging to maintain and broaden the symmetrical coverage of the potential optimal solution space during the search process, thereby enhancing the algorithm's optimization performance. Additionally, Cauchy mutation is applied to introduce symmetrical random perturbations to the positions of the best population individuals in both positive and negative directions. This reduces the risk of the algorithm prematurely converging to local optima and improves its convergence speed. The hybrid mechanism alternates between dimension-wise reverse learning and Cauchy mutation with a symmetrical probability of 50%, enabling the dynamic and symmetrical updating of the best population individual positions. The mathematical model of the hybrid mechanism is as follows:

$$X^j_{newbest,1} = \begin{cases} \frac{(a_j+b_j)}{2} + \frac{(a_j+b_j)}{2n} - \frac{X^j_{best}}{n}, \lambda < 0.5 \\ X^j_{best} + X^j_{best} \times Cauchy(0,1), \lambda \geq 0.5 \end{cases} \tag{13}$$

where λ is a random number on [0, 1].

3.1.3. Adaptive Variable Spiral Search and Inertia Weight

To address the limitations of the CPO algorithm in global exploration capability, this paper introduces an adaptive variable spiral search strategy with rotational and radial symmetry to enhance the population individual position update process [37]. The adaptive variable spiral search strategy initiates the search from the optimal individual position in a symmetric state and dynamically adjusts through rotational symmetry, thereby improving the global exploration capability of the CPO algorithm. The adaptive variable spiral search strategy is defined by Equations (14) and (15):

$$\vec{X}(t+1) = \vec{D'} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X^*}(t) \tag{14}$$

$$\vec{D'} = \left| 2r\vec{X^*}(t) - \vec{X}(t) \right| \tag{15}$$

where $\vec{X}(t+1)$ represents the position of the whale individual at iteration $t+1$; $\vec{X^*}$ is the global optimal individual position; b is a constant defining the shape of the spiral position; l is a random number within $[-1, 1]$; r is a random variable within [0, 1].

Applying the adaptive variable spiral search strategy to the position update process of the crested porcupine population individuals, while introducing an inertia weight, aims to gradually enhance the influence of the best crested porcupine population individuals in the later iterations, enabling rapid convergence to the optimal position and improving the algorithm's convergence speed.

$$\omega = 0.2\cos(0.5\pi(1 - \frac{t}{T})) \tag{16}$$

The inertia weight ω dynamically changes as the number of iterations increases. In the early stages of the CPO algorithm's search, the best population individual has a relatively small influence on other individuals, allowing them to engage in thorough global exploration. During the middle and later iterations, the influence of the best population individual gradually strengthens, encouraging other individuals to move closer to it and thereby accelerating convergence. During each spiral search, whales approach their prey along fixed spiral symmetric paths, which can easily lead to them falling into local optima.

https://doi.org/10.3390/sym18020288

Therefore, this paper modifies the spiral constant b into a parameter that varies with the number of iterations, dynamically adjusting the spiral symmetric shape to expand the global search area of the population and enhance its global search capability. The formula for the variable spiral parameter η^* is:

$$\eta^* = e^{5\cos(\pi(1-\frac{t}{T}))} \tag{17}$$

From Equation (17), it can be seen that the number of iterations is inversely proportional to the spiral shape. During stages such as perceptual defense and physiological defense, this ensures that individuals within the crested porcupine population conduct global exploration within the maximum symmetric space range. In the later iterations, they gradually approach the optimal feasible solution, thereby ensuring improvements in both the algorithm's global search capability and optimization accuracy. The corresponding position update methods for each stage are as follows:

$$\vec{X_i^{t+1}} = \omega\vec{X_i^t} + \eta^* \cdot D \cdot e^{bl}\cos(2\pi p) + \tau_1 \times \left|2 \times \tau_2 \times \vec{x_{cp}^t} - \vec{y_i^t}\right| \tag{18}$$

$$\vec{x_i^{t+1}} = \left(1 - \vec{U_1}\right) \times \omega\vec{x_i^t} + \left(1 - \vec{U_1}\right) \times \eta^* \cdot D \cdot e^{bl}\cos(2\pi p) + \vec{U_1} \times \left(\vec{y} + \tau_3\left(\vec{x_{r1}^t} - \vec{x_{r2}^t}\right)\right) \tag{19}$$

$$\vec{x_i^{t+1}} = \left(1 - \vec{U_1}\right) \times \omega\vec{x_i^t} + \left(1 - \vec{U_1}\right) \times \eta^* \cdot D \cdot e^{bl}\cos(2\pi p) + \vec{U_1} \times \left(\vec{x_{r1}^t} + S_i^t \times \left(\vec{x_{r2}^t} - \vec{x_{r3}^t}\right) - \tau_3 \times \vec{\delta} \times \gamma_t \times S_i^t\right) \tag{20}$$

$$\vec{x_i^{t+1}} = \omega\vec{x_{cp}^t} + \eta^* \cdot D \cdot e^{bl}\cos(2\pi p) + (\alpha(1-\tau_4) + \tau_4) \times \left(\delta \times \vec{x_{cp}^t} - \vec{x_i^t}\right) - \tau_5 \times \delta \times \gamma_t \times \vec{F_i^t} \tag{21}$$

3.2. Validation of the Effectiveness of the Hybrid Strategy Improved Crested Porcupine Optimizer (ICPO)

3.2.1. Comparative Analysis of Numerical Simulation Experimental Results

To evaluate the optimization performance of the hybrid strategy improved CPO algorithm (ICPO), this paper selected 12 benchmark test functions, encompassing unimodal, multimodal, and composite modal types, for numerical simulation experiments. To ensure fairness in the comprehensive comparative analysis with algorithms such as CPO, SSA, WOA, and PSO, the population size for all algorithms was set to 50, and the maximum number of iterations was uniformly set to 1000. The numerical simulation results after 30 independent runs for each algorithm consist of the mean value and standard deviation. The statistical results of the numerical simulations for each benchmark test function are presented in Table 1.

Table 1. Simulation results of benchmark test functions.

Test Function	Metric	ICPO	CPO	SSA	WOA	PSO
f_1	Average	0	4.45×10^{-152}	5.66×10^{-70}	1.27×10^{-91}	8.41×10^{-59}
	Standard	0	2.43×10^{-151}	3.10×10^{-69}	6.38×10^{-91}	1.95×10^{-58}
f_2	Average	0	1.11×10^{-103}	6.50×10^{-61}	1.11×10^{-103}	8.10×10^{-35}
	Standard	0	3.93×10^{-103}	3.41×10^{-60}	3.93×10^{-103}	5.53×10^{-35}
f_3	Average	0	1.98×10^{-83}	2.88×10^{-64}	1.98×10^{4}	3.14×10^{-15}
	Standard	0	1.00×10^{-82}	1.58×10^{-63}	1.07×10^{4}	1.44×10^{-14}
f_4	Average	2.19×10^{-5}	2.62×10^{1}	2.36×10^{1}	2.73×10^{1}	2.70×10^{1}
	Standard	5.35×10^{-5}	7.67×10^{-1}	4.83×10^{-1}	5.54×10^{-1}	6.99×10^{-1}

https://doi.org/10.3390/sym18020288

Table 1. *Cont.*

Test Function	Metric	ICPO	CPO	SSA	WOA	PSO
f_5	Average	-1.12×10^4	-1.21×10^4	-9.36×10^3	-5.79×10^3	-6.06×10^3
	Standard	1.53×10^3	2.79×10^2	2.60×10^3	4.32×10^2	8.87×10^2
f_6	Average	0	0	0	3.79×10^{-15}	1.54×10^{-1}
	Standard	0	0	0	1.44×10^{-14}	8.42×10^{-1}
f_7	Average	4.44×10^{-23}	5.63×10^{-16}	4.44×10^{-16}	3.40×10^{-15}	1.52×10^{-14}
	Standard	0	6.49×10^{-16}	0	2.65×10^{-15}	2.65×10^{-15}
f_8	Average	0	0	0	5.04×10^{-3}	2.62×10^{-3}
	Standard	0	0	0	2.10×10^{-2}	8.06×10^{-3}
f_9	Average	3.01×10^{-4}	3.07×10^{-4}	3.07×10^{-4}	6.87×10^{-4}	3.77×10^{-3}
	Standard	3.14×10^{-6}	1.92×10^{-15}	7.72×10^{-18}	3.22×10^{-4}	7.55×10^{-3}
f_{10}	Average	-3.32	-3.34	-3.26	-3.23	-3.26
	Standard	1.36×10^{-15}	2.17×10^{-2}	7.38×10^{-2}	1.11×10^{-1}	7.56×10^{-2}
f_{11}	Average	-10.153	-10.05	-10.03	-9.13	-9.23
	Standard	7.07×10^{-15}	4.10×10^{-12}	2.42×10^{-7}	2.07×10^0	2.14×10^0
f_{12}	Average	-10.53	-10.48	-10.45	-10.42	-10.12
	Standard	3.02×10^{-7}	5.19×10^{-12}	8.73×10^{-16}	3.02×10^{-7}	1.75×10^0

Unimodal benchmark functions (e.g., f_1, f_2, f_3 and f_4) contain only one global optimum and are widely adopted to assess the local exploitation capacity of intelligent optimization algorithms. According to the mean and standard deviation data in Table 1, the ICPO algorithm demonstrates better performance on f1–f4 compared to other intelligent algorithms, confirming its enhanced local exploitation capability.

Multimodal benchmark functions (e.g., f_5, f_6, f_7 and f_8) contain multiple local optima that grow with dimensionality, which effectively tests the global exploration ability of intelligent optimization algorithms. Results from f5–f8 indicate that ICPO generally excels over CPO, SSA, WOA, and PSO in global exploration. This improvement stems mainly from the hybrid mechanism integrating dimension-wise pinhole imaging opposition-based learning and Cauchy mutation, which boosts the probability of approaching the global optimum and alleviates CPO's susceptibility to local optima.

Composite benchmark functions (e.g., f_9, f_{10}, f_{12} and f_{13}) possess complex structures with many local minima and are commonly employed to evaluate how well an algorithm balances global search and local exploitation. Previous studies highlight that maintaining this balance is key to escaping local traps and reducing premature convergence. Numerical simulations on f9–f13 reveal that ICPO achieves a better balance between exploration and exploitation than CPO, SSA, WOA, and PSO. This is primarily due to the adaptive variable spiral search and inertia weight strategies, which refine individual position updates during perception and physiological defense phases, thus improving adaptive balancing throughout the search process.

To further examine the statistical behavior of ICPO, box plots based on 30 independent runs per algorithm are displayed in Figure 4. In each box, the central line denotes the median of solutions, the upper and lower edges correspond to the first quartile (Q1) and third quartile (Q3), and the "+" symbols mark outliers. The box plots indicate that ICPO produces fewer outliers and shows more concentrated solution distributions, reflecting better stability and robustness during optimization.

https://doi.org/10.3390/sym18020288

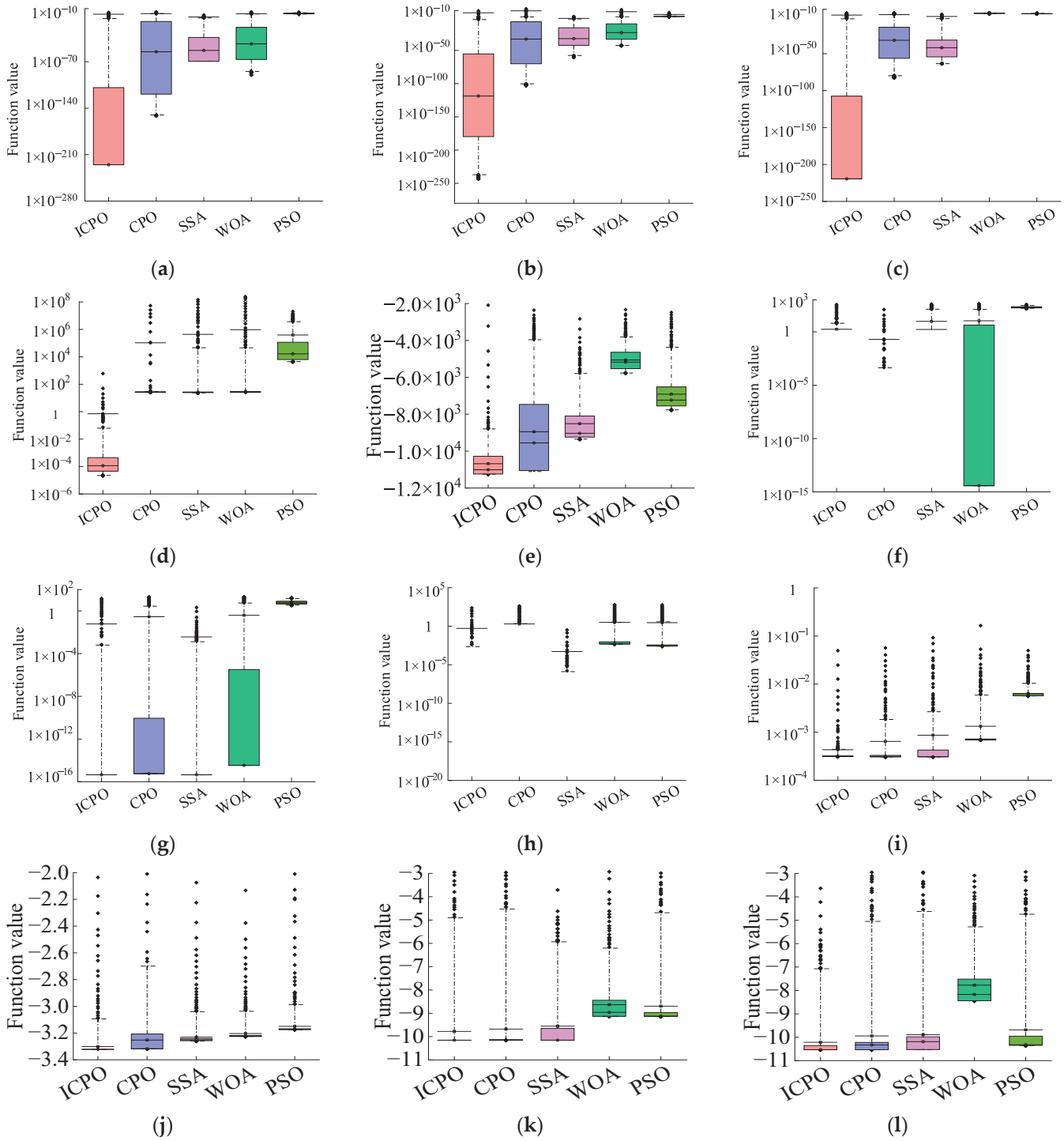


Figure 4. Statistical results of benchmark test function simulations: (**a**) f_1; (**b**) f_2; (**c**) f_3; (**d**) f_4; (**e**) f_5; (**f**) f_6; (**g**) f_7; (**h**) f_8; (**i**) f_9; (**j**) f_{10}; (**k**) f_{11}; (**l**) f_{12}.

To evaluate the robustness and fairness of the ICPO algorithm, this paper employs the Wilcoxon rank-sum test to statistically analyze the algorithm's performance at a significance level of 5%. Since ICPO cannot be compared with itself, statistical comparisons were conducted with intelligent optimization algorithms such as CPO, SSA, WOA, and PSO, with the results recorded as SP1, SP2, SP3, and SP4, respectively. The *p*-values of the Wilcoxon rank-sum test for each algorithm are shown in Table 2. From the results in Table 2, it can be observed that the *p*-values for the comparisons between the ICPO algorithm and each of the comparative algorithms across all 12 benchmark test functions are less than

https://doi.org/10.3390/sym18020288

1.38×10^{-3}, which is significantly lower than the 5% significance threshold. This statistical result indicates that the superior performance of the ICPO algorithm in terms of solution accuracy and convergence rate during the optimization of the benchmark test functions is highly statistically significant.

Table 2. Performance significance test of ICPO algorithm vs. comparison algorithms.

Test Function	P1	P2	P3	P4
f_1	1.35×10^{-162}	7.53×10^{-139}	1.34×10^{-162}	2.67×10^{-161}
f_2	3.33×10^{-165}	8.18×10^{-141}	3.32×10^{-165}	2.55×10^{-161}
f_3	3.26×10^{-165}	1.05×10^{-138}	3.33×10^{-165}	5.30×10^{-164}
f_4	3.33×10^{-165}	3.33×10^{-165}	3.32×10^{-165}	3.33×10^{-165}
f_5	7.45×10^{-165}	3.37×10^{-165}	3.34×10^{-165}	3.34×10^{-165}
f_6	1.38×10^{-3}	1.68×10^{-65}	4.48×10^{-172}	3.32×10^{-165}
f_7	1.24×10^{-175}	3.62×10^{-18}	8.22×10^{-173}	3.32×10^{-165}
f_8	1.35×10^{-42}	4.46×10^{-18}	3.38×10^{-167}	3.24×10^{-165}
f_9	2.80×10^{-3}	5.74×10^{-7}	3.33×10^{-165}	3.33×10^{-165}
f_{10}	1.09×10^{-156}	2.41×10^{-162}	6.48×10^{-165}	3.32×10^{-165}
f_{11}	6.84×10^{-100}	9.90×10^{-117}	2.88×10^{-132}	1.74×10^{-164}
f_{12}	1.46×10^{-122}	6.65×10^{-141}	3.28×10^{-165}	3.32×10^{-165}

3.2.2. Comparative Analysis of Algorithm Convergence

By analyzing the convergence curves of benchmark test functions, the convergence accuracy and speed of each intelligent optimization algorithm can be intuitively assessed, along with their ability to escape local optima and sustain continuous optimization. Figure 5 presents the convergence curves of the 12 benchmark test functions, where the horizontal axis represents the number of iterations, and the vertical axis indicates the average fitness value from 30 independent runs.

As shown in the convergence curves of functions f_1, f_2 and f_3, the ICPO algorithm achieves solution accuracy comparable to other comparative algorithms within approximately 200 iterations. Beyond this point, the hybrid mechanism combining dimension-by-dimension pinhole imaging opposition-based learning with Cauchy mutation, along with the inertia weight factor, introduces perturbations that enable the algorithm to effectively escape local regions and continue exploring for superior solutions.

For functions f_4, f_7, f_9 and f_{11}, the convergence curves reveal that during the early iterations, a larger inertia weight w drives the individuals of the crested porcupine population to rapidly converge toward the optimal individual. After about 100 iterations, an inflection point emerges in the ICPO curve. As iterations proceed, the increased likelihood of applying dimension-by-dimension pinhole imaging opposition-based learning leads to a sharp decline in fitness values.

In the case of f_5 and f_{12}, the ICPO algorithm exhibits multiple inflection points throughout the iteration process, unlike other algorithms. This behavior demonstrates that the enhancements in position updating during the perception and physiological defense phases—achieved through adaptive variable spiral search and inertia weight strategies—effectively balance global exploration and local exploitation, thereby improving both convergence speed and optimization accuracy.

Regarding f_6 and f_8, the ICPO algorithm consistently outperforms other comparative algorithms in fitness value from the early iterations onward. This indicates that the Circle map initialization yields a higher-quality population, which enhances the algorithm's global exploration capability to some extent. Moreover, ICPO reaches optimal accuracy within just 50 iterations, confirming its faster convergence rate.

https://doi.org/10.3390/sym18020288

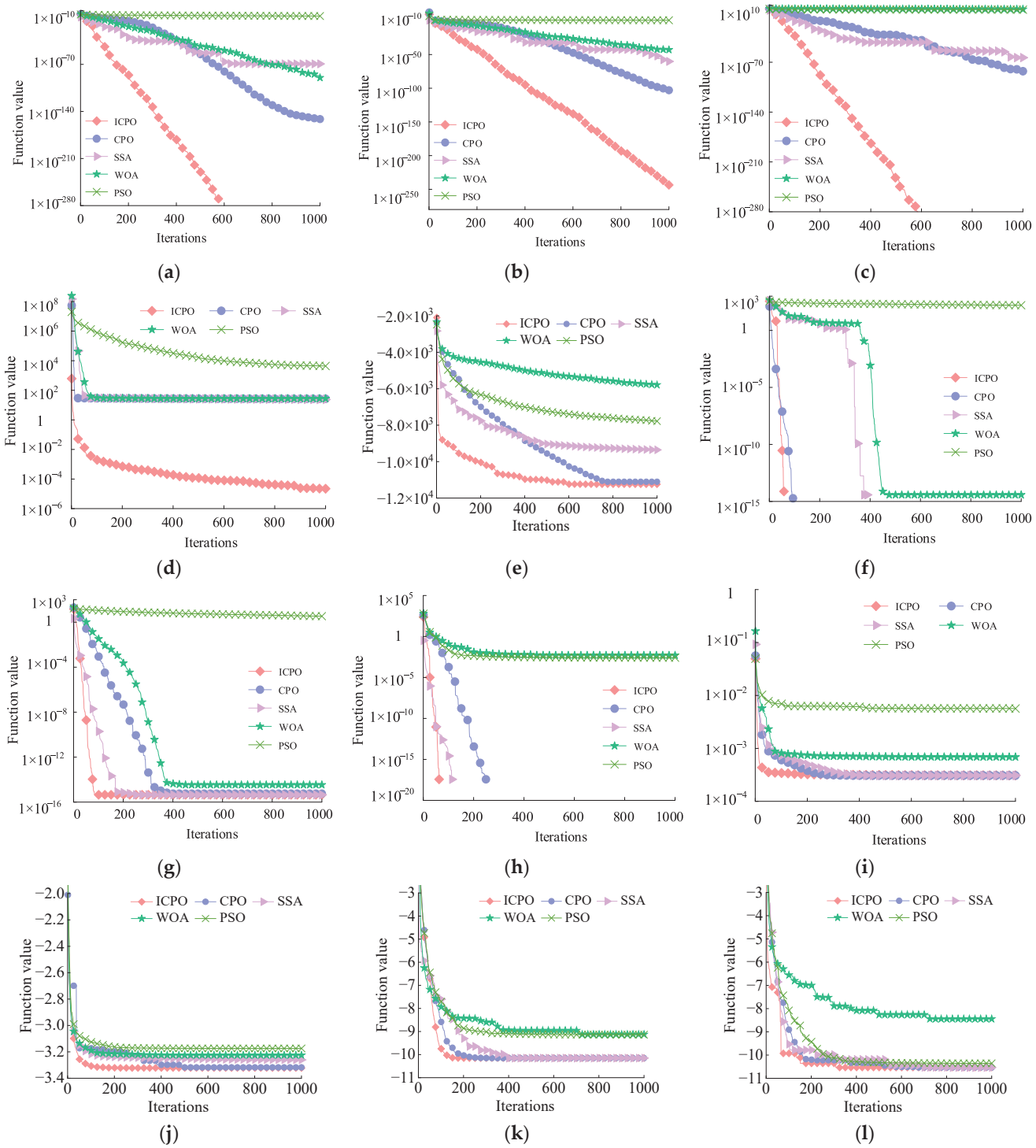


Figure 5. Convergence curves of benchmark test function simulations: **(a)** f_1; **(b)** f_2; **(c)** f_3; **(d)** f_4; **(e)** f_5; **(f)** f_6; **(g)** f_7; **(h)** f_8; **(i)** f_9; **(j)** f_{10}; **(k)** f_{11}; **(l)** f_{12}.

3.2.3. Comparative Analysis of Algorithm Time Complexity

Assuming the CPO algorithm has a population size of N and a space dimension of d, let the runtime for parameter initialization be t_1, the time for random population generation and distribution be t_2, and the time required for evaluating the fitness function during optimization be $f(d)$. Then, the time complexity of the initial phase is:

$$O(t_1 + N(dt_2 + f(d))) = O(d + f(d)) \tag{22}$$

https://doi.org/10.3390/sym18020288

Let the time for updating the position of each crested porcupine population individual across all dimensions be t_3, and the times for comparing superiority/inferiority and selecting the optimal position after each iteration be t_4 and t_5, respectively. Then, the time complexity of the CPO algorithm during the optimization phase can be expressed as:

$$O(N(dt_3 + f(d) + t_4) + t_5) = O(d + f(d)) \tag{23}$$

In summary, the time complexity for the CPO algorithm to obtain the optimal feasible solution per iteration is:

$$T(d) = O(d + f(d)) + O(d + f(d)) = O(d + f(d)) \tag{24}$$

In the ICPO algorithm, the time complexity of the initialization phase is essentially the same as that of the CPO algorithm. Assuming the time for updating the positions of crested porcupine population individuals during the optimization phase is e_1, the time for updating the best individual position using the hybrid mechanism of dimension-by-dimension pinhole imaging opposition-based learning and Cauchy mutation is e_2, the time for comparing and selecting the best individual is e_3, and the time for updating population individual positions using adaptive variable spiral search and inertia weight is e_4, then the time complexity of the ICPO algorithm at this stage is:

$$O(N(de_1 + f(d) + t_4) + t_5 + N(e_2 + e_4) + e_3) = O(d + f(d)) \tag{25}$$

In summary, the time complexity for the ICPO algorithm to obtain the optimal feasible solution per iteration is:

$$T(d) = O(d + f(d)) + O(d + f(d)) = O(d + f(d)) \tag{26}$$

In summary, the time complexity of the ICPO algorithm is fundamentally consistent with that of the CPO algorithm. To provide an intuitive comparative analysis of the time complexity between the two algorithms, this study separately executed the 12 benchmark test functions 50 times each using both ICPO and CPO algorithms on the same platform. The benchmark test functions were categorized by dimension, and the average time consumption for the corresponding 50 numerical simulation experiments was calculated. The average time consumption of the numerical simulation experiments is presented in Table 3.

Table 3. Comparison of average time consumption between ICPO and CPO.

Function Type	Average Time (s)		Ratio of ICPO to CPO (%)
	ICPO	CPO	
Unimodal Benchmark Functions	0.899	0.916	101.89
Multimodal Benchmark Functions	1.103	1.098	99.55
Composite Benchmark Functions	1.132	1.159	102.39
Average	1.045	1.058	101.18

From Table 3, it can be observed that the ICPO algorithm shows essentially the same time consumption as the original algorithm when solving multimodal benchmark test functions. Furthermore, when solving unimodal and composite benchmark test functions, it not only avoids increasing the algorithm's time complexity but also enhances convergence speed and optimization accuracy through hybrid improvement strategies.

Given that the quantitative analysis of all algorithms is based on the Mean Absolute Error (MAE) across the 12 benchmark test functions, ranking the numerical simulation test

https://doi.org/10.3390/sym18020288

results can effectively validate the algorithm's performance metrics. Table 4 presents the MAE ranking of the benchmark test functions, with its formula given by:

$$MAE = \frac{\sum_{i=1}^{N_f} |m_i - o_i|}{N_f} \tag{27}$$

where m_i is the average value of the algorithm's solution results, o_i is the theoretical value of each benchmark test function, N_f is the total number of benchmark test functions.

Table 4. MAE ranking of the algorithms.

Algorithm	MAE	Rank
ICPO	4.01×10^{1}	1
CPO	1.10×10^{2}	2
SSA	5.45×10^{2}	3
WOA	7.80×10^{2}	4
PSO	2.22×10^{3}	5

From Table 4, it can be observed that the ICPO algorithm achieves the smallest MAE value for the numerical simulation results of the 12 benchmark test functions, ranking first among the other intelligent optimization algorithms. This indicates that the ICPO algorithm holds a significant advantage in optimizing the theoretical values of the benchmark test functions.

To further analyze the effectiveness of the hybrid strategy on the ICPO algorithm, various strategies—including Circle mapping, the hybrid mechanism combining dimension-wise reverse learning via pinhole imaging with Cauchy mutation, adaptive variable spiral search, and inertia weight—were individually applied to improve the CPO algorithm. The CPO algorithms enhanced by these respective strategies are denoted as CPO1, CPO2, CPO3, and CPO4. The Wilcoxon rank-sum test was employed to compare and analyze the differences in optimization results between the ICPO algorithm and CPO1–CPO4 across 12 benchmark functions. The Wilcoxon test results for ICPO versus CPO1, CPO2, CPO3, and CPO4 are labeled as PC1, PC2, PC3, and PC4, respectively. The p-values of the Wilcoxon rank-sum test for the 12 benchmark functions are presented in Table 5. As shown in the table, all p-values for ICPO compared with each of the other algorithms across the 12 benchmark functions are less than 1.91×10^{-16}, indicating statistically significant differences between ICPO and the other algorithms.

Table 5. Significance test of hybrid-strategy ICPO algorithm.

Test Function	PC1	PC2	PC3	PC4
f_1	2.89×10^{-117}	2.68×10^{-147}	7.98×10^{-65}	2.48×10^{-87}
f_2	8.95×10^{-49}	9.68×10^{-127}	1.82×10^{-77}	3.57×10^{-69}
f_3	9.32×10^{-54}	6.95×10^{-134}	6.91×10^{-78}	6.56×10^{-59}
f_4	7.84×10^{-58}	9.47×10^{-119}	8.55×10^{-66}	7.57×10^{-79}
f_5	9.56×10^{-57}	4.59×10^{-104}	9.55×10^{-73}	9.54×10^{-77}
f_6	6.73×10^{-33}	3.78×10^{-155}	1.75×10^{-58}	4.49×10^{-53}
f_7	2.31×10^{-18}	8.83×10^{-156}	3.96×10^{-26}	3.55×10^{-37}
f_8	7.49×10^{-38}	1.39×10^{-115}	6.37×10^{-68}	5.69×10^{-61}
f_9	1.61×10^{-48}	9.54×10^{-149}	2.11×10^{-23}	1.91×10^{-16}
f_{10}	4.15×10^{-47}	1.19×10^{-148}	1.82×10^{-91}	8.44×10^{-93}
f_{11}	3.11×10^{-94}	2.89×10^{-124}	4.04×10^{-116}	1.28×10^{-98}
f_{12}	5.99×10^{-109}	9.72×10^{-149}	2.88×10^{-127}	9.38×10^{-101}

https://doi.org/10.3390/sym18020288

3.3. Construction of the FIG-ICPO-XGBoost Prediction Model

3.3.1. Fuzzy Information Granulation

Given the stochastic and uncertain nature of subway passenger flow and weather data, this paper employs fuzzy information granulation to partition the subway passenger flow data into fuzzy granulation windows and perform fuzzy granulation of the information. The purpose of fuzzy granulation window partitioning is to divide the data into several continuous sub-windows according to predefined time periods, based on the fluctuation characteristics and periodic patterns of the subway passenger flow data. This approach aims to capture local data features and their interrelationships within certain continuous time intervals. Information fuzzy granulation involves constructing appropriate fuzzy information granules to simplify data scale and spatial dimensionality while preserving the original window information and variation trends. The membership function for fuzzy information granulation is determined by constructing a histogram distribution of the data. The histogram distribution of metro passenger flow data is shown in Figure 6. It can be observed from Figure 6 that a symmetric triangular membership function effectively facilitates the fuzzy information granulation of metro passenger flow data. By employing a symmetric fuzzy granulation method to process metro passenger flow data, the global symmetric characteristics inherent in the data can be analyzed, enabling localized adaptive modeling within the granulation window period. This approach enhances the accuracy and applicability of interval predictions for metro passenger flow.

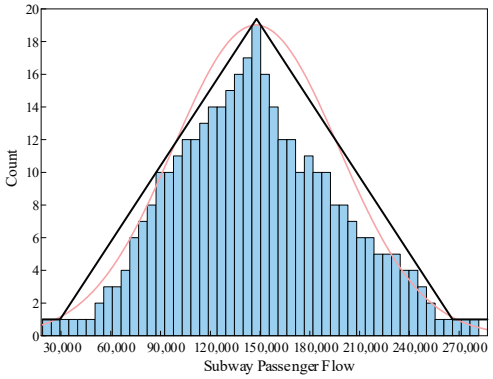


Figure 6. Histogram distribution of metro passenger flow and fuzzy membership function.

The triangular membership function for fuzzy granulation is expressed in Equation (28):

$$A(x,a,m,b) = \begin{cases} 0, & x < a \\ \frac{x-a}{m-a}, & a \le x \le m \\ \frac{b-x}{b-m}, & m < x \le b \\ 0, & x > b \end{cases} \tag{28}$$

where a, m, b represent the minimum, mean, and maximum values, respectively, characterizing the fluctuation of subway passenger flow data after window granulation.

3.3.2. Extreme Gradient Boosting Algorithm

Given that the metro passenger flow interval prediction dataset exhibits nonlinear, complex multidimensional, and random fluctuation characteristics [38], traditional probability models—which assume independent and identically distributed errors along with normal homoscedasticity—often lead to significant prediction errors. Therefore, this paper selects the Extreme Gradient Boosting (XGBoost) algorithm, known for its data-driven nature, as the method for metro passenger flow interval prediction. XGBoost is a boosting

https://doi.org/10.3390/sym18020288

machine learning algorithm that treats each data sample as a leaf node with a weight on low-precision decision trees and iteratively integrates them to form a high-precision ensemble model [39]. XGBoost can flexibly select loss functions based on the overall distribution of the data, thereby generating intervals with variable widths and local asymmetry, which enhances the accuracy and robustness of metro passenger flow interval predictions. The sum of leaf weights on the ensemble decision trees after XGBoost iterations is given by:

$$\overline{y}_i(k) = \sum_{k}^{K} f_k(x_i) \tag{29}$$

where $\overline{y}_i(k)$ is the sum of leaf weights in the ensemble model of all decision trees after the k-th iteration; $f_k(x_i)$ is the leaf weight of data sample x_i on the k-th decision tree.

The objective function of XGBoost consists of a loss function that measures the error between the true value and the output value, and a regularization function that controls the model complexity. Its mathematical model is:

$$O_{bj} = \sum_{i=1}^{m} L(y_i, \overline{y}_i) + \sum_{k=1}^{k} \Omega(f_k) \tag{30}$$

where L is the loss function, Ω is the regularization function; m is the total number of data samples on the k-th decision tree.

3.3.3. Procedure of the FIG-ICPO-XGBoost Prediction Model

Based on the analysis of subway passenger flow fluctuation characteristics and influencing factors, this study uses a 7-day cycle to partition the subway passenger flow data into fuzzy granulation windows. According to the analysis results of the three data features—LOW, R, and UP—the ICPO algorithm is employed to optimize the XGBoost parameters, thereby training and fitting a subway passenger flow fluctuation interval prediction model. The procedural steps of the FIG-ICPO-XGBoost-based subway passenger flow fluctuation interval prediction model are as follows, with the flowchart illustrated in Figure 7.

(1) Based on the fluctuation characteristics of subway passenger flow and the degree of correlation with influencing factors, perform fuzzy information granulation on the subway passenger flow data using a 7-day cycle.
(2) According to the features of the LOW, R, and UP data groups, set the algorithm population size to 25 and the number of iterations to 35.
(3) Initialize the algorithm population parameters using the Circle map.
(4) Calculate the fitness value of each individual in the population and record the optimal position.
(5) Compute and update parameters such as ω and η^* according to Equations (16) and (17).
(6) Apply the hybrid mechanism of dimension-by-dimension pinhole imaging opposition-based learning and Cauchy mutation for reverse learning. Evaluate and compare solutions based on the fitness value of the objective function. Update the solutions across all dimensions using this method, and update the position of the best crested porcupine population individual according to Equation (13).
(7) Update the positions of population individuals based on adaptive variable spiral search and inertia weight, following Equations (18)–(21).
(8) Determine whether the algorithm has reached the maximum number of iterations. If so, stop the iterative computation and output the optimal individual position and fitness value. If not, repeat steps (3) through (8).

https://doi.org/10.3390/sym18020288

(9) Output the global optimal feasible solution of the ICPO algorithm, and determine the sum of leaf weights (29) and the objective function (30) on the decision tree of the XGBoost model.

(10) Use the FIG-ICPO-XGBoost model to predict the fluctuation interval range of subway passenger flow and validate the prediction performance.

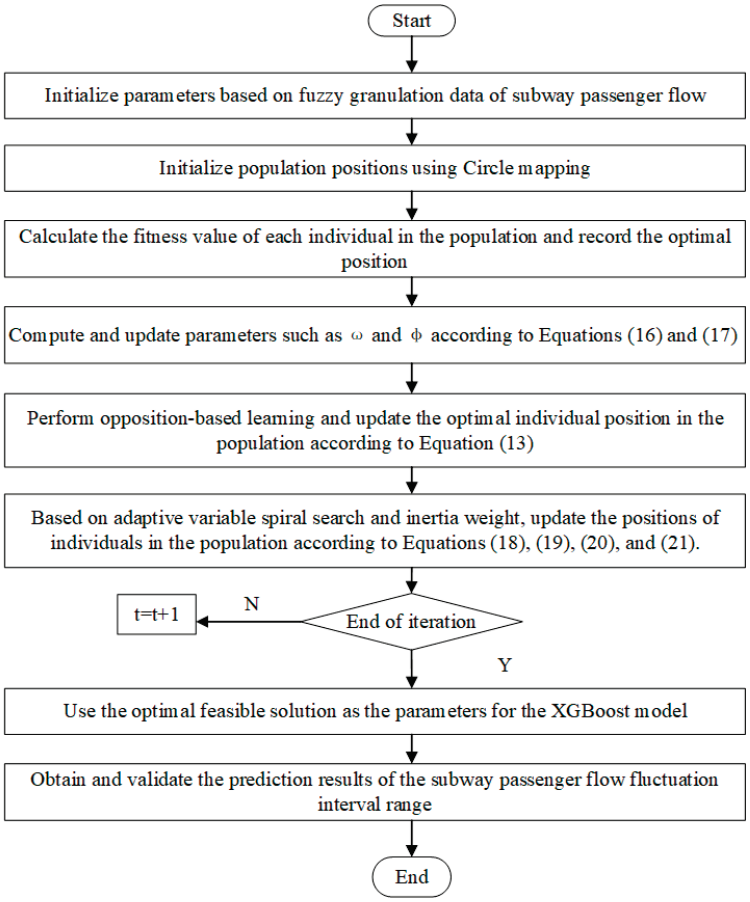


Figure 7. Flowchart of the FIG-ICPO-XGBoost prediction model.

4. Case Study

4.1. Fuzzy Information Granulation Processing of Subway Passenger Flow

Based on fare card entry data and weather condition data from Harbin Metro Line 1 spanning 357 days (from 6 January to 27 December 2020), and in accordance with the analysis results of periodic fluctuation characteristics in subway passenger flow, this study partitions the 357 days of subway passenger flow sample data into 51 fuzzy granulation windows using a 7-day cycle as the window length. Triangular fuzzy membership functions are applied for fuzzy granulation to obtain indicators representing the fluctuation interval range of subway passenger flow: the minimum value (LOW), the mean value (R), and the maximum value (UP). The sample data after fuzzy granulation processing are shown in Figure 8a, and the data normalized to the (0, 1) interval are presented in Figure 8b.

4.2. Analysis of Subway Passenger Flow Fluctuation Interval Prediction Results

In this study, the subway passenger flow data were partitioned using a 7-day window length, resulting in a total of 51 sets of sample data processed through fuzzy information granulation. The sample data from weeks 1–25 were used as the training set to construct the prediction model and fit the regression function, while the sample data from weeks 26–51 served as the test set to validate the effectiveness of the prediction model. The

https://doi.org/10.3390/sym18020288

population size parameter (POP) of the ICPO algorithm was set to 25, the maximum number of iterations (MAX) was set to 35, the lower bounds of the variable ranges (LB) were set to [0.001, 10, 0.0001], the upper bounds (UB) were set to [0.01, 50, 0.1], and the variable dimension (dim) was set to 3. Weather conditions such as temperature, air quality index, and wind scale from Section 4.1 were used as feature variables, while the LOW, R, and UP window data after fuzzy information granulation of subway passenger flow served as the output variables. The ICPO algorithm was employed to optimize and determine the regression parameters of the prediction model, thereby achieving subway passenger flow fluctuation interval prediction. To evaluate the performance of the prediction model results, linear regression analysis was conducted to compare predicted values with actual values, as shown in Figure 9. From Figure 9, it can be observed that the linear regression slopes for the LOW, R, and UP data groups are 0.8338, 0.8496, and 0.9567, respectively, with corresponding coefficients of determination (R^2) of 0.8164, 0.8173, and 0.8191. These fitting results indicate a high linear correlation between predicted and actual values, with minimal deviations, demonstrating that the proposed prediction model possesses good prediction accuracy and applicability.

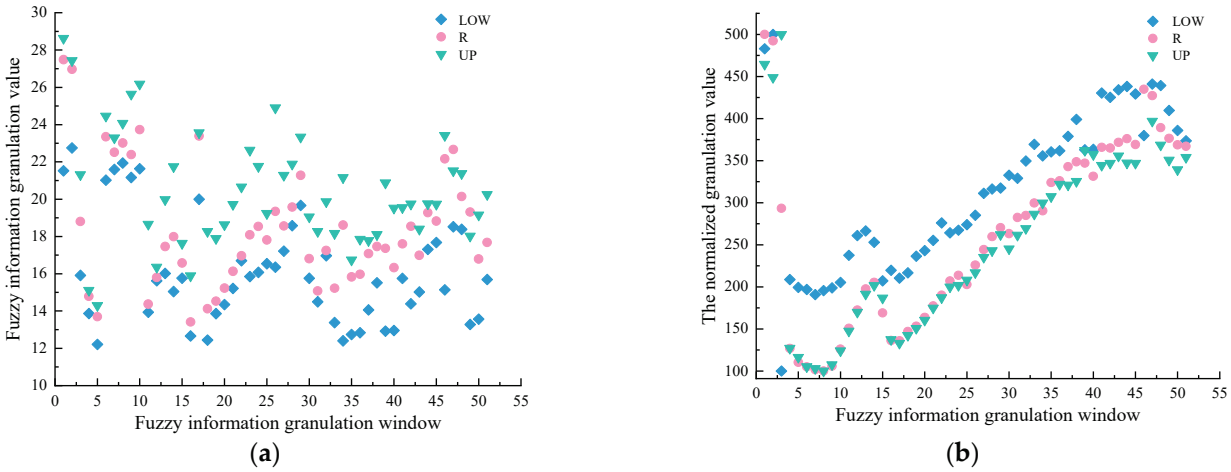


Figure 8. Granulation values: (**a**) Fuzzy information granulation values; (**b**) Normalized granulation values.

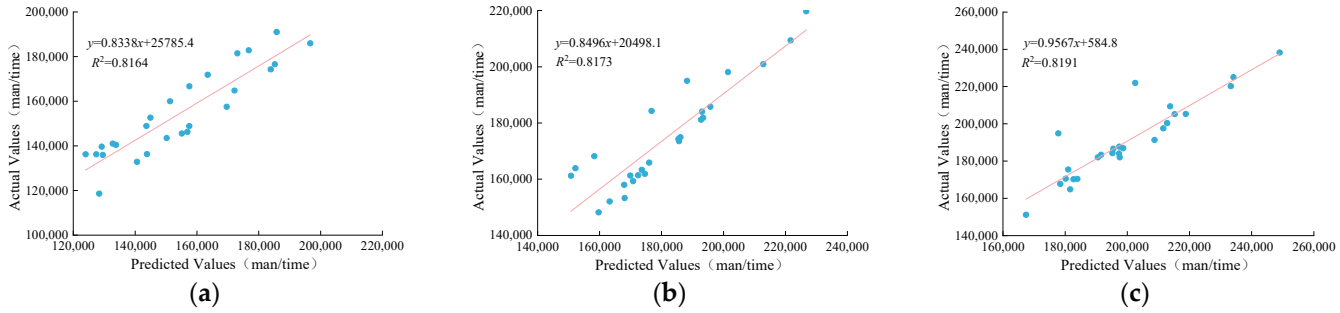


Figure 9. Correlation between actual and predicted values: (**a**) LOW; (**b**) R; (**c**) UP.

4.3. Error Analysis of Subway Passenger Flow Fluctuation Interval Prediction

To ensure the predictive performance of the ICPO-XGBoost, CPO-XGBoost, and PSO-BiLSTM models on the LOW, R, and UP datasets derived from fuzzy information granulation of metro passenger flow, the population size for all algorithms (ICPO, CPO, PSO, and GA) was set to 25, and the maximum number of iterations was uniformly set to 35. The statistical results of prediction errors for each model are shown in Figure 10. As can be seen from Figure 10, the statistical error values of the ICPO-XGBoost model are consistently

https://doi.org/10.3390/sym18020288

lower than those of the existing models. The prediction error shows a positive correlation with the fuzzy granulation values of subway passenger flow. The primary reason may be that as the fuzzy granulation interval indicators increase, the fluctuation interval range of subway passenger flow expands, leading to a significant enhancement in the internal heterogeneity and non-stationarity of the data within the fuzzy granulation windows. This makes it more challenging for the models to effectively extract data feature patterns, thereby reducing prediction accuracy and generalization capability.

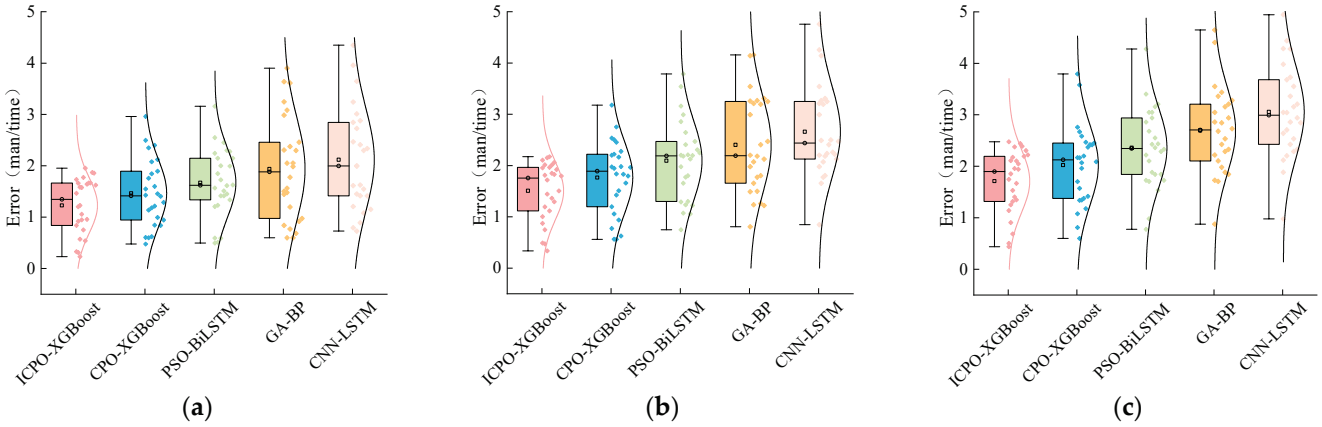


Figure 10. Comparative analysis of errors: (**a**) LOW; (**b**) R; (**c**) UP.

From Table 6, it can be observed that the mean values of MAE, RMSE, and MAPE for the LOW, R, and UP datasets predicted by ICPO-XGBoost after fuzzy granulation of subway passenger flow are 10,291, 10,612, and 5.8%, respectively. Compared to CPO-XGBoost, these values represent improvements of 4541, 5258, and 2.6%, respectively; compared to PSO-BiLSTM, improvements of 7223, 8250, and 4.0%, respectively; compared to GA-BP, improvements of 10,124, 11,065, and 5.5%, respectively; and compared to CNN-LSTM, improvements of 13,161, 14,613, and 7.2%, respectively. These results indicate that ICPO-XGBoost demonstrates superior accuracy in predicting subway passenger flow fluctuation intervals. Furthermore, the MAPE values of all models show a positive correlation with the fuzzy granulation values, suggesting that as the fuzzy granulation values increase, the prediction errors for subway passenger flow also increase. The ICPO-XGBoost model exhibits the smallest relative change in this trend, indicating its strong robustness and applicability when predicting subway passenger flow across different fuzzy granulation intervals.

Table 6. Performance metrics of subway passenger flow prediction models.

Category	LOW			R			UP		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
ICPO-XGBoost	8485	8677	5.6	10,350	10,592	5.8	12,036	12,566	6.0
CPO-XGBoost	12,293	13,386	8.2	15,075	16,074	8.5	17,124	18,147	8.7
PSO-BiLSTM	14,655	16,027	9.5	17,663	18,971	9.7	20,222	21,587	10.1
GA-BP	16,697	17,939	10.8	20,947	22,302	11.4	23,599	24,789	11.9
CNN-LSTM	19,369	21,754	12.5	24,039	25,746	13.1	26,945	28,173	13.5

To further compare and analyze the performance of the FIG-ICPO-XGBoost-based metro passenger flow fluctuation interval prediction model, the study selects metrics such as the Prediction Interval Coverage Probability (PICP) and the Prediction Interval Normalized Average Width (PINAW) to measure the coverage capability of the prediction intervals and their compactness, respectively [40]. PICP reflects the proportion of metro passenger flow fluctuation interval predictions that cover the actual values and is a core metric for

https://doi.org/10.3390/sym18020288

evaluating the reliability of the model's interval predictions. A PICP value closer to the model's nominal coverage rate (95% in this study) indicates higher reliability in covering the true values. PINAW measures the average width of the prediction intervals and is standardized by the range of the true values to eliminate scale effects [41]. The PINAW value is inversely proportional to the compactness of the prediction intervals. A smaller PINAW value indicates a better ability to characterize the fluctuation features of metro passenger flow prediction intervals. As shown in Table 7, the PICP and PINAW values for the FIG-ICPO-XGBoost model are 96.12% and 0.1762, respectively. Compared to CPO-XGBoost, PSO-BiLSTM, GA-BP, and CNN-LSTM, these values represent improvements of 3.61–10.45% in PICP and reductions of 0.1092–0.2361 in PINAW. These findings demonstrate that the FIG-ICPO-XGBoost model exhibits superior performance in predicting metro passenger flow fluctuation intervals.

Table 7. Interval performance metrics of metro passenger flow prediction models.

Category	PICP	PINAW
FIG-ICPO-XGBoost	96.12%	0.1762
CPO-XGBoost	92.51%	0.2854
PSO-BiLSTM	90.68%	0.3478
GA-BP	88.67%	0.4123
CNN-LSTM	85.67%	0.3918

From the data in Table 8, it can be seen that the lower and upper bounds of the subway passenger flow data intervals after triangular fuzzy information granulation encompass the actual range of subway passenger flow data. Combining Figure 10 and the data from Table 6, it is observed that the ICPO-XGBoost model predicts the fuzzy granulation values for the 51st week as 146,313 (LOW), 184,276 (R), and 221,954 (UP) for the LOW, R, and UP datasets, respectively. Compared to existing prediction models, these results demonstrate higher accuracy. Furthermore, compared to the previous week's fuzzy granulation values of 143,641, 167,950, and 191,562, the model predictions show an overall upward trend, which aligns closely with the actual data trends. This indicates the reliability of the model's predictions, providing valuable data support and decision-making basis for formulating subway vehicle operation scheduling plans and passenger flow management strategies at stations, thereby demonstrating a certain level of practicality.

Table 8. Prediction of subway passenger flow fluctuation intervals.

Category	Time Series							Subway Passenger Flow Fluctuation Interval
Date	14	15	16	17	18	19	20	Actual Variation Range (Fuzzy Granule Description)
Actual Passenger Flow	191,502	189,207	187,950	184,405	190,364	161,078	191,502	[Low, R, UP] = [143,641, 167,950, 191,562]
Date	21	22	23	24	25	26	27	Predicted Variation Range (Fuzzy Granule Description)
Actual Passenger Flow	189,673	186,801	183,295	194,737	207,464	155,280	189,673	[Low, R, UP] = [146,313, 184,276, 221,954]

To analyze the applicability of the ICPO-XGBoost model, this paper statistically examines the average running time of the subway passenger flow interval prediction model, as

https://doi.org/10.3390/sym18020288

shown in Figure 11. According to Figure 11, the average running time of the ICPO-XGBoost model is 498.6 s, which is reduced by 8.1 s, 18.2 s, and 44.5 s compared to the CPO-XGBoost, PSO-BiLSTM, and GA-BP models, respectively. Although the running time of the ICPO-XGBoost model is longer than that of the CNN-LSTM model, it achieves higher accuracy in the interval prediction results for subway passenger flow. Therefore, it demonstrates good applicability.

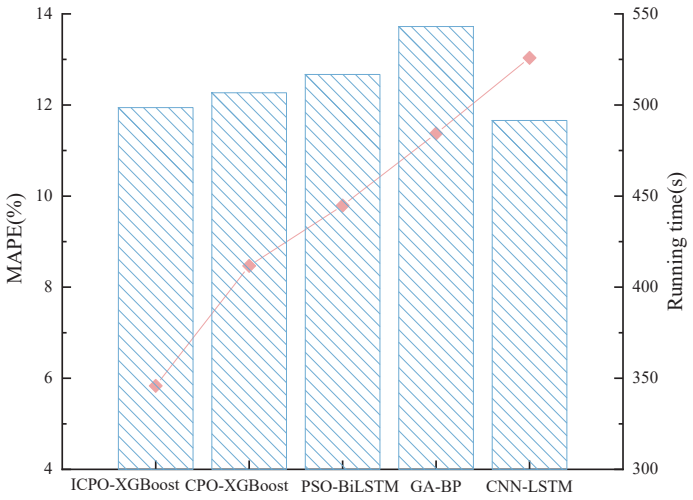


Figure 11. Average running time of each model algorithm.

4.4. Sensitivity Analysis of Parameters in the Subway Passenger Flow Fluctuation Interval Prediction Model

In the subway passenger flow fluctuation interval prediction model, the maximum number of iterations is the upper limit for how many times the ICPO algorithm updates the positions of other samples in an attempt to approach the global optimal solution after determining the current optimal sample position, based on the characteristics of the dataset. To investigate the impact of the maximum iteration count on subway passenger flow prediction performance, this paper selected the fuzzy-granulated R dataset for analysis. Under fixed conditions of other parameters in the prediction model, the maximum iteration count was set sequentially to 5, 10, 15, . . ., 50. The resulting MAPE values of the predictions are shown in Figure 12a. Analysis of Figure 12a reveals that when the maximum iteration count of the ICPO algorithm exceeds 35, the MAPE value does not decrease significantly with further increases in iterations, indicating that prediction accuracy stabilizes, while the model runtime increases notably. Beyond this threshold, the MAPE value and iteration count are not negatively correlated. Setting an excessively high iteration count for the ICPO algorithm can lead to samples becoming trapped in ineffective repetitive iterations, reducing the operational efficiency and applicability of the prediction model. Therefore, setting a reasonable maximum iteration count helps balance prediction accuracy and runtime.

The population size in the subway passenger flow fluctuation interval prediction model is the search space range determined by the ICPO algorithm during each iteration based on dataset characteristics. A larger population size aids in enhancing the ICPO algorithm's exploration capability within the solution space, thereby increasing the probability of discovering the global optimal feasible solution. To investigate the impact of population size on subway passenger flow prediction performance, this paper selected the fuzzy-granulated R dataset for analysis. Under fixed conditions of other parameters in the prediction model, the population size was set sequentially to 5, 10, 15, . . ., 50. The resulting MAPE values of the predictions are shown in Figure 12b. Analysis of Figure 12b shows that when the population size of the ICPO algorithm is below 25, the MAPE value decreases as the population size increases, exhibiting a certain negative correlation trend. However,

https://doi.org/10.3390/sym18020288

when the population size exceeds this threshold, the MAPE value does not improve significantly with further increases in population size, while the model runtime increases substantially. The subway passenger flow prediction results indicate that although an overly large population size can expand the search space, it may lead to redundant computations of data feature information. Setting an excessively large population size for the ICPO algorithm not only fails to effectively improve prediction accuracy but also reduces the operational efficiency and applicability of the prediction model. Therefore, setting a reasonable population size helps balance prediction accuracy and operational efficiency.

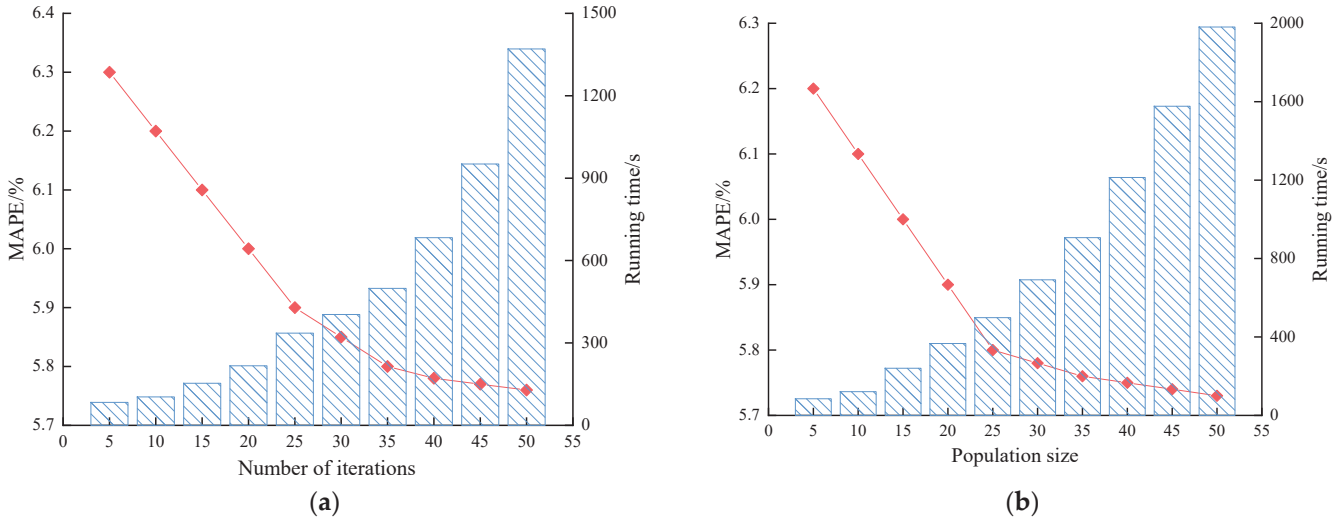


Figure 12. Impact of model parameters on recognition results: (**a**) Number of iterations; (**b**) Population size.

5. Conclusions

Based on the IC card data of Harbin Metro Line 1 and the analysis of correlations between weather conditions and subway passenger flow data, this paper constructed a FIG-ICPO-XGBoost model for predicting subway passenger flow fluctuation intervals. Twelve benchmark test functions and fuzzy information granulated subway passenger flow data were selected for numerical simulation experiments and prediction performance validation. The main research conclusions are as follows:

(1) This paper employed hybrid strategies such as Circle mapping, a hybrid mechanism combining dimension-by-dimension pinhole imaging opposition-based learning and Cauchy mutation, adaptive variable spiral search, and inertia weight to improve the Crested Porcupine Optimizer (CPO). These improvements focused on population initialization and individual position update methods during the perception and defense phases. While preserving the advantages of CPO, these strategies increased population diversity and quality, enhanced the algorithm's ability to balance global exploration and local exploitation, and mitigated the issue of premature convergence to local optima, which typically leads to insufficient optimization accuracy. These improvement strategies collectively strive to establish and maintain an efficient symmetric or balanced relationship within the algorithm's search process, such as uniformity in initial distribution (spatial symmetry), mirror-like expansion of the solution space generated by opposition-based learning (mirror symmetry), and dynamic trade-offs between exploration and exploitation forces (process symmetry). Using the Wilcoxon rank-sum test, the numerical simulation results and convergence characteristics of the 12 benchmark test functions were compared and analyzed. The results show that the mean and standard deviation of the solutions obtained by the

https://doi.org/10.3390/sym18020288

Improved CPO (ICPO) algorithm are closer to the optimal values. Compared to algorithms such as CPO, SSA, WOA, and PSO, the ICPO algorithm demonstrates better convergence accuracy and speed. Simulation results on unimodal and multimodal benchmark functions indicate that using Circle mapping and the hybrid mechanism of dimension-by-dimension pinhole imaging opposition-based learning and Cauchy mutation effectively improves the algorithm's local exploitation and global exploration capabilities. Simulation results on composite modal benchmark functions demonstrate that integrating adaptive variable spiral search and inertia weight strategies helps enhance the CPO algorithm's ability to balance global search and local development.

(2) This paper applied the fuzzy information granulation method to reduce the spatiotemporal complexity of subway passenger flow data while preserving its fluctuation characteristics, and constructed a FIG-ICPO-XGBoost model for predicting subway passenger flow fluctuation intervals. For the LOW, R, and UP datasets after fuzzy granulation, the model achieved mean MAE, RMSE, and MAPE values of 10,291, 10,612, and 5.8%, respectively. Compared to existing prediction models, these represent improvements ranging from 4541 to 13,161 for MAE, 5258 to 14,613 for RMSE, and 2.6% to 7.2% for MAPE, demonstrating good prediction accuracy and robustness. The findings contribute to reasonable anticipation of subway passenger flow fluctuation intervals, providing data support and a decision-making basis for formulating subway vehicle operation schedules and assisting passengers in planning travel routes and times effectively.

Based on the IC card data and weather condition data of the Harbin metro passenger flow, this paper validates the applicability of the FIG-ICPO-XGBoost model. However, due to the difficulty of obtaining metro passenger flow data from other cities, the study did not further verify the model's applicability and robustness using datasets from different urban contexts. Future research should address the following aspects:

(1) Conduct comparative analyses of the relationships between metro passenger flow and various influencing factors—such as weather conditions, built environment, traffic operational status, socioeconomic factors, and network structures—across different cities. This would help uncover the spatiotemporal distribution patterns of metro passenger flow in diverse urban settings and develop prediction models that incorporate both the influencing factors and spatiotemporal dynamic characteristics. Such efforts are expected to further enhance the accuracy and applicability of metro passenger flow fluctuation interval predictions.

(2) Based on the analysis of the spatiotemporal dynamic characteristics of subway passenger flow in different cities, the optimization mechanism and structural features of the Crested Porcupine Optimizer algorithm are improved. A deep learning method is designed to dynamically adjust hyperparameters and the weights of influencing factors according to the spatiotemporal distribution characteristics of subway passenger flow. This enhances the scalability of the subway passenger flow interval prediction model and overcomes the limitations of insufficient prediction accuracy caused by the specificity of subway passenger flow data.

Author Contributions: Conceptualization, C.M., Y.F., L.M. and X.L.; Methodology, L.M. and J.Z.; Software, Q.H.; Validation, X.L.; Investigation, Q.H.; Data Curation, Y.F., X.L. and J.Z.; Writing—Original Draft Preparation, Q.H., L.M. and J.Z.; Writing—Review and Editing, Y.F. and C.M.; Visualization, Q.H. and Y.F.; Supervision, C.M.; Funding Acquisition, Q.H. and C.M. All authors have read and agreed to the published version of the manuscript.

https://doi.org/10.3390/sym18020288

Funding: This research was funded by the Lanzhou 2025 Annual Philosophy and Social Sciences Planning Project (grant number 25-A21), the Young Scholars Science Foundation of Lanzhou Jiaotong University (grant number 2025021) and the National Natural Science Foundation of China (grant number 72571121).

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors give grateful thanks to all reviewers for their selfless contributions to science and to all editors for their outstanding work in improving the quality of this paper.

Conflicts of Interest: The author Lin Ma was employed by Linxia Daohe Investment Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CPO	Crested Porcupine Optimizer
ICPO	Improved Crested Porcupine Optimizer
SSA	Sparrow Search Algorithm
WOA	Whale Optimization Algorithm
PSO	Particle Swarm Optimization

References

1. Lyu, T.; Xu, M.; Zhang, J.; Wang, Y.; Yang, L.; Gao, Y. Influential Factor Analysis and Prediction on Initial Metro Network Ridership in Xi'an, China. *J. Adv. Transp.* **2022**, *2022*, 2842949. [CrossRef]
2. Chen, E.; Ye, Z.; Wang, C.; Xu, M. Subway Passenger Flow Prediction for Special Events Using Smart Card Data. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 1109–1120. [CrossRef]
3. Zuo, T.; Tang, S.; Zhang, L.; Kang, H.; Song, H.; Li, P. An Enhanced TimesNet-SARIMA Model for Predicting Outbound Subway Passenger Flow with Decomposition Techniques. *Appl. Sci.* **2025**, *15*, 2874. [CrossRef]
4. Nong, X.; Geng, M.; Jia, X. Prediction of Land Use around Urban Metro Stations Using the CA-Markov Model. *J. Asian Archit. Build. Eng.* **2025**, *24*, 3177–3189. [CrossRef]
5. Zhang, J.; Chen, F.; Cui, Z.; Guo, Y.; Zhu, Y. Deep Learning Architecture for Short-Term Passenger Flow Forecasting in Urban Rail Transit. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 7004–7014. [CrossRef]
6. Yu, L.; Cui, M.; Dai, S. Deviation of Peak Hours for Metro Stations Based on Least Square Support Vector Machine. *PLoS ONE* **2023**, *18*, e0291497.
7. Da Silva, D.; Shalaby, A. Forecasting Short-Term Subway Passenger Flow Using Wi-Fi Data: Comparative Analysis of Advanced Time-Series Methods. *J. Intell. Transp. Syst.* **2024**, *10*, 1–21.
8. Chen, D.; Li, H.; Sun, S.; Bai, J.; Huang, Z. Point and Interval Forecasting Approach for Short-Term Urban Subway Passenger Flow Based on Residual Term Decomposition and Fuzzy Information Granulation. *Appl. Soft Comput.* **2024**, *166*, 112187. [CrossRef]
9. Huang, X.; Wang, Y.; Lin, P.; Yu, H.; Luo, Y. Forecasting the All-Weather Short-Term Metro Passenger Flow Based on Seasonal and Nonlinear LSSVM. *Promet-Traffic Transp.* **2021**, *33*, 217–231.
10. Xu, X.; Zhang, K.; Mi, Z.; Wang, X. Short-Term Passenger Flow Prediction During Station Closures in Subway Systems. *Expert Syst. Appl.* **2024**, *236*, 121362. [CrossRef]
11. Xing, Z.; Huang, M.; Peng, D. Overview of Machine Learning-Based Traffic Flow Prediction. *Digit. Transp. Saf.* **2023**, *2*, 164–175. [CrossRef]
12. Cao, B.; Li, Y.; Chen, Y.; Yang, A. A CNN-LSTM Model for Short-Term Passenger Flow Forecast Considering the Built Environment in Urban Rail Transit Stations. *J. Transp. Eng. Part A Syst.* **2024**, *150*, 04024072. [CrossRef]
13. Nagaraj, N.; Gururaj, H.L.; Swathi, B.H.; Hu, Y.-C. Passenger Flow Prediction in Bus Transportation System Using Deep Learning. *Multimed. Tools Appl.* **2022**, *81*, 12519–12542. [CrossRef]
14. Baghbani, A.; Rahmani, S.; Bouguila, N.; Patterson, Z. TMS-GNN: Traffic-Aware Multistep Graph Neural Network for Bus Passenger Flow Prediction. *Transp. Res. Part C Emerg. Technol.* **2025**, *174*, 105107. [CrossRef]
15. Qi, Q.; Cheng, R.; Ge, H. Short-Term Inbound Rail Transit Passenger Flow Prediction Based on BILSTM Model and Influence Factor Analysis. *Digit. Transp. Saf.* **2023**, *2*, 12–22. [CrossRef]

https://doi.org/10.3390/sym18020288

16. Pu, Y.; Xu, X.; Fan, Q.; Zhang, S.; Chen, J. Short-Term Interval Prediction of Inbound Passenger Flow of Subway Station Under Failure Events. *J. Adv. Transp.* **2024**, *2024*, 6833793. [CrossRef]
17. Kwak, J.; Ku, D.; Jo, J.; Wong, S.; Lee, S.; Lee, S. A Hybrid Clustering–Regression Approach for Predicting Passenger Congestion in a Carriage at a Subway Platform. *Expert Syst. Appl.* **2025**, *268*, 126169. [CrossRef]
18. Chen, D.; Kun, Y.; Chun, G. The Relationship Between Built Environment Characteristics and Metro Stations Pedestrian Catchment Areas. *J. Adv. Transp.* **2024**, *2024*, 2196799. [CrossRef]
19. Mo, Y.; Zhang, J.; Wang, C.; Hao, X.; Yang, L.; Gao, Z. A Semi-Conv-Transformer Model for Inflow Prediction of Newly Expanding Subway Lines. *IEEE Trans. Intell. Transp. Syst.* **2025**, *26*, 8497–8512. [CrossRef]
20. Ma, C.; Liu, T. Survey of Short-Term Traffic Flow Prediction Based on LSTM. *Int. J. Mod. Phys. C* **2025**, *36*, 2450177. [CrossRef]
21. Xie, C.; Li, X.; Chen, B.; Lin, F.; Lin, Y.; Huang, H. Subway Sudden Passenger Flow Prediction Method Based on Two Factors: Case Study of the Dongsishitiao Station in Beijing. *J. Adv. Transp.* **2021**, *2021*, 5577179. [CrossRef]
22. Sivakumar, R.; Angayarkanni, S.A.; Ramana Rao, Y.V. Traffic Flow Forecasting Using Natural Selection Based Hybrid Bald Eagle Search—Grey Wolf Optimization Algorithm. *PLoS ONE* **2022**, *17*, e0275104.
23. Jiang, Q. GMM Clustering Based on WOA Optimization and Space–Time Coupled Urban Rail Traffic Flow Prediction by CEEMD–SE–BiGRU–AM. *Mob. Inf. Syst.* **2022**, *2022*, 7846630. [CrossRef]
24. Liu, S.; Yao, E. Holiday Passenger Flow Forecasting Based on the Modified Least-Square Support Vector Machine for the Metro System. *J. Transp. Eng. Part A Syst.* **2017**, *143*, 04016005. [CrossRef]
25. Liu, J.; He, Q.; Yue, Z.; Pei, Y. A Hybrid Strategy-Improved SSA-CNN-LSTM Model for Metro Passenger Flow Forecasting. *Mathematics* **2024**, *12*, 3929. [CrossRef]
26. Tang, M.; Wang, H.; Qiu, J.; Tao, Z.; Yang, T. Classification of Offshore Wind Grid-Connected Power Quality Disturbances Based on Fast S-Transform and CPO-Optimized Convolutional Neural Network. *PLoS ONE* **2024**, *19*, e0314720. [CrossRef]
27. Zhang, H.; Guo, C.; Zhai, D.; Wang, Y.; Liu, H.; Chen, F.; Xu, D. Application of Improved Crown Porcupine Optimizer in UAV Path Planning Based on Dynamic Weighted JAYA-CPO Attack Strategy. *Prot. Control Mod. Power Syst.* **2025**, *10*, 101–127. [CrossRef]
28. Wang, D.; Gao, B.; Zhang, L. Optimization on Electric Construction Machinery Considering Time-of-Use Electricity Price Based on the Improved Crested Porcupine Optimizer Algorithm. *Energy Sci. Eng.* **2025**, *13*, 2973–2986. [CrossRef]
29. Liu, S.; Jin, Z.; Lin, H.; Lu, H. An Improve Crested Porcupine Algorithm for UAV Delivery Path Planning in Challenging Environments. *Sci. Rep.* **2024**, *14*, 20445. [CrossRef]
30. Adalja, D.; Patel, P.; Mashru, N.; Jangir, P.; Jangid, R.; Gulothungan, G.; Khishe, M. A New Multi Objective Crested Porcupines Optimization Algorithm for Solving Optimization Problems. *Sci. Rep.* **2025**, *15*, 14380. [CrossRef] [PubMed]
31. Gao, Q.; Jin, A.; Wang, C.; Zhang, L.; Zhang, Q.; Xiong, X.; Yang, F. AEGWCPO: A Cooperative Path Planning Method for UAV Swarm Penetration in Complex Mountainous Environments. *Meas. Sci. Technol.* **2025**, *36*, 116301. [CrossRef]
32. Chen, S.; Chen, C.; Li, Y.; Meng, L.; Wei, L.; Guan, B. Optimization of Operating Parameters Scheme for Water Injection System Based on a Hybrid Particle Swarm–Crested Porcupine Algorithm. *Sustainability* **2025**, *17*, 8057. [CrossRef]
33. Liu, H.; Zhou, R.; Zhong, X.; Yao, Y.; Shan, W.; Yuan, J.; Xiao, J.; Ma, Y.; Zhang, K.; Wang, Z. Multi-Strategy Enhanced Crested Porcupine Optimizer: CAPCPO. *Mathematics* **2024**, *12*, 3080. [CrossRef]
34. Sriprateep, K.; Pitakaso, R.; Khonjun, S.; Srichok, T.; Luesak, P.; Gonwirat, S.; Kaewta, C.; Kosacka-Olejnik, M.; Enkvetchakul, P. Multi-Objective Optimization of Resilient, Sustainable, and Safe Urban Bus Routes for Tourism Promotion Using a Hybrid Reinforcement Learning Algorithm. *Mathematics* **2024**, *12*, 2283. [CrossRef]
35. Abdel-Basset, M.; Mohamed, R.; Abouhawwash, M. Crested Porcupine Optimizer: A New Nature-Inspired Metaheuristic. *Knowl.-Based Syst.* **2024**, *284*, 111257. [CrossRef]
36. Gao, Z.M.; Zhao, J.; Zhang, Y.J. Review of Chaotic Mapping Enabled Nature-Inspired Algorithms. *Math. Biosci. Eng.* **2022**, *19*, 8215–8258.
37. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
38. Zhang, J.; Pei, Y.; Sun, J.; Lai, Y.; Wang, S.; Easa, S.M. Achieving bus electrification: Strategy for bus fleet replacement in cold region. *Transp. Res. Part D Transp. Environ.* **2025**, *145*, 104825. [CrossRef]
39. Wang, T.; Han, Y.; Li, W.; Ye, X.; Yuan, Q. A Takeover Risk Assessment Approach Based on an Improved ANP-XGBoost Algorithm for Human–Machine Driven Vehicles. *IEEE Access* **2024**, *12*, 48379–48387. [CrossRef]
40. Nikulchev, E.; Chervyakov, A. Prediction intervals: A geometric view. *Symmetry* **2023**, *15*, 781. [CrossRef]
41. Zha, W.; Li, X.; Du, Y.; Liang, Y. Interval forecast method for wind power based on GCN-GRU. *Symmetry* **2024**, *16*, 1643. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/sym18020288

symmetry

MDPI

Article

Solving Bilevel Multi-Robot Cooperative Path Planning Problems via a Memetic Framework

Zhixin Wang [1], Shi Cheng [1,*], Yifei Sun [2,*], Sicheng Hou [2] and Mingming Zhang [3]

[1] School of Artificial Intelligence and Computer Science, Shaanxi Normal University, Xi'an 710019, China; 2023wzx@snnu.edu.cn
[2] School of Physics and Information Technology, Shaanxi Normal University, Xi'an 710019, China; sicheng_hou@snnu.edu.cn
[3] Department of Mechanical and Mechatronics Engineering, The University of Auckland, Auckland 1010, New Zealand; mzha509@aucklanduni.ac.nz
* Correspondence: cheng@snnu.edu.cn (S.C.); yifeis@snnu.edu.cn (Y.S.)

Abstract

With the increasing use of multi-robot systems in emergency scenarios, collaborative path planning for robots has attracted greater attention. The multi-robot path-planning problem was modeled as a bilevel cooperative path planning model and solved using a memetic algorithm with a dynamic window approach and a parking scheduling strategy (MA-DWAPSS). The bilevel path planning model has divided the problem into two parts: global (static) path planning to find a near-optimal route and dynamic path planning to avoid path conflicts. Corresponding to the proposed MA-DWAPSS method, an improved memetic algorithm was developed based on genetic algorithm to find an optimal global path and a cubic Bézier curve to smooth the path and avoid sharp turns. The dynamic window approach (DWA) and parking scheduling strategy (PSS) obtain real-time sensor data and coordinate the docking and movement of robots in dynamic environments, handling obstacles in real time and preventing conflicts or unnecessary stops to improve efficiency. DWA further accounts for the dynamic characteristics of robot motion, making the path planning flexible and adaptive to rapid environmental changes. Simulation results show that the proposed method outperforms three other algorithms in path distance, time, obstacle avoidance, and smoothness.

Keywords: memetic algorithm; dynamic window approach; multi-robot path planning; bilevel model; parking scheduling strategy

1. Introduction

In recent years, the application of multi-robot systems (MRSs) [1] has grown significantly in emergency scenarios due to their advantages in efficiency, coverage, and reliability [2]. For example, firefighting missions are often highly dangerous, complex, and have uncertain environments for multiple robots. Traditional human intervention faces many challenges, including dynamic and fixed obstacles caused by high temperatures and unstable structures at the scene, which make rescue operations difficult and time-consuming. Robots need to plan an effective global route in complex environments. Fire scenes may have many obstacles, such as collapsed buildings, piles of rubble, and rolling flames. Finding the optimal path between these obstacles is a key problem in path planning. Traditional global path planning algorithms work well in known environments, but their effectiveness is limited when the environment changes or when unknown factors

are introduced. In dynamic, uncertain fire rescue environments, adjusting paths to account for environmental changes is a critical issue.

The challenge is how to achieve multi-robot collaborative dynamic obstacle avoidance in local dynamic environments. Fire scenes are constantly changing, and in addition to fixed obstacles, robots must avoid suddenly appearing dynamic obstacles, such as flames or collapsing objects. Multi-robot collaboration not only requires individual robots to have path planning abilities, but also demands dynamic adjustment of their tasks and paths, improving overall rescue efficiency.

The balance between global and local path planning is a key challenge in multi-robot path planning, ensuring that robots can perform multiple tasks efficiently in complex environments [3]. For example, a hybrid algorithm that combines the global search capability of genetic algorithms with the local search advantages of simulated annealing optimization has been introduced [4].

A symmetrical process was proposed to solve multi-robot cooperative path planning problems [5]. The method combines an improved memetic algorithm (MA) [6] with a dynamic window approach (DWA) [7] featuring a parking scheduling strategy (PSS). By integrating these techniques, the proposed approach effectively balances global and local planning in rescue scenarios, thereby enhancing collaboration and path planning efficiency in multi-robot systems. This hybrid approach ensures the generation of optimal global paths while avoiding sharp turns or abrupt directional changes that could destabilize robot trajectories. Furthermore, the generated paths are refined using cubic Bézier curve fitting to enhance smoothness and reduce sudden directional variations that may hinder robot navigation.

In the proposed algorithm, an improved memetic algorithm was used for global (static) path planning. DWA was used for robot dynamics, including linear velocity, angular velocity, and acceleration. This method dynamically adjusts paths in real time to accommodate environmental changes. To enhance this capability, we integrate a Parking Scheduling Strategy (PSS) into DWA (DWAPSS). This improved approach enables robots to avoid obstacles effectively and respond quickly to dynamic changes, such as moving obstacles or changing terrain. The PSS mechanism also optimizes pause positions when robots encounter dynamic obstacles. It ensures that robots maintain collision-free, near-optimal trajectories in real time.

Robotic environments are complex, with unpredictable obstacles and the need for timely decisions. These factors often limit traditional path planning. Two contributions are made in this paper:

- The bilevel path planning model has divided the problem into two parts: global (static) path planning to find a near-optimal route and dynamic path planning to avoid path conflicts.
- The proposed MA-DWAPSS combines a memetic algorithm and DWAPSS. This aims to improve accuracy and efficiency, ensuring robots complete tasks swiftly while avoiding collisions.

The paper is structured as follows. Section 2 reviews existing robot path planning algorithms. Section 4 describes the proposed algorithm in detail. Section 5 presents experimental results and analysis. Section 6 concludes and discusses implications for future research.

2. Related Works

Various algorithms have been proposed for solving the multi-robot path planning problem, and they can be classified into three categories: traditional path planning, bio-inspired intelligence path planning, and reinforcement learning-based path planning [8].

https://doi.org/10.3390/sym18030499

The proposed MA-DWAPSS algorithm integrates bio-inspired intelligent methods with traditional path planning methods.

Traditional path planning algorithms mainly rely on classical mathematical models and algorithms, using pre-known environmental maps for planning. These algorithms typically assume that the environment is static and that the robot can accurately perceive its current state. They can be divided into two categories: global path planning and local path planning [9,10]. Global path planning algorithms assume that the environment is known and usually compute the optimal path from the starting point to the destination by modeling the entire environment. These algorithms focus on the robot's overall movement path and do not consider dynamic changes in the local environment. By modeling the entire environment, precise calculations can be made, resulting in a theoretically optimal path in static environments. However, for large-scale and complex environments, global path planning algorithms may require significant computational resources, especially during environment modeling, which can be time- and resource-intensive. Moreover, they lack sensitivity to dynamic environments, as they cannot adapt to real-time changes. Common global path planning algorithms include A* [11], Dijkstra [12], dynamic programming (DP) [13], and Rapidly exploring Random Trees (RRT) [14].

Local path planning algorithms focus on obstacles or dynamic changes in the environment that the robot encounters during operation, adjusting the robot's path in real time. These algorithms rely on real-time sensor data rather than the global environment map. The task of local path planning is to ensure the robot avoids obstacles while adjusting its motion trajectory. While local path planning has real-time dynamic capabilities, it only considers local information and cannot comprehensively understand the entire environment. As a result, it may lead to locally optimal solutions instead of globally optimal ones. Particularly in complex or unknown environments, local path planning can fall into local pitfalls (such as a "deadlock" situation), making it difficult to find the best path. Common local path planning algorithms include the Artificial Potential Field (APF) [15], curve fitting-based algorithms [16], sampling and numerical optimization-based algorithms [17], DWA [7], LPA* [18], and D* Lite [19].

Bio-inspired intelligence path planning algorithms draw inspiration from biological systems and mimic the behavior of animals in nature to solve path planning problems. These algorithms do not rely on traditional mathematical models. Instead, they simulate biological behaviors, such as swarm intelligence and genetic mechanisms, to navigate through environments [20]. These algorithms are particularly effective in dynamic and complex environments. When faced with real-time updates, such as moving obstacles or changing paths, they can quickly adjust the planned route, demonstrating strong adaptability. Additionally, by imitating natural processes, these algorithms can search for globally optimal solutions, overcoming the limitations of traditional path planning methods in handling complex and dynamic environments. Biologically inspired algorithms are robust, capable of addressing uncertainties and randomness. For instance, even with incomplete or inaccurate environmental data, these algorithms can still find viable paths. However, they generally require extensive simulations and involve randomness, which increases computational load. In large-scale environments, these algorithms might take a longer time to search and optimize the path. Compared to classical algorithms, biologically inspired methods might suffer from slower time efficiency. The performance of these algorithms is also highly sensitive to the choice of parameters. Improper tuning can result in slow convergence or even failure to converge. While these algorithms often find globally optimal solution, they may still fall into local optima in complex environments with multiple local solutions. Common biologically inspired path planning algorithms include genetic

https://doi.org/10.3390/sym18030499

algorithm (GA) [21], particle swarm optimization (PSO) [22], and ant colony optimization (ACO) [23].

Reinforcement learning-based path planning algorithms optimize path decisions by learning from historical data or by interacting with the environment. Unlike traditional methods, they do not rely on precise environmental models; instead, they train models to predict or generate optimal paths. These algorithms can adapt to dynamic, complex environments. When faced with real-time changes, they can quickly modify their strategies, showing remarkable adaptability. Reinforcement learning algorithms find globally optimal solutions by maximizing accumulated rewards, overcoming the limitations of traditional path planning methods in high-dimensional, nonlinear, or partially observable environments [24]. However, reinforcement learning algorithms typically require extensive interaction experience, leading to higher computational costs. Training deep neural networks, for example, may require extended optimization times. During the initial training phase, reinforcement learning algorithms can be less efficient compared to classical methods. Furthermore, the performance of these algorithms is highly dependent on model structure, hyperparameter selection, and training strategies. Poor tuning of these parameters can lead to slow convergence or suboptimal performance. While reinforcement learning algorithms often approach optimal solutions, they may still converge to suboptimal strategies in highly complex or dynamic environments. Common reinforcement learning-based path planning algorithms include Q-learning [25], deep Q-networks (DQNs) [26], and dueling double deep Q-Networks (D3QNs) [27].

Robot path planning encompasses numerous algorithms, each with distinct advantages and limitations (see Table 1: Classification of path planning algorithms). However, critical challenges persist in this field. To address these limitations, researchers have proposed hybrid algorithms that integrate the strengths of existing methods. For example, Li et al. [28] developed a fused path planning algorithm that combines an improved GA with the DWA. Their method uses the GA to generate global paths and extract key waypoints, which then serve as temporary sub-goals for the DWA at each stage. This design mitigates the impact of dynamic obstacles on the overall path. While the GA exhibits strong global search capabilities, it may still converge to local optima, especially in large search spaces. Suboptimal parameter tuning can further lead to unnecessarily elongated paths.

Han et al. [29] proposed a multi-robot path planning method combining an improved A* algorithm with the DWA. Their approach introduces a distance cost concept into the heuristic function of the traditional A* algorithm, enabling adaptive adjustments based on the robot's current distance to the target. However, the improved A* algorithm requires additional dynamic adjustments and computations during path planning, which may lead to high computational and resource costs, especially in multi-robot systems.

Lin et al. [30] addressed the path planning problem for unmanned surface vehicles (USVs) in dynamic environments by proposing an enhanced Q-learning algorithm. Their method incorporates α- and ε-decay mechanisms to optimize the learning and exploration rates, respectively. The improved algorithm demonstrates strong performance in both single- and multi-USV scenarios, generating shorter paths, reducing computation time, and achieving fast convergence. It also exhibits high adaptability and efficiency in handling dynamic obstacles and multi-agent coordination. Despite its advantages, the algorithm is sensitive to parameter tuning. While the α-decay and ε-decay mechanisms improve convergence speed, their decay rates require careful adjustment. Improper settings may lead to premature convergence to suboptimal policies or insufficient exploration, ultimately degrading path quality.

Building on the strengths and limitations of existing approaches, this study presents a novel multi-robot path planning method that integrates the MA-DWAPSS algorithm. This

https://doi.org/10.3390/sym18030499

combines the global and local search advantages of the memetic algorithm, the real-time obstacle avoidance capability of the DWA, and the multi-robot collaborative obstacle-avoidance capability of parking–scheduling strategies. This enables multi-robot path planning to be efficient and safe in complex, dynamic environments. In addition, we use cubic Bézier curves [31] during the global path planning stage to fit the generated paths. This reduces the turning angles and sharp direction changes during the robot's movement, thus improving the smoothness and stability of its motion.

Table 1. Classification of path planning algorithms.

Types	Subtypes	Advantage	Disadvantage	Example
Traditional method	Global Path Planning	Exact Calculation, Global Optima	Poor Adaptability to Dynamic Environments	[11–14,32]
	Local Path Planning	Strong Dynamic Real-time Performance	Prone to Local Optima	[7,15–19]
Bio-inspired intelligence	—	Strong Adaptability, Efficiency, Robustness	Parameter Sensitivity, Local Optima Issue	[21–23]
Reinforcement learning	—	Environmental Adaptability, Self-adaptability	High Training Computational Cost, Prone to Suboptimal Strategies	[25–27]

3. Bilevel Multi-Robot Cooperative Path Planning Model

The bilevel model can be divided into two levels. The global (static) path planning level is performed before the robot moves, providing an optimal or near-optimal route from start to goal and reducing the need for time-consuming recalculations. The dynamic path planning level is performed to avoid path conflicts.

3.1. Problem Description

In multi-robot path planning, the goal is to ensure that robots can quickly and efficiently complete tasks in complex environments. To evaluate the quality of a path, we use three key metrics: distance to the target, distance to obstacles, and path smoothness. Each metric is designed to optimize the robot's performance in both dynamic and static environments, ensuring the path is efficient in avoiding obstacles and quickly reaching the target. The details are shown in Figure 1.

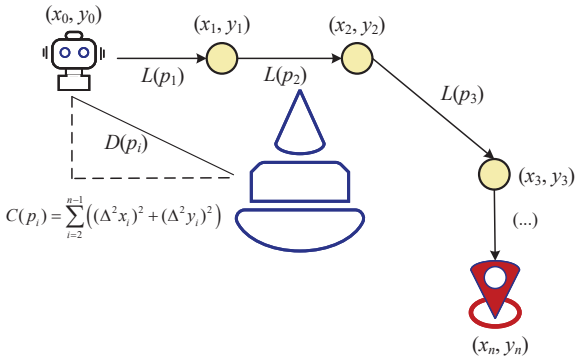


Figure 1. The modeling of robot path planning.

The distance to the target L is an intuitive and important metric in path planning. This metric measures the shortest distance from the robot's current position to the target point. In rescue scenarios, time is often the most valuable resource, so shortening the path means improved rescue efficiency. To calculate this distance, we use Euclidean distance, summing the distances between adjacent index points.

https://doi.org/10.3390/sym18030499

The distance to obstacles D is another key metric in path planning, related to the robot's safety. The robot needs to ensure a safe distance from obstacles to avoid collisions. We treat static obstacles, moving obstacles, and other robots as obstacles and calculate the distance to them using Euclidean distance. To enhance path safety, we define the minimum distance between the robot and obstacles ($D_{\min}$), which is the closest distance along the path. When planning the path, the robot should choose routes that are as far as possible from obstacles to avoid collisions.

Path smoothness C reflects the curvature between consecutive index points on the path. A smooth path reduces sharp turns, making the robot's movement more stable, minimizing wear on mechanical parts, and improving task precision. We represent index points with two-dimensional coordinates and use the second-order difference sum of squares to measure the path's smoothness. Smoothness is positively correlated with curvature: the smaller the smoothness, the smaller the curvature, and the smoother the path. Conversely, the larger the smoothness, the larger the curvature, and the more winding the path.

3.2. Process of Bilevel Model

In the path planning problem, the first step is to generate a feasible path that satisfies the basic constraints. Based on this, the path is optimized to improve its efficiency, safety, or other performance metrics. In this paper, the A* algorithm is used in the global planning layer to search for the initial path population $p = \{p_1, p_2, \cdots, p_n\}$, where each path is represented by a set of node coordinates $p_i = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$, providing an initial solution for subsequent path optimization. Euclidean distance is adopted as the heuristic function $h(n) = \sqrt{(x_n - x_g)^2 + (y_n - y_g)^2}$, where (x_n, y_n) is the coordinate of the current node n and (x_g, y_g) is the coordinate of the target node g, and the evaluation function is defined as $f(n) = g(n) + h(n)$, where $g(n)$ is the actual cost from the start node to the current node n.

Path quality is evaluated based on several factors, including path length, smoothness, and the distance between the path and obstacles. The fitness function quantifies these factors comprehensively. The fitness function is as follows:

$$\min \quad f(p_i) = w_1 \cdot L(p_i) + w_2 \cdot C(p_i) + w_3 \cdot D(p_i) \tag{1}$$

where $L(p_i)$ represents the total length of the path, which directly affects efficiency. A shorter path reduces time and resource consumption. We use Euclidean distance for calculation, as shown in Equation (2). $C(p_i)$ measures the smoothness of the path. Large turns or winding paths may lead to instability and increased energy consumption. We provide a detailed description of the smoothness calculation in Equation (3). $D(p_i)$ represents the distance between the path and obstacles, ensuring safety and avoiding collisions. We also use Euclidean distance for calculation. The weights w_1, w_2, w_3 in the fitness function reflect the influence of different factors on path selection and can be adjusted. In this paper, for the fire rescue scenario, the weights are set as $w_1 = 0.3$, $w_2 = 0.3$, $w_3 = 0.4$, where safety is prioritized over path length, and the path smoothness is relatively high to ensure the robot can perform tasks stably in complex environments and avoid collisions with dynamic obstacles.

$$L(p_i) = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{2}$$

Smoothness is an important metric for assessing the smoothness of a path. When planning paths for multiple robots, it is essential not only to consider the path's effectiveness and safety but also to ensure its smoothness. A smooth path helps robots ex-

https://doi.org/10.3390/sym18030499

ecute tasks more efficiently and steadily, reducing energy consumption and wear and improving coordination.

In path planning, smoothness quantifies the rate of change in the path. Specifically, given a path consisting of discrete points $\{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$, the smoothness can be measured by the second-order differences to assess the curvature of the path. The mathematical expression is as follows:

$$C(p_i) = \sum_{i=2}^{n-1} \left((\Delta^2 x_i)^2 + (\Delta^2 y_i)^2 \right) \tag{3}$$

where $\Delta^2 x_i$ and $\Delta^2 y_i$ represent the second-order differences in the x and y directions, respectively, indicating the rate of change in the change between adjacent points. The second-order differences are calculated as follows:

$$\Delta^2 x_i = x_{i+1} - 2x_i + x_{i-1} \tag{4}$$

$$\Delta^2 y_i = y_{i+1} - 2y_i + y_{i-1} \tag{5}$$

By summing the squares of all second-order differences, we obtain the overall smoothness value of the path. A smaller smoothness value indicates a smoother path, while a larger value suggests a more jagged path.

The memetic algorithm combines the global search ability of the genetic algorithm with the local search capabilities of simulated annealing (SA) . The SA utilizes local optimization to enhance search efficiency and accuracy. This combination helps avoid the local optimum problem in traditional genetic algorithms while improving convergence speed and accuracy. During the global path planning period, the improved memetic algorithm explores the entire path space using the genetic algorithm. It generates multiple path candidates via crossover and mutation, and then further optimizes each candidate using local search methods. The memetic algorithm demonstrates its superiority and effectively addresses the shortcomings of traditional genetic algorithms in complex dynamic environments or multi-objective optimization problems.

In the selection phase of the genetic algorithm, we use the tournament selection strategy, where individuals with higher fitness are selected from the population p. After selection, a crossover point k is randomly chosen, and the paths p_a and p_b are crossed over to generate a new path p_{new}. In the mutation phase, a Gaussian perturbation is applied to the path p_i, generating new coordinates (x'_j, y'_j) according to the following equation:

$$(x'_j, y'_j) = (x_j, y_j) + N(0, \sigma) \tag{6}$$

where σ is the mutation strength, which controls the degree of random perturbation on the path. A larger σ value helps escape local optima in complex environments but may affect convergence stability. On the other hand, a smaller σ improves path accuracy but reduces the algorithm's search speed. Therefore, finding an appropriate σ value is crucial. Based on experiments, the σ value is set to 0.03 of the total path length in this study.

Due to the possibility of genetic algorithms getting stuck in local optima, after the global search is completed, we apply local search and annealing mechanisms to perform a local perturbation on the generated path p_i, creating a new path p'_i. The perturbation equation is as follows:

$$p'_i = \text{swap}(p_i, x_i, x_j) \tag{7}$$

where two randomly selected coordinate nodes x_i and x_j are swapped to perturb the path, generating a new path p'_i.

https://doi.org/10.3390/sym18030499

After obtaining the new path, we calculate the change in fitness Δf to decide whether to accept the perturbed path. The fitness change equation is

$$\Delta f = f(p_i') - f(p_i) \tag{8}$$

where $f(p_i')$ is the fitness of the new path after perturbation, and $f(p_i)$ is the fitness of the original path before perturbation. If $\Delta f < 0$, we accept the new path. Otherwise, we accept the new path based on the designed acceptance probability p_{accept}, which is given by

$$p_{\text{accept}} = \exp(-\frac{\Delta f}{T}) \tag{9}$$

where T is the current iteration temperature. We initialize the temperature T_0 and update the temperature for each iteration using the following equation:

$$T_{k+1} = \alpha \cdot T_k \tag{10}$$

This completes the annealing process, resulting in the final new path P'.

To ensure a smoother generated path and avoid dangerous sharp turns, we fit the generated path using a cubic Bézier curve based on the population P'. The equation is as follows:

$$B_i(t) = (1-t)^3 x_i' + 3(1-t)^2 t x_{i+1}' + 3(1-t) t^3 x_{i+2}' + t^3 x_{i+3}' \tag{11}$$

where the parameter t ranges from $[0, 1]$, and x_i', x_{i+1}', x_{i+2}', x_{i+3}' represent four key control points selected from the discrete coordinates of the new path, after applying the Ramer–Douglas–Peucker algorithm to remove redundant points. These control points define the shape of the Bézier curve. When $t = 0$, the curve passes through x_i', and when $t = 1$, it passes through x_{i+3}'. The fitting process results in a smooth path, as shown in Figure 2.

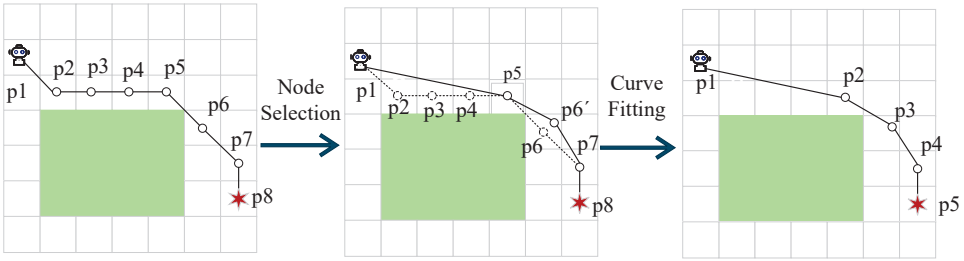


Figure 2. Fitting process.

4. MA-DWAPSS Algorithm

4.1. MA-DWAPSS Algorithm Framework

The MA-DWAPSS algorithm integrates the improved memetic algorithm with the DWAPSS algorithm. The memetic algorithm is a heuristic optimization method that efficiently searches the solution space and finds the global optimal solution. The DWAPSS algorithm focuses on real-time dynamic obstacle avoidance and path tracking, adjusting the path based on factors like the target, obstacles, and the robot's own speed. The MA-DWAPSS algorithm combines the advantages of both, achieving a good balance in path planning by considering both the global optimal solution and real-time environmental changes.

In this algorithm, the improved memetic algorithm first performs global optimization on the overall path to determine a rough optimal path. Then, the dynamic window algorithm adjusts the robot's speed, heading, and distance to the target based on real-time

https://doi.org/10.3390/sym18030499

sensory data, ensuring effective obstacle avoidance and quick target reaching in a complex dynamic environment; the pseudocode is shown in Algorithm 1.

Algorithm 1: The process of the MA-DWAPSS lgorithm

Input: P, $v_{\max}$, $v_{\min}$, $\omega_{\max}$, $\omega_{\min}$, $itmax$, $iter$
Output: the final path P'
Initialize P, $v_{\max}$, $v_{\min}$, $\omega_{\max}$, $\omega_{\min}$, $itmax$, $iter = 0$;
use A* algorithm to initialize paths for each individual in population P;
while $iter < itmax$ **do**
    /* use GA and SA to update P */
    if $\Delta f < 0$ **then** /* Refer to Equation (8) */
        $P \to P'$
    else
        $P \to P'$ or P /* P' is selected with p_{accept}, refer to Equation (9) */
    $iter = iter + 1$;
Optimize the path P' using a cubic Bézier curve /* refer to Equation (11) */
set dynamic window constraints V_d /* refer to Equation (12) */
set trajectory evaluation function $G(v, \omega)$ /* refer to Equation (13) */
$(v, \omega) \to (v^*, \omega^*)$ /* $G(v^*, \omega^*)$ is the optimal trajectory evaluation value. Refer to Equation (14) */
if *reach the target point* **then**
    Output the final path P';
else
    Return to step 9 ;

In fire rescue scenarios, the environment is usually very complex and changes rapidly. The MA-DWAPSS algorithm can accurately identify rescue targets in such dynamic environments. It not only finds a reasonable path through global optimization but also dynamically evaluates the robot's motion plan based on environmental changes, choosing the optimal trajectory. This allows the robot to handle complex obstacles and rapidly changing environmental conditions in a fire scenario, effectively avoiding dynamic obstacles and ensuring the smooth completion of rescue tasks.

According to the above-described optimization problem, the MA-DWAPSS algorithm proposed in this paper is designed to enhance the collaboration efficiency and accuracy of path planning in multi-robot systems. This study presents an innovative improvement to both the memetic algorithm and the Dynamic Window Approach, integrating them into an efficient two-stage path planning framework that comprises global planning and local planning, as illustrated in Figure 3. The pseudocode of the MA-DWAPSS algorithm is given in Algorithm 1.

The primary objective of the global planning stage is to determine an optimal global path for the robot group, from the starting point to the target destination. This stage considers the overall environmental layout, aiming to provide a general directional guide for the robots, enabling them to avoid major obstacles and prevent the generation of excessively winding or lengthy paths. The improved memetic algorithm is employed in this stage. As a hybrid metaheuristic method, the memetic algorithm combines genetic algorithms with local search strategies, simulating natural selection and evolutionary processes to efficiently identify the global optimal solution. Particularly in complex multi-robot coordination tasks, the enhanced memetic algorithm achieves rapid convergence

https://doi.org/10.3390/sym18030499

to the global optimal path by incorporating refined local search mechanisms, thereby mitigating the risk of getting trapped in local optima during the global search.

The local planning stage focuses on real-time path adjustments in response to dynamic environmental changes based on the coarse path generated during the global planning phase. This ensures that robots can effectively avoid sudden obstacles and other robots, thereby preventing collisions. Local planning emphasizes precise, short-term trajectory modifications, especially when the robot is in proximity to the target or needs to respond to real-time environmental variations. This stage utilizes the DWAPSS algorithm. DWAPSS computes the feasible velocity range and selects the most appropriate velocity within this range to guide the robot in obstacle avoidance, ensuring smooth and efficient local path adjustments. This enhancement enables the robot to dynamically evade unexpected obstacles and make agile decisions within confined spaces, thereby ensuring both safety and timeliness in path execution.

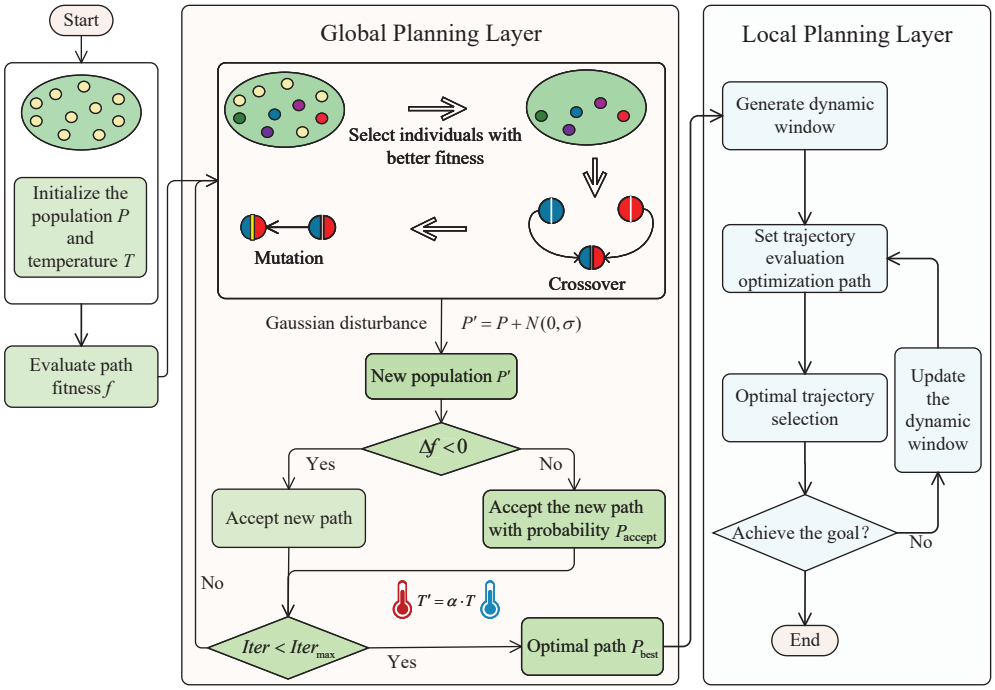


Figure 3. The framework of the MA-DWAPSS algorithm.

4.2. Dynamic Window Approach with Parking Scheduling Strategy

The dynamic window approach is a widely used local planning algorithm for mobile robots, especially in dynamic environments. It generates feasible control inputs by considering both the robot's motion model and environmental obstacles, enabling obstacle avoidance and goal-directed tasks. This algorithm is highly efficient in terms of real-time performance and computational speed, making it widely applied in multi-robot collaborative path planning, such as in fire rescue operations.

DWAPSS aims to improve robot path planning and obstacle avoidance functions. Assuming the same robot speed, the robot's priority is determined by its path distance to the target during global planning. When robots encounter each other, the one with the shorter distance to the target is considered the higher-priority robot and is allowed to pass first, while the lower-priority robot must park and wait. This is particularly useful in human–robot collaboration or multi-robot systems, enhancing the robots' ability to avoid collisions in dynamic environments. The parking scheduling strategy monitors the relative position and motion status between robots to ensure collision avoidance during collaboration. This strategy takes into account not only the robot's current motion state

https://doi.org/10.3390/sym18030499

but also information such as speed and position, ensuring effective collision avoidance and coordination. Specifically, when one robot approaches another, the parking scheduling strategy adjusts the robot's movement to prevent unnecessary collisions. When combined with the DWA, the dynamic window selection considers not only static obstacles in the environment but also other robots and potential dynamic obstacles. The introduction of parking scheduling strategies makes robot path planning more intelligent and flexible in complex environments, reducing the response time for collision avoidance during movement.

The steps of the DWAPSS algorithm are as follows:

(1) *Initialize parameters*

- Define the robot's motion model parameters: maximum linear velocity $v_{\max}$, minimum linear velocity $v_{\min}$, maximum angular velocity $w_{\max}$, and minimum angular velocity $w_{\min}$;
- Define the weight coefficients for the trajectory evaluation function: α, β and γ;
- Set the obstacle avoidance parameters: obstacle detection distance d_{obs}, minimum safety distance $d_{\min}$, obstacle tolerance ϵ_{obs}, obstacle velocity v_{obs}.

(2) *Dynamic Window Generation*

Based on the robot's current state (linear velocity v and angular velocity w), calculate the current dynamic window V_d. The range of this dynamic window is constrained as follows:

$$V_d = \{(v, w) \mid v \in [v_{\min}, v_{\max}],\ w \in [w_{\min}, w_{\max}]\} \tag{12}$$

The range of linear velocity is represented by $[v_{\min}, v_{\max}]$, and the range of angular velocity is represented by $[w_{\min}, w_{\max}]$.

(3) *Trajectory Generation and Evaluation*

For each velocity pair (v, w), a trajectory is generated. The evaluation function $G(v, w)$ is then calculated for each trajectory. The evaluation function consists of three parts:

- heading(v, w): Consistency between the trajectory direction and the target direction.;
- dist(v, w): The minimum distance between the trajectory and obstacles (the farther the trajectory is from obstacles, the higher the evaluation);
- velocity(v, w): The speed of the trajectory.

The trajectory evaluation function is expressed as

$$G(v, w) = \alpha \cdot \text{heading}(v, w) + \beta \cdot \text{dist}(v, w) + \gamma \cdot \text{velocity}(v, w) \tag{13}$$

where α is the heading weight, which controls the deviation from the target direction. To ensure the robot reaches the target accurately, α is set to 0.5. β is the distance weight, which controls the distance from the target or obstacles to avoid collisions, and it is set to 6. γ is the velocity weight, used to control the speed of movement, set to 2.

(4) *Optimal Trajectory Selection*

For all possible velocity pairs (v, w), the evaluation function $G(v, w)$ is calculated. The trajectory with the maximum evaluation function value is selected as the optimal trajectory:

$$(v^*, w^*) = \arg \max_{(v,w) \in V_d} G(v, w) \tag{14}$$

where (v^*, w^*) represents the optimal trajectory considering the target direction, obstacle avoidance, and speed at the current time. The optimal velocity v^* and angular velocity w^* are then converted into control commands to drive the robot.

(5) *Update Robot State*

https://doi.org/10.3390/sym18030499

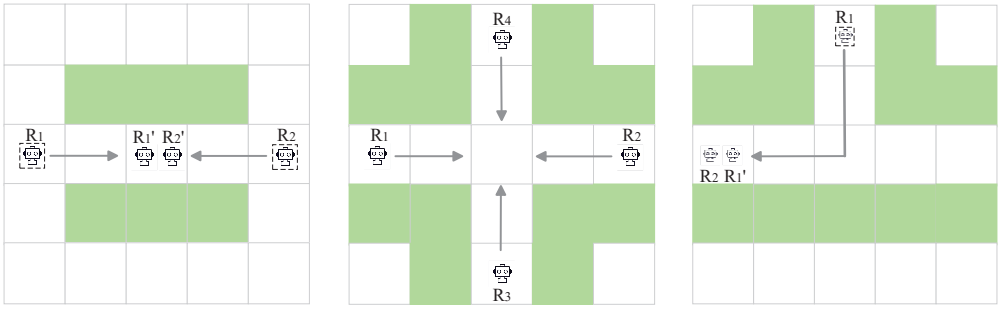
After the robot executes the control commands, its current position, velocity, and angular velocity are updated. Steps (2) to (4) are repeated until the target is reached or the exit conditions are met.

4.3. Parking Scheduling Strategy

In multi-robot collaborative path planning, resolving path conflicts in dynamic environments and ensuring effective obstacle avoidance between robots is a major challenge. Robots not only need to complete their tasks but must also adjust their paths in real-time to avoid collisions with other robots or obstacles. Common types of path conflicts include robot collision conflicts, node conflicts, and deadlock conflicts.

A parking scheduling strategy can effectively address these conflicts by calculating the distance of the robots from obstacles and target points to determine the execution order, enabling efficient and safe collaboration. This strategy ensures that each robot, during task execution, decides when to perform tasks, how to avoid other robots, and how to coordinate movement efficiently based on its priority, minimizing conflicts and optimizing system performance. Figure 4 shows three types of path conflicts.

1. Collision conflict, shown in Figure 4a, usually occurs in narrow passages or intersections. Robots with higher priority pass first, while robots with lower priority must pause or adjust their paths to avoid collisions.
2. Node conflict, shown in Figure 4b, occurs when multiple robots plan to reach the same or adjacent nodes. The solution is to give priority to higher-priority robots, while lower-priority robots must wait.
3. In the deadlock conflict shown in Figure 4c, the system avoids robot blockage through predefined, reasonable path planning. If deadlock occurs, the system initiates a dynamic replanning mechanism, assigning new paths to lower-priority robots while ensuring that the tasks of higher-priority robots remain unaffected, thus avoiding long waiting times or deadlock.



(**a**) Collision conflict (**b**) Node conflict (**c**) Deadlock conflict

Figure 4. Three types of path conflicts.

In dynamic environments, robots often face complex changes in obstacles, especially when interacting with dynamic obstacles. To ensure both the safety of the robot and the completion of tasks, parking scheduling strategies become crucial. In response to this challenge, we propose an improved parking scheduling strategy based on DWA, as shown in Figure 5. Algorithm 2 outlines the parking scheduling strategy, providing a clearer description of its implementation.

https://doi.org/10.3390/sym18030499

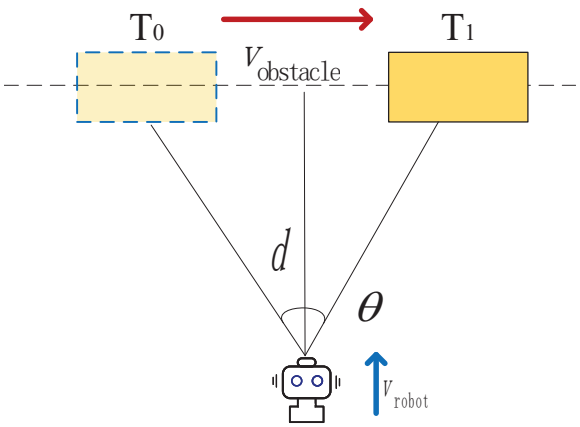


Figure 5. Robot parking scheduling strategy.

Algorithm 2: Parking scheduling strategy

```
   /* Step 1: Detect dynamic obstacles                                    */
1  while true do
2      detected ← Detect_obstacles(robot_pos, obs);
3  if detected is not empty then
       /* Step 2: Identify nearest obstacle                               */
4      nearest ← Nearest_obstacle(detected);
       /* Step 3: Compute distance threshold for parking                  */
5      d_thresh ← Compute_dist_thresh(robot_vel, nearest_vel);
       /* Step 4: Compute distance and angle of nearest obstacle          */
6      dist ← Compute_dist(robot_pos, nearest_pos);
7      angle_thresh ← Compute_angle_thresh(dist, max_dist, base_angle, spread);
       /* Step 5: Evaluate parking condition                              */
8      if dist < d_thresh and nearest_angle < angle_thresh then
9          action ←"Park" /* Initiate parking                             */
10     else
11         action ←"Move" /* Continue movement                            */
12 else
13     action ←"Move" /* No obstacles, continue                          */
       /* Step 6: Perform action                                          */
14 if action = "Park" then
15     Execute_parking(robot)/* Execute parking                          */
16     Set_robot_state("Parked");
17     Output "Parking...";
18 else
19     Set_robot_state("Moving") /* Continue moving                       */
20     Output "Moving...";
   /* Step 7: Stop when parking is complete                               */
21 if robot_state = "Parked" then
22     break /* End algorithm after parking                              */
```

This strategy combines real-time monitoring of the environment and adaptive adjustments based on the robot's current operating state. Specifically, when a parking demand occurs (for multiple robots, the order is determined by dist(v, w), the robot first evaluates

https://doi.org/10.3390/sym18030499

the distribution and movement trajectories of dynamic obstacles in the current environment. In cases where the obstacle density is high and abrupt stops could be dangerous, the robot will select a safe parking position to temporarily stop and, ensuring safety, reasonably choose the right time to park.

The core part of this strategy is the introduction of a "Park–Observe State," where the robot temporarily stops moving. It uses sensors and the environmental perception system to collect real-time data on changes in the surrounding environment and the movement trends of obstacles. Based on this information, the robot can quickly decide whether to continue moving or wait for the dynamic obstacles to pass. This strategy effectively prevents path deviations caused by abrupt stops or improper parking, improving both path planning accuracy and overall operational efficiency.

The improved algorithm constrains the robot's distance check by setting distance and angle thresholds. To make the robot more scientifically assess distance and angle thresholds in various environments and respond promptly to avoid collisions, we propose a dynamic adjustment strategy. We introduce dynamically adjusted distance and angle thresholds to enhance the robot's responsiveness and adaptability.

First, for the distance threshold, we propose dynamically adjusting it based on the speed of the obstacle, the robot's speed, and the complexity of the environment. When the obstacle moves faster, the robot needs to react earlier. Thus, a proportional relationship is set so that the faster the obstacle moves, the smaller the distance threshold. On the other hand, the faster the robot moves, the larger the distance threshold, providing the robot with enough time to react. The specific equation is as follows:

$$d = k \times \frac{v_{\text{robot}}}{v_{\text{obstacle}}} \tag{15}$$

where k is a constant set to 2, and v_{robot} and v_{obstacle} represent the speeds of the robot and the obstacle, respectively.

Secondly, adjusting the angle threshold is equally important. Traditional fixed angle thresholds may be too general in some cases, causing the robot to struggle with obstacles in different relative positions. We propose dynamically adjusting the angle threshold based on the distance and relative direction of the obstacle. When the obstacle is closer, the angle threshold should be smaller to increase sensitivity. When the obstacle is farther away, the angle threshold can be increased to improve the robot's flexibility in responding. The dynamic adjustment equation is as follows:

$$\theta = \theta_0 \times (1 + \sigma \times \frac{d}{d_{\text{max}}}) \tag{16}$$

where θ_0 is the minimum angle threshold (set to 30°), σ is 0.5 to adjust the range of angle threshold changes, d is the current distance between the obstacle and the robot, and d_{max} is the maximum distance.

5. Experimental Study

To better address the issues presented in this paper, we designed and conducted simulation experiments with multiple robots in dynamic unknown environments to test the performance of the MA-DWAPSS algorithm. In these experiments, we used grid maps of different sizes to evaluate the algorithm's performance in environments of varying scales. During the experiments, we assessed the algorithm's performance based on three key factors: optimal path length, computation time, and path smoothness. This allowed us to gain a comprehensive understanding of the advantages and limitations of the MA-DWAPSS algorithm in dynamic unknown environments.

https://doi.org/10.3390/sym18030499

5.1. Grid Map and Parameter Setting

To validate the performance of the proposed algorithm, three types of algorithms—intelligent optimization algorithms, traditional algorithms, and reinforcement learning algorithms—are selected for comparison. The chosen algorithms are the improved genetic algorithm with the improved dynamic window approach (IGA-IDWA) [28], the A* algorithm with the dynamic window approach (A*-DWA) [29], and the enhanced *Q*-learning (E-Q-Learning) algorithm [30].

5.1.1. Grid Map Setting

Three test scenarios were set up in the experiments: grid maps of sizes 20×20, 30×30, and 50×50. To make the setup more realistic, each grid cell was set to 1 m × 1 m. Figure 6 shows the static obstacle grids at the initial state for each scenario. The blue triangle represents the starting position of the rescue robot in each scenario, while the red circle indicates the target rescue position.

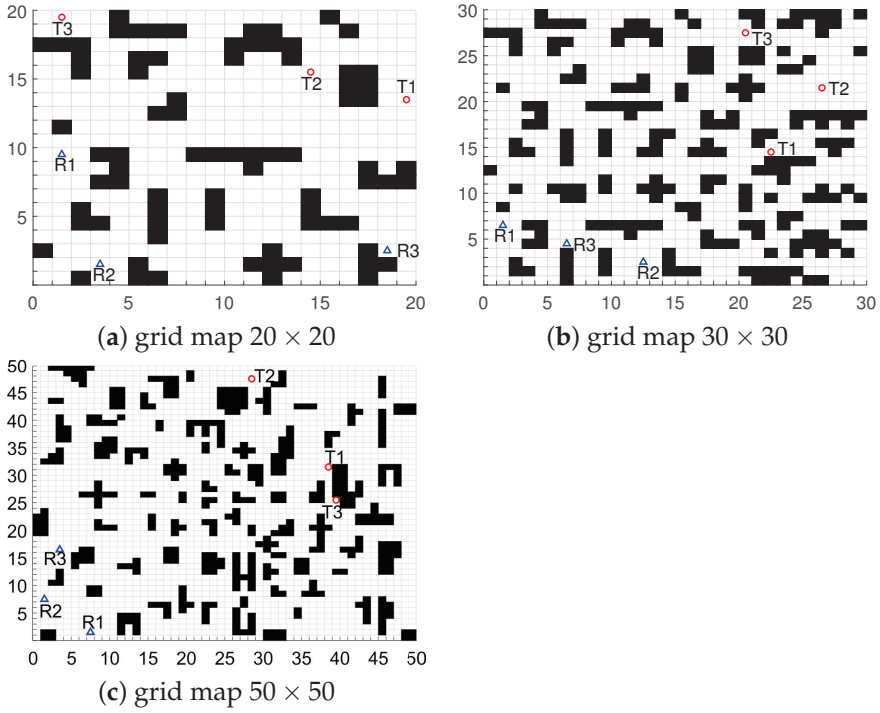


Figure 6. Initial grid models of different sizes.

5.1.2. Parameter Setting

The relevant parameters are set as follows:

- Improved memetic algorithm parameters:
  - Simulated annealing temperature: $T = 200$
  - Population size: $NP = 100$
  - Maximum evolution generations: $max_gen = 100$
  - Crossover probability: $p_c = 0.8$
  - Mutation probability: $p_m = 0.5$
- DWAPSS parameters:
  - Maximum linear velocity: $v_{\max} = 1$
  - Minimum linear velocity: $v_{\min} = -0.5$
  - Maximum angular velocity: $w_{\max} = 90.0 \times \pi/180.0$
  - Minimum angular velocity: $w_{\min} = 10.0 \times \pi/180.0$
  - Maximum acceleration: $a = 0.3$

https://doi.org/10.3390/sym18030499

- Linear velocity resolution: v_resolution $= 0.02$
- Angular velocity resolution: yawrate_resolution $= 0.3 \times \pi/180.0$
- Time step: $dt = 0.1$
- Heading weight: $\alpha = 0.5$
- Distance weight: $\beta = 6$
- Velocity weight: $\gamma = 2$.

5.2. Performance Comparison and Analysis

In the experimental analysis section, we compare global path planning experiments with dynamic path planning experiments. We evaluate the algorithms based on several metrics, including time, path length, path smoothness, and step size. Finally, in a dynamic unknown environment, we validate the effectiveness of the proposed MA-DWAPSS algorithm by comparing the curves of linear and angular velocities recorded on grid maps of sizes 20×20, 30×30, and 50×50. These comparisons provide a comprehensive assessment of the algorithm's performance in different environments and conditions.

5.2.1. Global Path Planning

In path planning on a grid map, global (static) path planning is performed before the robot moves, providing an optimal or near-optimal route from start to goal and reducing the need for time-consuming recalculations. During this phase, metrics of four algorithms are recorded to analyze performance and demonstrate the superiority of the proposed MA-DWAPSS algorithm. Figure 7 shows planning results for the four algorithms across three test scenarios. Table 2 indicates that MA-DWAPSS outperforms IGA-IDWA and A*-DWA in time, path length, and other metrics in every scenario. E-Q-Learning achieves the best planning time, but its path length and smoothness are inferior to the proposed algorithm.



(**a**) MA-DWAPSS (**b**) IGA-IDWA (**c**) A*-DWA (**d**) E-Q-learning

(**e**) MA-DWAPSS (**f**) IGA-IDWA (**g**) A*-DWA (**h**) E-Q-learning

(**i**) MA-DWAPSS (**j**) IGA-IDWA (**k**) A*-DWA (**l**) E-Q-learning

Figure 7. Global path planning results (with grid map sizes of 20×20, 30×30, 50×50 from top to bottom.

To clearly compare the advantages and disadvantages of the four algorithms across different metrics, we visualized the data from Table 2. All experiments have been run 20 times. The current result is the average value of 20 independent runs. The processed results are shown in Figures 8–10.

https://doi.org/10.3390/sym18030499

Table 2. Results of algorithm metrics at different sizes.

Grid Sizes	Algorithm	Times (s)	Original Path Lengths (m)	Smoothed Path Lengths (m)
20 × 20	MA-DWAPSS	26.39	66.52	63.96
	IGA-IDWA	35.13	68.75	65.90
	A*-DWA	31.81	64.60	62.97
	E-Q-Learning	22.21	70.75	63.93
30 × 30	MA-DWAPSS	36.64	78.27	75.57
	IGA-IDWA	38.78	79.47	76.17
	A*-DWA	40.86	77.91	75.58
	E-Q-Learning	23.27	82.53	75.69
50 × 50	MA-DWAPSS	45.58	134.76	129.14
	IGA-IDWA	60.75	141.01	133.76
	A*-DWA	63.59	134.76	131.40
	E-Q-Learning	32.14	143.14	131.23

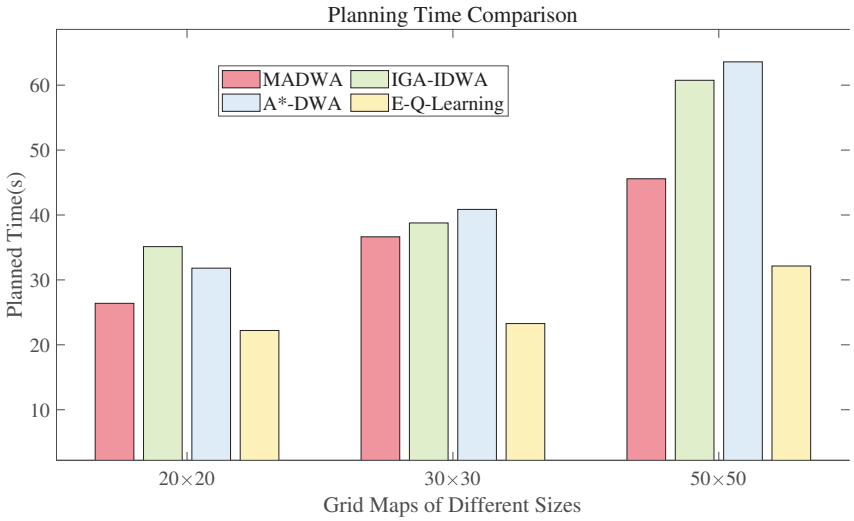


Figure 8. Comparison of planning time of the algorithm on grids of different sizes.

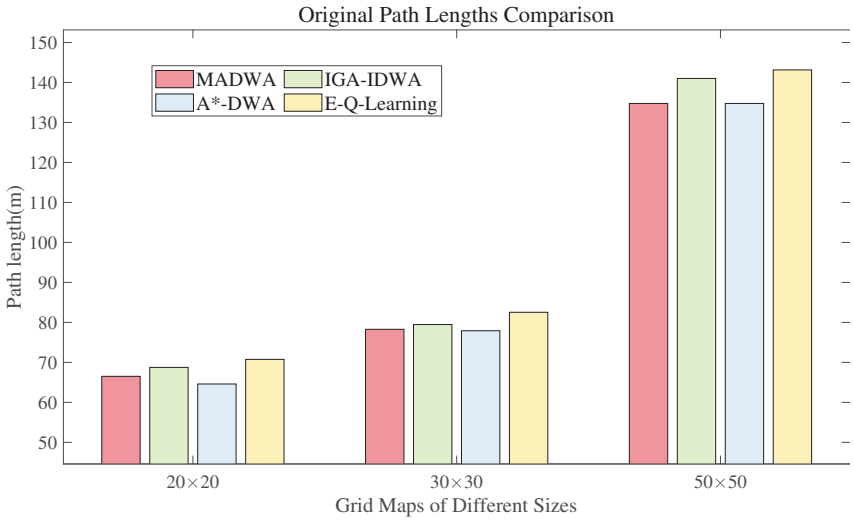


Figure 9. Comparison of the original path lengths of the algorithms on grid maps of different sizes.

https://doi.org/10.3390/sym18030499

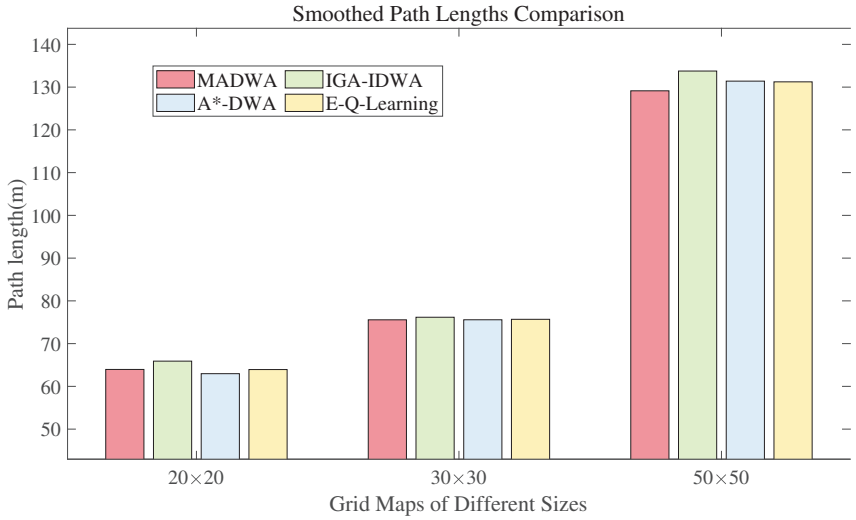


Figure 10. Comparison of the smoothed path lengths of the algorithms on grid maps of different sizes.

5.2.2. Dynamic Path Planning

In dynamic path planning on grid maps, the robot continuously adjusts its path in response to environmental changes. Unlike global path planning, dynamic path planning, also known as reactive path planning, allows the robot to adapt in real-time to obstacles or environmental changes. This is crucial for navigation in environments where unpredictable changes may occur, such as moving obstacles or sudden terrain changes. During this phase, the robot frequently recalculates its path to avoid collisions and ensure smooth navigation to the target. The process tracks relevant metrics for the four algorithms, focusing on their ability to adapt to dynamic obstacles and maintain efficient paths. To validate the real-time obstacle avoidance and optimal pathfinding capability of the proposed algorithm in unknown environments, we simulated a fire rescue scenario with a dynamic environment. Figure 11 shows the dynamic path planning results of the four algorithms in this dynamic environment, where the yellow blocks represent moving obstacles, and the gray blocks represent temporary obstacles (which do not appear in global path planning).

To further demonstrate the feasibility of the proposed algorithm, we recorded metrics such as path length, runtime, smoothness, and step size for dynamic path planning of the four algorithms in environments of different scales in Table 3, making it easier to compare their performance.

Table 3. Results of algorithm metrics at different sizes.

Grid Sizes	Algorithm	Times (s)	Steps	Path Lengths (m)	Smoothness (m^4)	Average Step Times (s)
20 × 20	MA-DWAPSS	**497.279**	**330.5**	62.235	**0.00160**	1.1593
	IGA-IDWA	530.595	345.15	65.409	0.00274	1.1896
	A*-DWA	504.662	446	62.490	0.00180	1.1214
	E-Q-Learning	509.154	449.05	**61.678**	0.00198	1.1550
30 × 30	MA-DWAPSS	**594.346**	**322.9**	**75.025**	**0.00240**	1.7889
	IGA-IDWA	620.617	350.85	78.275	0.00318	1.7783
	A*-DWA	631.113	362	76.300	0.00251	1.7553
	E-Q-Learning	1194.938	528.65	75.985	0.00251	2.2608
50 × 50	MA-DWAPSS	**1688.516**	571.3	**130.450**	**0.00244**	2.9567
	IGA-IDWA	1888.527	**564.05**	132.473	0.00264	3.2037
	A*-DWA	1879.056	557	134.830	0.00264	3.3825
	E-Q-Learning	1898.828	575.6	133.001	0.00261	3.3074

https://doi.org/10.3390/sym18030499



(**a**) MA-DWAPSS (**b**) IGA-IDWA (**c**) A*-DWA (**d**) E-Q-learning

(**e**) MA-DWAPSS (**f**) IGA-IDWA (**g**) A*-DWA (**h**) E-Q-learning

(**i**) MA-DWAPSS (**j**) IGA-IDWA (**k**) A*-DWA (**l**) E-Q-learning

Figure 11. Dynamic path planning results (with grid map sizes of 20 × 20, 30 × 30, 50 × 50 from top to bottom).

Figures 12–14 show the visualizations of time, path length, and smoothness for the four algorithms with grid sizes of 20 × 20, 30 × 30, and 50 × 50, as presented in Table 3. From the figures, it is clear that the proposed algorithm demonstrates significant improvements in time, path length, and smoothness compared to the other three algorithms.

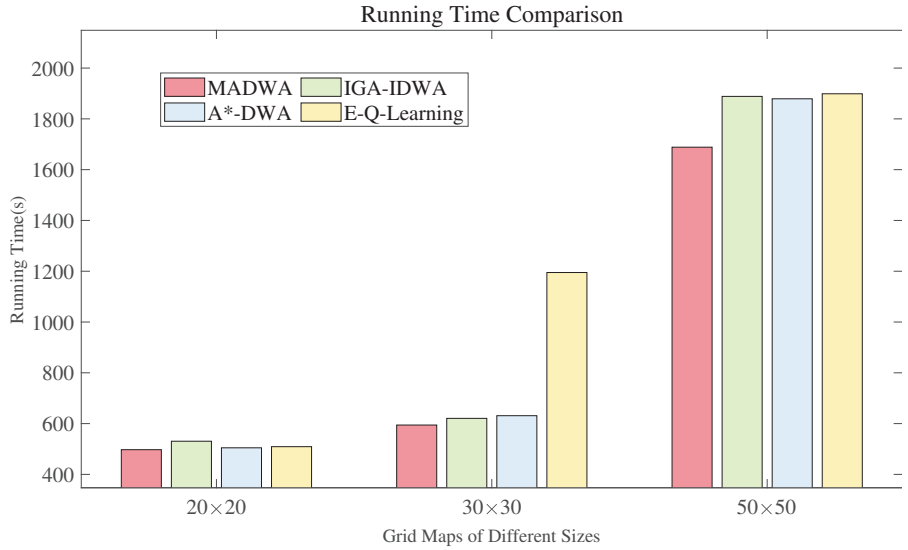


Figure 12. Comparison of Running Times of the Algorithm on Grids of Different Sizes.

Figure 15 shows the linear velocity and angular velocity curves of each robot using the MA-DWAPSS algorithm in a dynamic unknown environment. Taking the 20 × 20 grid as an example, for robot 3, the linear velocity drops to nearly 0 at step sizes 160 and 310. This indicates that the robot detected the moving unknown obstacles in time during its movement and took appropriate actions to successfully avoid collisions. This demonstrates the effectiveness of the parking scheduling strategy.

https://doi.org/10.3390/sym18030499

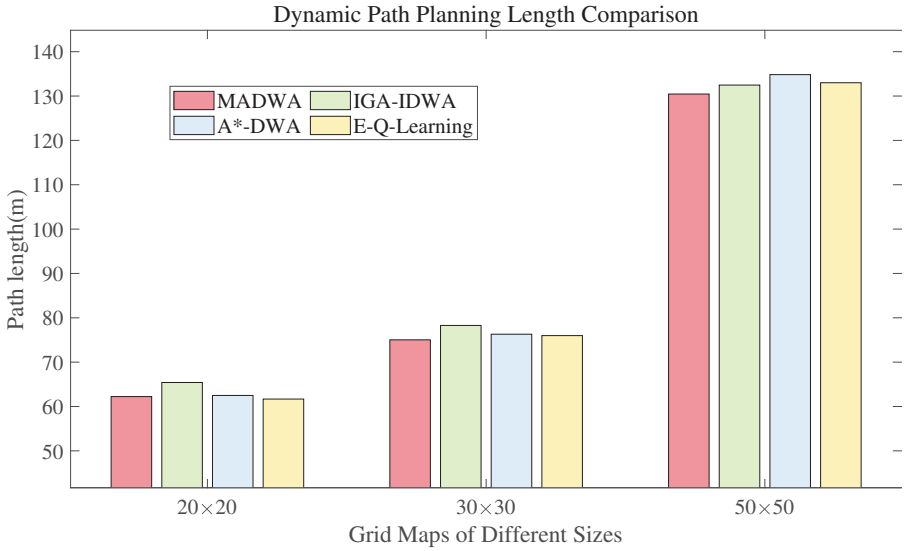


Figure 13. Comparison of dynamic path planning of the algorithm on grids of different sizes.

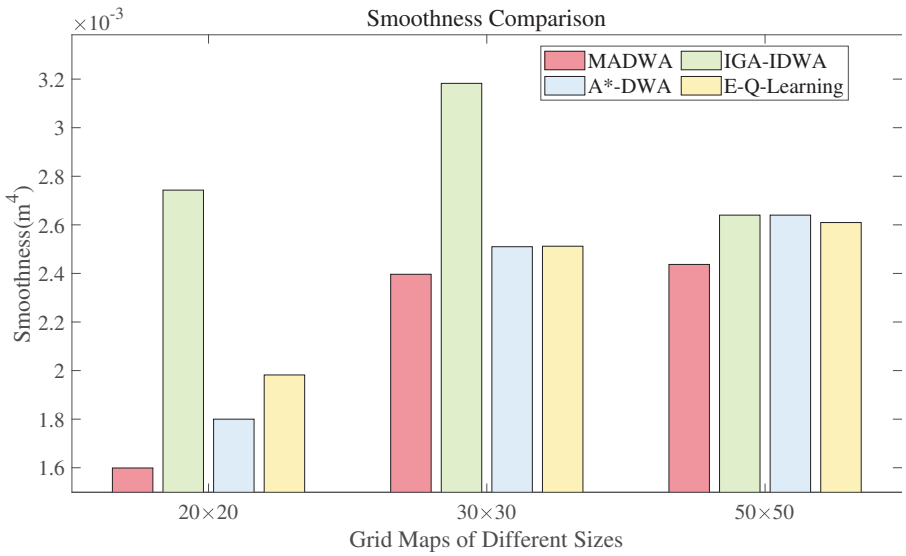
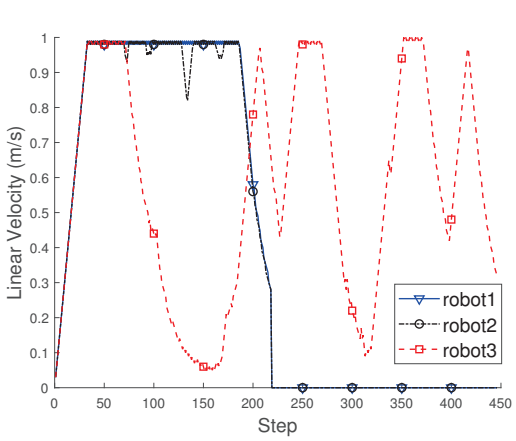
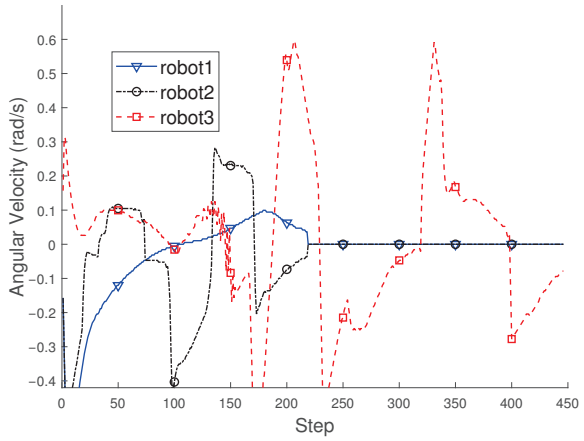


Figure 14. Comparison of smoothness of the algorithm on grids of different sizes.



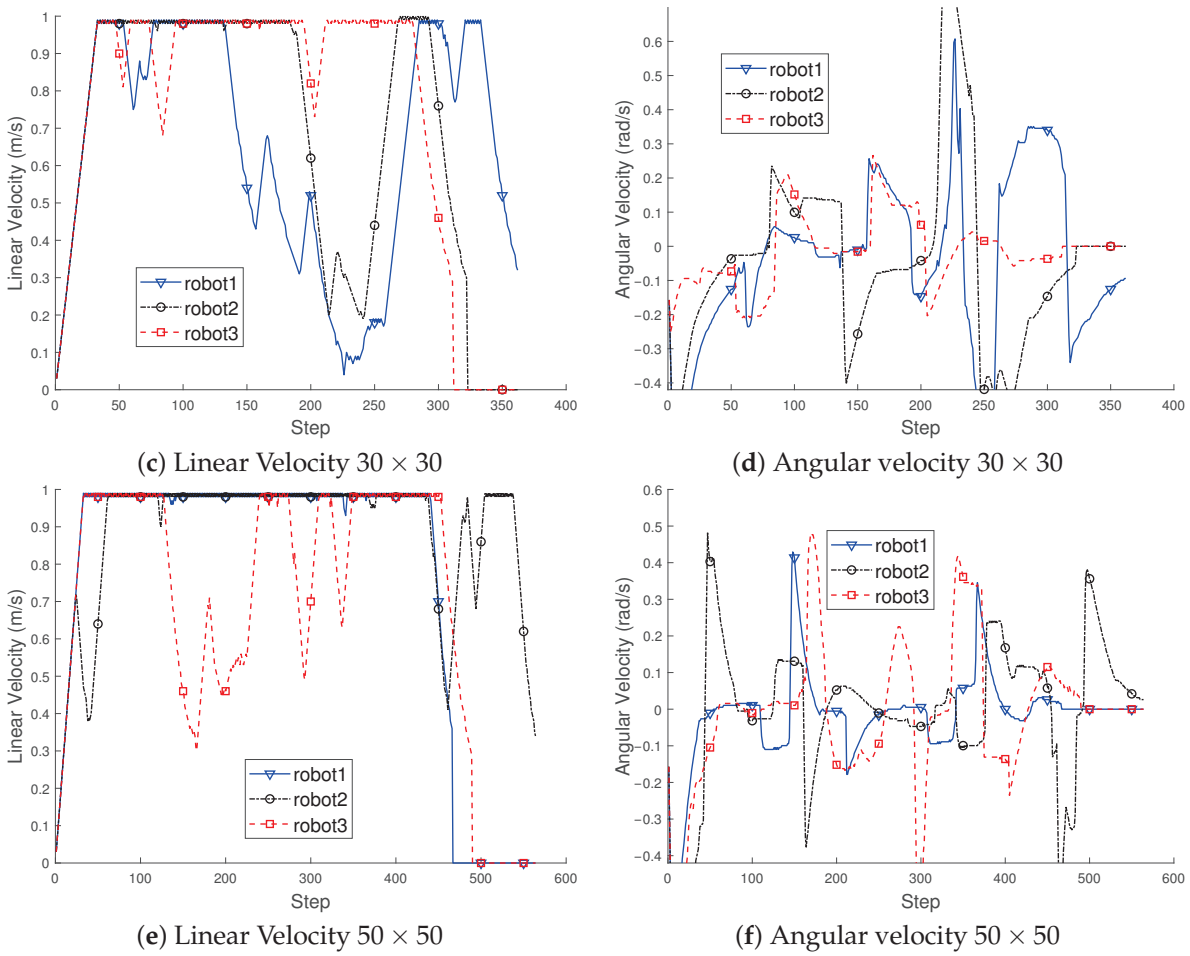
(**a**) Linear Velocity 20 × 20



(**b**) Angular velocity 20 × 20

Figure 15. *Cont.*

https://doi.org/10.3390/sym18030499



(**c**) Linear Velocity 30 × 30 (**d**) Angular velocity 30 × 30

(**e**) Linear Velocity 50 × 50 (**f**) Angular velocity 50 × 50

Figure 15. The linear velocity and angular velocity of MA-DWAPSS on grids of different sizes.

5.3. Ablation Experiment

To better understand the contribution of each module and evaluate the system's performance, we designed two ablation experiments. By sequentially removing certain components, we aim to assess their impact on the overall performance.

Experiment 1: Comparison of MA-DWAPSS and DWAPSS

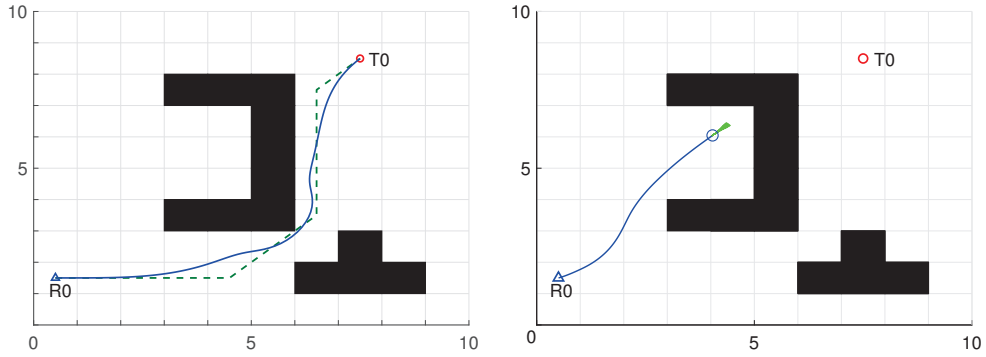
In Experiment 1, we compared two algorithms: MA-DWAPSS and DWAPSS. Specifically, we removed the MA's guidance for global routing, resulting in the DWAPSS algorithm. This experiment aims to verify the importance of the MA in global route planning. The MA optimizes global information, improving the efficiency and accuracy of overall path planning.

We designed a U-shaped obstacle to observe whether the robot could successfully avoid the obstacle. As shown in Figure 16, the comparison between the two algorithms illustrates the difference in performance. In Figure 16a, with the MA guiding the global planning, the robot effectively avoids the obstacle. In contrast, Figure 16b shows the DWAPSS algorithm without MA guidance, where the robot gets stuck in a local optimum when encountering the U-shaped obstacle and cannot find the optimal path. This clearly demonstrates the critical role of the MA algorithm in robot path optimization.

Experiment 2: Comparison of MA-DWAPSS and MA-DWA

In Experiment 2, we compared the MA-DWAPSS and MA-DWA algorithms to evaluate the impact of the parking scheduling strategy on robot motion and obstacle avoidance in dynamic environments. The MA-DWA algorithm does not consider parking scheduling, while the MA-DWAPSS algorithm incorporates this strategy.

https://doi.org/10.3390/sym18030499

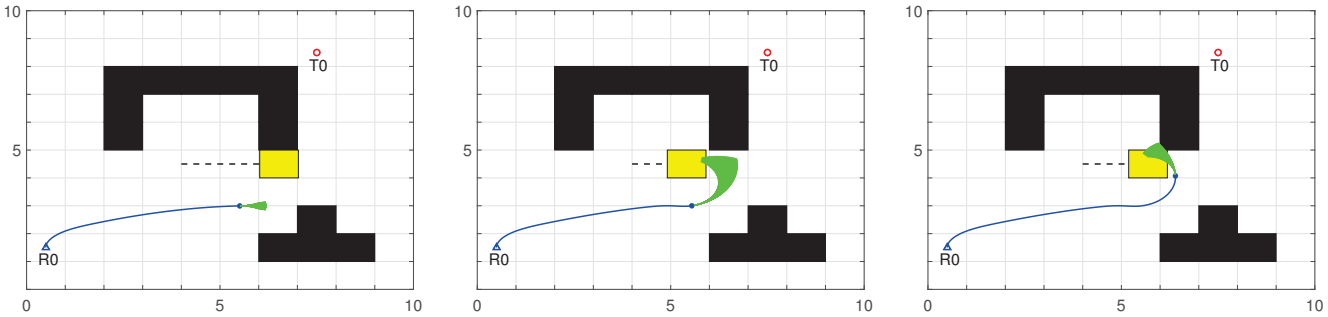


(**a**) MA-DWAPSS (**b**) DWAPSS

Figure 16. Comparison of MA-DWAPSS and DWAPSS.

In this experiment, compared to Experiment 1, we introduced moving obstacles. Figure 17 shows the experimental results, with yellow squares representing moving obstacles. In Figure 17a, when MA-DWAPSS encounters a moving obstacle, based on the PSS, the robot detects that its speed is slower than that of the obstacle, so it waits for the obstacle to clear its path before proceeding. In contrast, Figure 17b,c show the results for the MA-DWA algorithm at different time points. In Figure 17b, the robot detects the moving obstacle but lacks the PSS guidance for the next action. The robot tries to bypass the obstacle, but in Figure 17c, the robot fails to avoid the obstacle in time because its speed is slower than the obstacle, leading to a collision.

This experiment further demonstrates the critical role of the parking scheduling strategy in dynamic environments. It enhances the robot's obstacle avoidance capability, thereby improving the safety of its movement.



(**a**) MA-DWAPSS (**b**) MA-DWA (**c**) MA-DWA

Figure 17. Comparison of MA-DWAPSS and MA-DWA.

6. Conclusions and Future Work

This paper proposes an integrated multi-robot path planning method, MA-DWAPSS, specifically designed for emergency scenarios such as fire rescue. The goal is to improve the efficiency and accuracy of multi-robot systems in collaboration and path planning. The method first uses an improved memetic algorithm in the global path planning phase, combining the advantages of local search and the genetic algorithm's global search to obtain an appropriate global path. The path is then smoothed using cubic Bézier curve fitting to avoid large angle deviations. Next, the DWAPSS algorithm is applied for local path planning, considering the dynamic characteristics of robot motion. This approach helps the robots respond to the constantly changing environment and newly appearing obstacles, ensuring strong flexibility and adaptability in the path planning.

Through simulation experiments, the paper verifies the advantages of the proposed method across several aspects, including path distance, time, obstacle avoidance, and path

https://doi.org/10.3390/sym18030499

smoothness. The results show significant improvements over traditional path planning methods. In particular, in the complex and dynamic environment of fire rescue, combining global and local planning strategies allows the multi-robot system to complete tasks more safely and efficiently.

Although the method in this paper has achieved good results in fire rescue simulation scenarios, there is still room for improvement and expansion. The current method mainly considers static and regular dynamic obstacles, but fire scenes may present additional types, such as collapsed buildings or unpredictable changes in the fire source. Therefore, future research could focus on enhancing the real-time environmental awareness of path planning algorithms to improve the system's ability to respond to more complex dynamic obstacles. In multi-robot systems, cooperation and communication between robots are crucial. Future studies could explore how to improve collaborative strategies or incorporate advanced methods, such as deep learning, to enhance information sharing and task allocation efficiency among robots, enabling multiple robots to work together more effectively on complex tasks.

Author Contributions: Conceptualization, Z.W. and S.C.; methodology, Z.W. and S.C.; coding, Z.W.; validation, Z.W. and S.H.; investigation, Z.W. and S.H.; writing—original draft preparation, Z.W. and S.C.; writing—review and editing, Y.S. and M.Z.; visualization, Z.W. and S.C.; supervision, S.C.; project administration, Y.S.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by the National Natural Science Foundation of China (Grant No. 61806119), the Natural Science Basic Research Plan In Shaanxi Province of China (No. 2024JC-YBMS-516), and Fundamental Research Funds for the Central Universities (No. GK202507001).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: Thanks to the anonymous reviewers and editor from the journal.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DWA	Dynamic Window Approach
DWAPSS	Dynamic Window Approach with Parking Scheduling Strategy
PSS	Parking Scheduling Strategy
MA-DWAPSS	Memetic algorithm with Dynamic Window Approach and Parking Scheduling Strategy
IGA-IDWA	Improved genetic algorithm with the improved dynamic window approach
A*-DWA	A* algorithm with the dynamic window approach
E-Q-Learning	Enhanced *Q*-learning algorithm

References

1. Madridano, Á.; Al-Kaff, A.; Martín, D.; de la Escalera, A. Trajectory planning for multi-robot systems: Methods and applications. *Expert Syst. Appl.* **2021**, *173*, 114660. [CrossRef]
2. Cheng, S.; Wang, Z.; Yang, J.; Sun, Y.; Zhang, M.; Hou, S.; Lu, H.; Luo, C.; Shi, Y. Robot Path Planning in Unknown Environments based on a Learning-guided Optimization Approach. *Tsinghua Sci. Technol.* **2025**, *accepted*. [CrossRef]
3. Reda, M.; Onsy, A.; Haikal, A.Y.; Ghanbari, A. Path planning algorithms in the autonomous driving system: A comprehensive review. *Robot. Auton. Syst.* **2024**, *174*, 104630. [CrossRef]
4. İlhan, İ. An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm Evol. Comput.* **2021**, *64*, 100911. [CrossRef]

https://doi.org/10.3390/sym18030499

5. Shan, Y.; Wang, X.; Cheng, S.; Zhang, M.; Xing, L. Knowledge-Guided Parallel Hybrid Local Search Algorithm for Solving Time-Dependent Agile Satellite Scheduling Problems. *Symmetry* **2024**, *16*, 813. [CrossRef]
6. Moscato, P.; Sun, H.; Haque, M.N. Analytic Continued Fractions for Regression: A Memetic Algorithm Approach. *Expert Syst. Appl.* **2021**, *179*, 115018. [CrossRef]
7. Lee, D.H.; Lee, S.S.; Ahn, C.K.; Shi, P.; Lim, C.C. Finite Distribution Estimation-Based Dynamic Window Approach to Reliable Obstacle Avoidance of Mobile Robot. *IEEE Trans. Ind. Electron.* **2021**, *68*, 9998–10006. [CrossRef]
8. Liu, L.; Wang, X.; Yang, X.; Liu, H.; Li, J.; Wang, P. Path planning techniques for mobile robots: Review and prospect. *Expert Syst. Appl.* **2023**, *227*, 120254. [CrossRef]
9. Cheng, C.; Sha, Q.; He, B.; Li, G. Path planning and obstacle avoidance for AUV: A review. *Ocean Eng.* **2021**, *235*, 109355. [CrossRef]
10. Jian, Z.; Chen, S.; Zhang, S.; Chen, Y.; Zheng, N. Multi-Model-Based Local Path Planning Methodology for Autonomous Driving: An Integrated Framework. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4187–4200. [CrossRef]
11. Gan, X.; Huo, Z.; Li, W. DP-A*: For Path Planing of UGV and Contactless Delivery. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 907–919. [CrossRef]
12. Alshammrei, S.; Boubaker, S.; Kolsi, L. Improved Dijkstra Algorithm for Mobile Robot Path Planning and Obstacle Avoidance. *Comput. Mater. Contin.* **2022**, *72*, 5939–5954. [CrossRef]
13. Li, X.; Wang, L.; An, Y.; Huang, Q.L.; Cui, Y.H.; Hu, H.S. Dynamic path planning of mobile robots using adaptive dynamic programming. *Expert Syst. Appl.* **2024**, *235*, 121112. [CrossRef]
14. Ning, Y.; Li, T.; Yao, C.; Du, W.; Zhang, Y. HMS-RRT: A novel hybrid multi-strategy rapidly-exploring random tree algorithm for multi-robot collaborative exploration in unknown environments. *Expert Syst. Appl.* **2024**, *247*, 123238. [CrossRef]
15. Xie, S.; Hu, J.; Bhowmick, P.; Ding, Z.; Arvin, F. Distributed Motion Planning for Safe Autonomous Vehicle Overtaking via Artificial Potential Field. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21531–21547. [CrossRef]
16. Wang, J.; Meng, Q.; Liu, R.; Sun, Z.Y. Obstacle avoidance path planning algorithm for intelligent vehicles on structured road. *Trans. Inst. Meas. Control* **2025**, *47*, 937–949. [CrossRef]
17. Lim, W.; Lee, S.; Sunwoo, M.; Jo, K. Hybrid Trajectory Planning for Autonomous Driving in On-Road Dynamic Scenarios. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 341–355. [CrossRef]
18. Zhou, Y.; Zhang, Y.; Zhang, Y. HMLPA*: A hierarchical multi-target LPA* pathfinding algorithm designed for dynamic indoor path network. *Int. J. Geogr. Inf. Sci.* **2025**, *39*, 1484–1517. [CrossRef]
19. Yu, J.; Yang, M.; Zhao, Z.; Wang, X.; Bai, Y.; Wu, J.; Xu, J. Path planning of unmanned surface vessel in an unknown environment based on improved D*Lite algorithm. *Ocean Eng.* **2022**, *266*, 112873. [CrossRef]
20. Cheng, S.; Jin, H.; Lu, H.; Shi, Y. A *Q*-Learning Based Brainstorming Optimization Algorithm for Solving Multimodal Optimization Problems. *IEEE Trans. Consum. Electron.* **2025**, *71*, 7317–7327. [CrossRef]
21. Ab Wahab, M.N.; Nazir, A.; Khalil, A.; Ho, W.J.; Akbar, M.F.; Noor, M.H.M.; Mohamed, A.S.A. Improved genetic algorithm for mobile robot path planning in static environments. *Expert Syst. Appl.* **2024**, *249*, 123762. [CrossRef]
22. Yu, Z.; Si, Z.; Li, X.; Wang, D.; Song, H. A Novel Hybrid Particle Swarm Optimization Algorithm for Path Planning of UAVs. *IEEE Internet Things J.* **2022**, *9*, 22547–22558. [CrossRef]
23. Cui, J.; Wu, L.; Huang, X.; Xu, D.; Liu, C.; Xiao, W. Multi-strategy adaptable ant colony optimization algorithm and its application in robot path planning. *Knowl.-Based Syst.* **2024**, *288*, 111459. [CrossRef]
24. Cheng, S.; Zhang, M.; Hou, S.; Sun, Y.; Lu, H.; Guo, W. A multi-agent deep reinforcement learning driven adaptive construction and search algorithm for time-varying agile Earth observation satellite scheduling. *Expert Syst. Appl.* **2026**, *304*, 130723. [CrossRef]
25. Maoudj, A.; Hentout, A. Optimal path planning approach based on Q-learning algorithm for mobile robots. *Appl. Soft Comput.* **2020**, *97*, 106796. [CrossRef]
26. Li, X.; Liang, X.; Wang, X.; Wang, R.; Shu, L.; Xu, W. Deep reinforcement learning for optimal rescue path planning in uncertain and complex urban pluvial flood scenarios. *Appl. Soft Comput.* **2023**, *144*, 110543. [CrossRef]
27. Gök, M. Dynamic path planning via Dueling Double Deep Q-Network (D3QN) with prioritized experience replay. *Appl. Soft Comput.* **2024**, *158*, 111503. [CrossRef]
28. Li, Y.; Zhao, J.; Chen, Z.; Xiong, G.; Liu, S. A Robot Path Planning Method Based on Improved Genetic Algorithm and Improved Dynamic Window Approach. *Sustainability* **2023**, *15*, 4656. [CrossRef]
29. Han, Y.; Li, C.; An, Z. Based on the Integration of the Improved A* Algorithm with the Dynamic Window Approach for Multi-Robot Path Planning. *Appl. Sci.* **2025**, *15*, 406. [CrossRef]
30. Lin, Y.; Du, L.; Yuen, K.F. Multiple unmanned surface vehicles pathfinding in dynamic environment. *Appl. Soft Comput.* **2025**, *172*, 112820. [CrossRef]

https://doi.org/10.3390/sym18030499

31. Deng, X.; Li, R.; Zhao, L.; Wang, K.; Gui, X. Multi-obstacle path planning and optimization for mobile robot. *Expert Syst. Appl.* **2021**, *183*, 115445. [CrossRef]
32. Manzoor, M.F.; Wu, Q.; Masood, R.J. Coordination Control of Wheeled Mobile Robot Using MPC. In Proceedings of the 2015 7th International Conference on Computational Intelligence, Communication Systems and Networks, Riga, Latvia, 3–5 June 2015; pp. 241–246. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/sym18030499

algorithms

MDPI

Article

Flight Schedule Problem Optimization Based on Discrete Memory-Enhanced Restructured Particle Swarm Optimization Algorithm

Wei Gao [1], Bingnan Wu [2], Jianhua Liu [2,*] and Daoming Tang [2]

[1] School of Artificial Intelligence, Fuzhou Technology and Business University, Fuzhou 350715, China
[2] School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou 350118, China
* Correspondence: jhliu@fjnu.edu.cn

Abstract

Flight Schedule Problem optimization is a typical NP-hard combinatorial optimization problem that is challenging to solve using traditional algorithms, so metaheuristic algorithms are commonly adopted for such problems. This paper proposes a Discrete Memory-Enhanced Restructured Particle Swarm Optimization algorithm (DMERPSO) to address Flight Scheduling Problem optimization. Firstly, this paper designs a hybrid particle encoding scheme capable of simultaneously handling flight time adjustments (integer variables) and route selections (categorical variables) for the Flight Schedule Problem. Secondly, a new update equation of particle positions is provided based on probability selection within the three terms of the Memory-Enhanced Restructured Particle Swarm Optimization (MERPSO) algorithm, and the calculation of the selection probability is designed. Thirdly, the two strategies and perturbation terms of MERPSO are improved in order to be adapted to optimize the discrete Flight Schedule Problem. Finally, simulation experiments are conducted using DMERPSO based on real flight data from multiple Chinese airports with the objective of minimizing total flight delays, leading to better solutions that are faster than various benchmark algorithms. The DMERPSO algorithm exhibits significant advantages in reducing total delays, improving solution stability, and enhancing robustness, which validates that DMERPSO provides an effective new approach for solving Flight Schedule Problem optimization problems.

Keywords: flight schedule optimization; restructured particle swarm optimization; discrete algorithm; memory-enhanced

1. Introduction

Flight Schedule Problem optimization is aimed at efficiently allocating resources, such as aircraft and crews, in order to enhance overall efficiency, which represents a complex combinatorial optimization problem constrained by aircraft routes, runway slot capacities, and turnaround requirements [1]. Traditional methods for solving the Flight Schedule Problem rely primarily on mathematical programming and precise algorithms [2]. The classical paradigm of traditional Flight Schedule Problem optimization is centered on mathematical programming, particularly relying on Mixed-Integer Linear Programming (MILP) to construct precise deterministic models [3]. This paradigm rigorously formulates objective functions—such as maximizing the total operational profit or minimizing overall costs—alongside a set of hard constraints [4]. Because the Flight Schedule Problem is

an NP-hard problem, solving it primarily depends on exact algorithms such as branch-and-bound [5] and branch-and-price [6]. The branch-and-price algorithm decomposes the problem into a master problem (selecting optimal flight strings/crew pairings) and pricing subproblems (generating new feasible task columns within resource-constrained networks), thereby effectively handling large-scale combinatorial spaces. The seminal work of Barnhart et al. [7] systematically established this framework, laying the algorithmic foundation for theoretical research on core issues such as fleet assignment and crew scheduling. However, pure methods of this kind often face computational intractability when addressing ultra-large-scale problems, multi-stage integration or dynamic stochastic disruptions. In response, current research frontiers focus on enhancing the scalability and practicality of the traditional paradigm through several key avenues [8–10]. These innovations allow the traditional cornerstone of mathematical programming to sustain its vitality, offering interpretable and high-performance solutions for complex scheduling problems through integration with modern computational intelligence.

However, with the increase in the scale of the Flight Schedule Problem, these traditional algorithms have become computationally impractical. To overcome the computational bottlenecks of traditional methods, both academic and industrial communities commonly employ metaheuristic algorithms to obtain high-quality approximate optimal solutions, which demonstrate strong compatibility with the large-scale and high-dimensional characteristics of Flight Schedule Problem optimization problems [11,12]. Metaheuristic algorithms have emerged as a significant computational paradigm in modern combinatorial optimization and have been widely and successfully applied in the field of Flight Schedule Problem [13]. Their core advantages lie in their strong generality, low dependency on specific problem structures, and their ability to effectively explore vast solution spaces. For example, the Genetic algorithm approach was used to build feasible daily schedules and assign optimal aircraft for hub-and-spoke airlines [14]; Ant Colony Optimization mimics the pheromone communication of ants to find optimal paths through positive feedback, making it particularly suitable for flight string or aircraft routing optimization [15]; and Tabu Search utilizes memory structures to avoid cyclical searching, which has been successfully applied to flight schedule development, tail assignment, and crew scheduling [16].

The Particle Swarm Optimization (PSO) algorithm is a metaheuristic approach which is characterized by its conceptual simplicity, fast convergence speed, and strong global search capability, and has demonstrated significant potential in solving complex optimization problems [17]. However, when addressing Flight Schedule Problem optimization involving numerous discrete variables and complex multimodal search spaces, the standard PSO algorithm's inherent "global best" guidance mechanism often leads to premature loss of population diversity, thereby trapping the algorithm in local optima—a phenomenon known as "premature convergence". To address these limitations, researchers have explored algorithmic improvements through two primary approaches, for example, integrating PSO with neighborhood search [18].

In recent years, the Restructured Particle Swarm Optimization (RPSO) algorithm was developed based on the linear system theory of PSO, which kept the optimization performance alongside algorithmic simplicity [19]. RPSO simplified a standard PSO framework by designing a new particle position updating formulation with convergent term and perturbation term, introduced novel search mechanisms [20]. RPSO has relatively good convergence theoretical support and is simple and easy to use. However, RPSO loses the population's memory, including the historical positions and fitness information of the particles. A variant of RPSO was designed, the memory-enhanced reconfigured particle swarm algorithm (MERPSO), which uses an experience selection strategy and a block

https://doi.org/10.3390/a19030233

search strategy to store the memory, constructing two new learning samples to replace the original learning samples so to balance local development and global exploration [21].

However, the original RPSO algorithm was primarily designed for continuous optimization problems, whereas Flight Schedule Problem optimization involves discrete decision variables. Its reliance on a single global best still risks premature convergence, particularly in highly multimodal problems such as Flight Schedule Problem optimization. This paper presents a discretized RPSO by transforming the variant of RPSO, the MERPSO algorithm, into a Discrete Memory-Enhanced Restructured Particle Swarm Optimization algorithm (DMERPSO), which is then applied to solve single-objective Flight Schedule Problem optimization. Experimental results demonstrate significant improvements in both computational efficiency and solution reliability. The main contributions of this work are as follows:

(1) The deep analysis of the model of Flight Schedule Problem optimization is conducted so that a complete flight route can be mathematically described. For the Flight Schedule Problem optimization model to be solved by the RPSO algorithm, the decision variables affecting flights are obtained and the *equation of* particle expression is designed. A hybrid particle encoding scheme capable of simultaneously handling integer-type (adjustment time) and categorical-type (route selection) decision variables is designed.

(2) A probability-based discrete position update formula is developed, enabling particles to probabilistically select between "exploitation" samples (*sg*), "exploration" samples (*mp*), and random perturbation terms (*dt*) to achieve balanced trade-offs between exploitation and exploration in discrete solution spaces.

(3) The core learning strategies are restructured through discretization: the "mean" operation in continuous EDS was transformed into a "voting/majority rule" mechanism for discrete EDS; the "hyper-rectangular block" structure in continuous BSS was reformulated into adaptive "integer interval blocks" and "discrete set blocks" suitable for different variable types, while preserving its core concept of dynamic splitting and merging.

2. Memory-Enhanced Restructured Particle Swarm Optimization Algorithm

This section first introduces the standard Particle Swarm Optimization (PSO) algorithm and its significant variant—Restructured Particle Swarm Optimization (RPSO). Then, it elaborates on how the Memory-Enhanced Restructured Particle Swarm Optimization (MERPSO) algorithm significantly improves performance by incorporating the memory-enhancement mechanisms of three terms: "experience selection", "block search" and "random perturbation".

2.1. Restructured Particle Swarm Optimization (RPSO)

PSO was inspired by the foraging behavior of bird flocks in 1995 [22]; the classical equations for updating velocity and position are as the Equations (1) and (2):

$$v_{id}(t+1) = w.v_{id}(t) + c_1 r_1 (pbest_{id}(t) - x_{id}(t)) + c_2 r_2 (gbes_d(t) - x_{id}(t)) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

where *pbest* is the particle's own historically optimal position and *gbest* is the globally best-known position of the entire swarm; $v_{id}(t)$ and $x_{id}(t)$ represent the velocity and position of particle i in the d-th dimension during the t-th iteration, respectively; w denotes the inertia

https://doi.org/10.3390/a19030233

weight; c_1 and c_2 are learning factors; r_1 and r_2 are random numbers uniformly distributed within [0, 1]. The PSO is highly susceptible to premature convergence toward a non-global optimal *gbest*, leading to diversity loss and entrapment in local optima [23].

Based on the results of linear system theoretical analyses of PSO, the Restructured Particle Swarm Optimization (RPSO) algorithm was proposed [19,20], which employs only a single position update equation, as shown in Equation (3):

$$x_{i+1} = c_r \times pbest + (1 - c_r)gbest + G(t) \tag{3}$$

where c_r represents the acceleration coefficient, a random number within the interval [0, 1], and G(t) denotes the perturbation term that can be designed by the user. For instance, the calculation method for G(t) was formulated as in Equation (4) [19]:

$$G(t) = (s \times r_3) \times (0.8 + \omega) \times (1 - \frac{t}{T}) \tag{4}$$

Here, s is set to one-quarter of the problem search range, r_3 is a random number within the interval $[-1, 1]$, ω is similar to the inertia weight of the PSO algorithm, with its value linearly decreasing between 0.2 and 0.8, t denotes the current iterating generation, and T signifies the maximum iteration. The RPSO algorithm simplifies the parametric structure and directly "restructures" new particle positions, wherein each particle has a certain probability of either learning from the global best particle (*gbest*) and the personal best particle (*pbest*) to take on the exploitation, while G(t) controls the exploration which is set by the designer.

2.2. Memory-Enhanced Restructured Particle Swarm Optimization (MERPSO)

RPSO remains dependent on the single *gbest*. Therefore, if *gbest* is trapped in a local optimum of the searching area, the evolutionary trajectory of the entire particles will still be severely misguided. To address the issue of RPSO's over-reliance on *gbest*, a Memory-Enhanced Restructured Particle Swarm Optimization (MERPSO) algorithm is proposed [21]. Effective guiding information should not be derived solely from the current single "champion" *gbest*, but rather from the collective wisdom crystallized by all "elites" of the entire search history.

2.2.1. Core Idea and Update Mechanism

In MERPSO, the concepts of *pbest* and *gbest* are completely abandoned, and the position update mechanism is transformed into a probabilistic selection-based learning approach. The position updates of particles are determined through probabilistic selection from three "learning samples", representing distinct search strategies, instead of by velocity and displacement [21]. The position update formula is as shown in Equation (5):

$$x_{i,d}^{t+1} = \begin{cases} sg_d^t, & if \quad rand() \le P_{sg} \\ mp_{i,d}^t, & if \quad P_{sg} < rand() \le p_{sg} + p_{mp} \\ dt_{i,d}^t, & otherwise \end{cases} \tag{5}$$

where $x_{i,d}^{t+1}$ denotes the new position of particle i in the d-th dimension at the $(t + 1)$th iteration. The three core learning samples are defined as follows: (1) The experience-guided sample (*sg*), which represents exploitation of promising regions. This sample is generated from an "experience set" storing numerous historical optimal solutions. (2) The block exploration sample (*mp*), representing exploration of uncharted regions. This sample is generated through a "block search strategy" designed to guide particles toward exploring less visited novel regions of the solution space. (3) The random perturbation term (*dt*),

https://doi.org/10.3390/a19030233

which introduces minor stochastic variations to the current particle position for maintaining population diversity and preventing premature algorithm stagnation.

2.2.2. Core Learning Strategies

The generating quality of *sg* and *mp* of the MERPSO algorithm is critical, which incorporates two sophisticated memory-enhanced strategies.

(1) Experience Selection Strategy (EDS)

The core of the EDS strategy lies in maintaining an external archive called the Experience Set (ES), which maintains the top *Np* solutions with the best fitness performance throughout the entire search history. The ES is updated after each iteration following fitness evaluation, preserving the most promising positions. The EDS strategy generates learning samples from the ES, denoted as SG. Each dimension of SG is independently determined according to Equation (6), meaning that every dimension of SG is computed as the average value of that dimension across K randomly selected solutions from the ES. In this algorithm, the value of K is set to 10.

$$sg_d = \frac{1}{K} \times \sum_{k=1}^{K} ES_d(r_k) \tag{6}$$

where $ES_d(r_k)$ denotes the d-th dimension of the r_k-th solution in the ES, and r_k is a random integer within the interval [1, Np]. To enhance the algorithm's exploitation capability in the later stages, Np is gradually reduced to narrow the scope of solutions preserved in the experience set. Np is calculated according to Equation (7):

$$N_p = \left\| ps \times \left(1 - \frac{t-1}{T}\right) \right\| \tag{7}$$

where ps represents the population size, $||\ ||$ denotes the ceiling operation, t is the current generation number, and T is the maximum iteration limit.

ESD stores the most historically fit solutions discovered by the algorithm since its initiation. When generating *sg* samples to guide the entire population, the algorithm randomly selects multiple elite historical solutions from the ES and forms a "consensus" direction by calculating their mean.

(2) Block Search Strategy (BSS)

The block search strategy (BSS) dynamically partitions the entire multi-dimensional search space into a series of hyper-rectangular "blocks" and continuously monitors the frequency of particle visits to each block. The block search strategy (BSS) generates learning samples, denoted as MP. The method involves each particle X_i having its corresponding learning sample, labeled as mp_i. The vector mp_i is generated by randomly combining its personal best pb_i and a randomly selected position pr in the block, as determined by Equation (8):

$$mp_{i,d} = \begin{cases} pb_{i,d}, & rand > q_i \\ pr_d\{b\}, & else \end{cases} \tag{8}$$

where $mp_{i,d}$ denotes the position of the d-th dimension in the i-th learning sample mp_i. For each dimension, it has a probability q_i of being $pr_{i,d}\{b\}$ and a probability $(1 - q_i)$ of being pb_i. Here, $pr_{i,d}\{b\}$ represents a random position within the b-th block of the d-th dimension; b is the selection result from a roulette wheel choice among all blocks in that dimension. The selection weight for each block is given by $W_b = M_b - n_p$, where n_p is the number of particles within that block and M_b is the upper limit for the number of particles in a block. For the algorithm's exploration, BSS tends to generate exploratory samples mp from

https://doi.org/10.3390/a19030233

those "less visited" (low-visit-frequency) blocks, thereby guiding particles to explore new search territories.

(3) The random perturbation term (DT)

The learning sample SG generated by EDS enhances the algorithm's exploitation capability, while the learning sample MP produced by BSS strengthens its exploration capability. The position update equation of particles in the MERPSO algorithm is given by Equation (9):

$$X_i^{t+1} = c_s' \times sg^t + c_s \times mp_i^t + D(t) \tag{9}$$

$$D(t) = (space \times r) \times (0.8 + \omega) \times (1 - \frac{t}{T})^{(C_m \times \frac{t}{T})} \tag{10}$$

where ${X_i}^{t+1}$ represents the position of the i-th particle at generation $t + 1$. SG_t denotes the SG selected at generation t, and ${mp_i}^t$ is the i-th MP at generation t. $D(t)$ is a time-varying perturbation term, calculated according to Equation (10). The parameter *space* is set to 20% of the problem's search space, r is a random number uniformly distributed in the interval $[-1, 1]$, ω is the inertia weight ranging from 0.8 to 0.2, and C_m is the perturbation coefficient.

3. Discrete MERPSO Algorithm for Solving Single-Objective Flight Schedule Problem Optimization

The MERPSO algorithm demonstrates competitive performance in continuous optimization problems. However, the Flight Schedule Problem constitutes a discrete optimization problem characterized by two functionally distinct decision variables: adjusted times and selected routes. So that the MERPSO algorithm can solve Flight Schedule Problem optimization, this section designs Discrete Memory-Enhanced Restructured Particle Swarm Optimization (DMERPSO), which involves the strategies of constructing particle representations, reformulating position update equations, designing learning strategies, incorporating perturbation terms, and implementing parameter discretization.

3.1. The Representation of Flight Schedule Problem Optimization for RPSO Algorithm

3.1.1. A 4D Trajectory Model for the Flight Schedule Problem Optimization

The primary flight schedule information includes the number of flights, departure and arrival locations, scheduled departure times, estimated arrival times, and flight routes. Table 1 presents a basic flight schedule.

Table 1. An example of flight schedule.

Flight Number	Departure Airport	Landing at the Airport	Estimated Time of Departure	Estimated Time of Landing	Flight Routes
F_1	Xi'an	Beijing	9:00	11:20	$r^{(1,1)}$, $r^{(1,2)}$, $r^{(1,3)}$
F_2	Shanghai	Shenzhen	10:15	12:30	$r^{(2,1)}$, $r^{(2,2)}$
F_3	Xi'an	Shanghai	18:20	21:15	$r^{(3,1)}$, $r^{(3,2)}$, $r^{(3,3)}$, $r^{(3,4)}$
F_4	Beijing	Shenzhen	7:45	9:10	$r^{(4,1)}$
F_5	Shenzhen	Xi'an	11:35	15:00	$r^{(5,1)}$, $r^{(5,2)}$, $r^{(5,3)}$

Note: The Set of flights number are denoted as F = {F_1, F_2, F_3,. . ., F_{Nf}}, where N_f is the total number of flights; $r^{(i,j)}$ denotes the j-th route of the i-th flight F_i.

The routes in flight scheduling are planned with one or more candidate routes, and different routes may exist for the same origin and destination. During the actual implementation of flight operations, the flight schedule often considers factors of adverse weather conditions and airspace congestion levels. While the departure location (*orig*) and landing location (*des*) remain unchanged, the actual departure time τ^{orig} is dynamically adjusted

https://doi.org/10.3390/a19030233

according to real-time situations. The arrival time of a flight τ^{des} is influenced by multiple factors, including the route distance, flight speed v^f, and congestion level, at the destination airport. In practical flight execution, routes are selected based on operational requirements.

Evaluating the flight schedule requires real-time position information for each flight within the flight plan. The 4D trajectory model is a commonly used framework which encompasses three spatial dimensions (flight position coordinates) and one temporal dimension (time information). An aircraft's flight path along route r can be conceptualized as straight-line segments between waypoints (wp) as navigation checkpoints. The connecting paths between waypoints are termed flight segments L. A complete flight route $r^{(i,j)}$ can be mathematically described using either waypoints or flight segments using Equation (11):

$$r^{(i,j)} = \left(orig^i, wp_1^{(i,j)}, wp_2^{(i,j)}, \cdots, des^i\right) = \left(L_1^{(i,j)}, L_2^{(i,j)}, \cdots\right) \in \Gamma^{(i,j)} \tag{11}$$

where $orig^i$ and des^i denote the origin and destination of flight F_i, $\Gamma^{(i,j)}$ represents the set of flight route, and Figure 1 illustrates a basic example of a flight route. Once the flight schedule parameter ${\tau_i}^{\text{orig}}$, the decision variable S_i for route selection, and the flight speed ${v^f}_i$ are determined, the flight trajectory can be calculated accordingly.

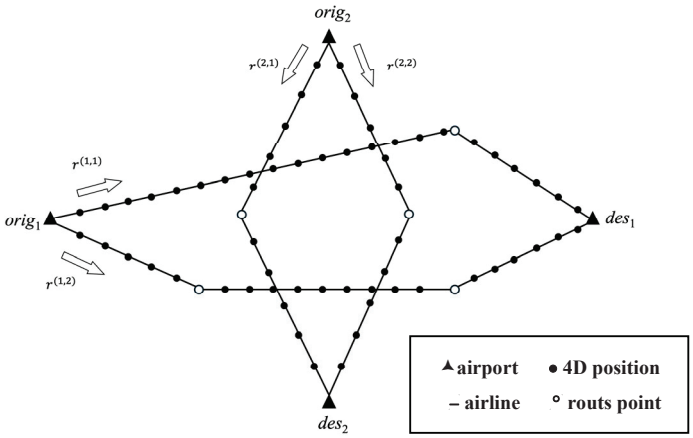


Figure 1. Flight route map.

3.1.2. Particle Representation and Flight Position Calculation in the RPSO Algorithm

For the Flight Schedule Problem optimization model, the decision variables affecting flights include the adjustment time of departure t^a and the selected route S^r for each flight. A particle X in the RPSO algorithm represents an adjustment scheme for the flight schedule, denoted as $X \leq t^s$. The dimension D of particle X equals $2N_f$, so the i-th particle X_i can be expressed by Equation (12):

$$X_i = \left\langle t^a_{(i,1)}, t^a_{(i,2)}, \cdots, t^a_{(i,Nf)}, S^r_{(i,1)}, S^r_{(i,2)}, \cdots, S^r_{(i,Nf)} \right\rangle \tag{12}$$

where ${t^a}_{(i,1)}$ denotes the adjusted departure time (in integer minutes) for the first flight of the i-th particle. Flight departure times are typically set at 5 min intervals, where $t^a = 3$ indicates a 15 min delay and $t^a = -4$ corresponds to a 20 min advance. ${S^r}_{(i,1)}$ represents the selected route for the first flight of particle i.

Since the altitude of flights is neglected and only their horizontal positions are considered, once the adjustment scheme X is determined, the position of flight j can be calculated using Equations (13)–(16):

$$T_j\left(wp_j^0\right) = \tau^{orig} + 5 \times t^a_{(i,j)} = T_j\left(orig_j\right) \tag{13}$$

https://doi.org/10.3390/a19030233

$$T_j\left(wp_j^k\right) = T_j\left(wp_j^{k-1}\right) + \frac{len\left(wp_j^k - wp_j^{k-1}\right)}{v_j^f}, j = 1, 2, \cdots, Nf \tag{14}$$

$$dis = 2\mathrm{R} \times \arcsin(\sqrt{\sin^2((latB - latA)/2)}) + \cos(latA) \times \cos(latB) \times \sin^2((lonB - lonA)/2) \tag{15}$$

$$loc_j(t^s) = \frac{t^s - T_j\left(wp_j^{z-1}\right)}{T_j\left(wp_j^z\right) - T_j\left(wp_j^{z-1}\right)} \times \left(wp_j^z - wp_j^{z-1}\right) \tag{16}$$

where $T_j(wp_j{}^k)$) denotes the time when flight j passes the k-th waypoint, τ^{orig} represents the scheduled departure time in the flight plan, $T_j(orig_j)$ indicates the adjusted departure time of flight j, and the arrival time is denoted as $T_j(des_j)$; $v^f{}_j$ signifies the average flight speed of flight j provided by the dataset; *len*(*A*-*B*) is a function calculating the distance between two waypoints A and B using the Haversine formula as Equation (15), where R represents the Earth's radius (6371 km), and *latA* and *lonA* respectively indicate the latitude and longitude coordinates of point A measured in radians; $loc_j(t^s)$ specifies the position of flight j at timestamp t^s, with t^s being discrete time points separated at 5 min intervals.

3.1.3. The Objective and Constraint Conditions of Flight Schedule Problem Optimization

The objective of Flight Schedule Problem optimization takes minimizing flight delay time (Delay) as the optimization objective, which consists of total ground delay (T^{dep}) and total arrival delay (T^{arr}). The calculation method is shown in Equations (17)–(19):

$$\min Delay = T^{dep} + T^{arr} \tag{17}$$

$$T^{dep} = \sum_{j=1}^{Nf}\left(5 \times t_j^a\right) \tag{18}$$

$$T^{arr} = \sum_{j=1}^{Nf}\left(T_j(des_j) - \tau_j^{des}\right) \tag{19}$$

The constraints of Flight Schedule Problem optimization primarily consider the utilization of airport runways, with the constraint conditions shown as Equations (20)–(23):

$$N_{ap}^{dep}(t^s) \le Q_{ap}^{dep}, \forall ap \in \Omega, \forall t^s \in [0, T^{\max}] \tag{20}$$

$$N_{ap}^{arr}(t^s) \le Q_{ap}^{arr}, \forall ap \in \Omega, \forall t^s \in [0, T^{\max}] \tag{21}$$

$$\left|t_{j,ap} - t_{j+1,ap}\right| \ge \tau_{sep}, i = 1, \cdots, Nf - 1, \forall ap \in \Omega \tag{22}$$

$$t^a \in [T_{\min}^a, T_{\max}^a] \tag{23}$$

where all airports ap constitute the set Ω. For each airport, the number of flights simultaneously taking off and landing within a given period is subject to upper limits, known as the airport's departure capacity $Q_{ap}{}^{dep}$ and arrival capacity $Q_{ap}{}^{arr}$; therefore, Equations (20) and (21) ensure that the number of departing and arriving flights at a given airport do not exceed its respective capacity. Furthermore, the interval between two consecutive departures at the same airport must exceed a specified threshold τ_{sep}, so Equation (22) guarantees that the separation time between consecutive departures at the same airport adheres to the safety threshold *Tsc*, which is set to 5 min. Equation (23) constrains the adjustment time for an individual flight within the schedule to a reasonable operational

https://doi.org/10.3390/a19030233

range; $T^a_{\min}$, $T^a_{\max}$ respectively represent the minimum and maximum values of the flight schedule adjustment.

3.2. Discrete MERPSO Algorithm

The MERPSO algorithm framework was originally proposed to address continuous optimization problems, whereas discrete optimization problems possess a different feasible solution space. This section extends the MERPSO framework to the Discrete MERPSO (DMERPSO) so to solve discrete optimization problems. Firstly, the methodology of the DMERPSO algorithm is designed; secondly, the discretization processing of each strategy of the algorithm is introduced; finally, the particle initialization method is presented.

3.2.1. Position Update Formula of DMERPSO

The discretization method of the PSO algorithm employs a transfer function to map velocity values onto the interval [0, 1]. Subsequently, these mapped values within the [0, 1] interval are converted into binary values (0 and 1) through stochastic probability.

Since the MERPSO algorithm does not require particle velocity for position updates, the position update of particles in the DMERPSO algorithm is influenced by three components: the learning samples *sg* and *mp*, as well as the perturbation term *dt*. Specifically, sample *sg* enhances both the convergence speed and precision of the particle, *mp* strengthens the particle's exploration capability, while *dt* maintains particle diversity. These three components directly take on particle position updates while serving distinct functional roles. Therefore, MERPSO directly updates particle positions based only on probability. The corresponding particle update equation is presented in Equation (24):

$$x^{t+1}_{i,d} = \begin{cases} sg^t_d, & rand \le P^{sg}_{i,d} \\ mp^t_{i,d}, & P^{sg}_{i,d} < rand \le \left(P^{sg}_{i,d} + P^{mp}_{i,d}\right) \\ dt^t_{i,d}, & other \end{cases} \tag{24}$$

where the update value of each dimension of a particle is selected from the corresponding dimensions of *sg*, *mp*, or *dt* with selection probabilities P^{sg}, P^{mp}, and P^{dt} respectively, where the sum of these three probabilities equals 1. The probability of selecting each term is determined based on factors including particle characteristics and the population's search status.

3.2.2. Calculation of Selection Probability

The DMERPSO algorithm determines the probability of updating particle positions by evaluating and scoring multiple scenarios, followed by normalizing the evaluated scores to obtain probabilities, thereby ensuring that the sum of the three probabilities equals 1. Each evaluation score can be expressed as $SC = W$ (weight value) $\times$ B (base score). The parameter values or ranges in this section are presented in Table 2. The three selection probabilities P^{sg}, P^{mp}, and P^{dt} in Equation (25) are calculated with Equations (26)–(28).

$$P^l = \frac{SC^l}{SC^{sg} + SC^{mp} + SC^{dt}}, l \in \{sg, mp, dt\} \tag{25}$$

$$SC^{sg} = w_1 \times b_{sg} \tag{26}$$

$$SC^{mp} = w_1 \times b_{mp} \tag{27}$$

$$SC^{dt} = \frac{w_2}{w_1} \times B_3 \tag{28}$$

https://doi.org/10.3390/a19030233

where the weight w_1 is a parameter that linearly iterates with the number of generations within the interval [0.5, 2], which makes particles to undergo random mutation with a higher probability during the early stages, thereby expanding the population's search scope while reducing the mutation probability in later stages so to enhance the convergence precision. The weight w_2 governs the maintenance of convergence in one-dimensional space, where particles remain identical to both learning samples sg and mp during the pursuit of convergence accuracy in later evolutionary stages, which is calculated using Equation (29).

$$w_2(i,d) = \begin{cases} 0, & x_{i,d} = sg_{i,d} = mp_{i,d} \;\&\; t > \frac{T}{4} \\ 1, & other \end{cases} \tag{29}$$

Table 2. Relevant parameters of selection probability.

Parameter	Size or Range
w_1	[0.5, 2]
w_2	0 or 1
B_1	20
B_2	100
B_3	20

The scores b_{sg} and b_{mp} function similarly to offset parameters in continuous MERPSO algorithms, computed through Equations (30) and (31), which regulate particles to exhibit greater displacement probability during initial exploration phases while maintaining positional stability or enabling minor adjustments with increased probability in subsequent optimization stages. In the two equations, ps denotes the number of particles of each generation of the algorithm, and $rank(f(X_i))$ is the sorting number of fitness of the objective function for the ith particle with the population.

$$b_{sg}(i,d) = \begin{cases} b_2 \times \frac{\text{rank}(f(X_i))}{ps}, & t < \frac{T}{2} \;\&\; dis_s \geq dis_m \;\&\; d \leq Nf \\ b_2 \times \left[1 - \frac{\text{rank}(f(X_i))}{ps}\right], & t \geq \frac{T}{2} \;\&\; dis_s < dis_m \;\&\; d \leq Nf \\ B_1, & \text{other} \end{cases} \tag{30}$$

$$b_{mp}(i,d) = \begin{cases} b_2 \times \frac{\text{rank}(f(X_i))}{ps}, & t < \frac{T}{2} \;\&\; dis_s < dis_m \;\&\; d \leq Nf \\ b_2 \times \left[1 - \frac{\text{rank}(f(X_i))}{ps}\right], & t \geq \frac{T}{2} \;\&\; dis_s \geq dis_m \;\&\; d \leq Nf \\ B_1, & \text{other} \end{cases} \tag{31}$$

Since the particle consists of two components, the adjustment time and trajectory, for the former $dis_m(i,d) = |x_{i,d} - mp_{i,d}|$ and $dis_s(i,d) = |x_{i,d} - sg_d|$, larger values indicate greater temporal discrepancies between the particle's corresponding dimension and the learning samples, which carries practical physical significance; for the latter component, differences between these two values merely reflect distinct route selections, making comparative magnitude analysis meaningless. Consequently, the scoring mechanisms for these two particle components differ, with parameter b_2 calculated using Equation (32).

$$b_2 = B_2 * \left| \frac{1}{1 + e^{0.01 \times (t - \frac{T}{2})}} - 0.5 \right| + B_1 \tag{32}$$

3.2.3. Improvement of Strategies and Perturbation Terms

Although the previous subsection adapted the position update formula of the MERPSO algorithm to Equation (24) and introduced a probability-based selection method

https://doi.org/10.3390/a19030233

to better solve discrete optimization problems, it still cannot guarantee that the resulting positions are integers, as is required for solutions to the Flight Schedule Problem optimization problem. Therefore, this section improves the experience selection strategy (ES), the block search strategy (BSS), and the algorithm's perturbation term, thereby transforming the MERPSO algorithm into the discrete variant, the MERSPO algorithm, tailored for solving the Flight Schedule Problem optimization problem.

(1) Experience Selection Strategy

In the discrete experience selection strategy, the storage principle and method of the Experience Set (ES) remain unchanged. The ES maintains an archive of the N_p best-performing solutions in terms of fitness throughout the entire search history, which is updated after each iteration's fitness evaluation. Each dimension of the learning sample sg is still selected independently; however, instead of calculating the average value of the selected solutions during the selection process, K solutions are randomly selected with replacement from the ES, and the most frequently occurring value among these selected solutions is chosen as the corresponding dimensional value for sg. In cases where multiple values have equal frequencies, one of these values is randomly selected as the dimensional value for sg. Here, K = 10, and N_p is calculated using Equation (33):

$$N_p = \left\| (ps - 5) \times \left(1 - \frac{t-1}{T}\right) + 5 \right\| \tag{33}$$

(2) Block Search Strategy

In the discrete block search strategy, the configuration of blocks varies according to particle dimensions. For the first to N_f-th dimensions of particles (i.e., flight adjustment time), blocks still maintain boundaries, where all integers on these boundaries can serve as selection ranges for mp. Taking Figure 2 as an example, which represents a block with a lower boundary of 1 and an upper boundary of 3, when selecting $pr_d\{b\}$ according to Equation (15), mp can take any integer value within {1,2,3}. The splitting and merging operations continue to be applied to blocks in the first to Nf-th dimensions. An additional operation is introduced: when the difference between block boundaries is even and splitting becomes necessary, a new block will be formed by combining the original lower boundary with the integer value of (upper boundary + lower boundary)/2, as illustrated in Figure 2. However, for the (Nf + 1)-th to 2Nf-th dimensions of particles (i.e., route selection components), there exists no continuous practical significance between adjacent integers. Consequently, blocks in this dimension range are defined without boundaries—for instance, block {5,8} indicates that eight particles have selected route number 5 in this dimension. Therefore, neither merging nor splitting operations are performed on blocks within this dimensional range.

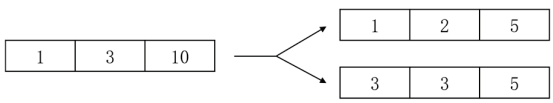


Figure 2. Discrete block splitting operation.

With the increase in particle dimensions, higher demands are placed on memory storage. To conserve memory and reduce the computational time, this section stores the data blocks of dimensions 1 to Nf and dimensions Nf + 1 to 2Nf separately in two matrices, BL1 and BL2. Two additional matrices are employed to record the starting positions of each dimension, which are referred to as boundary matrices (Edge Matrices) EM1 and EM2. The correspondence between BL and EM is illustrated in Figure 3.

https://doi.org/10.3390/a19030233

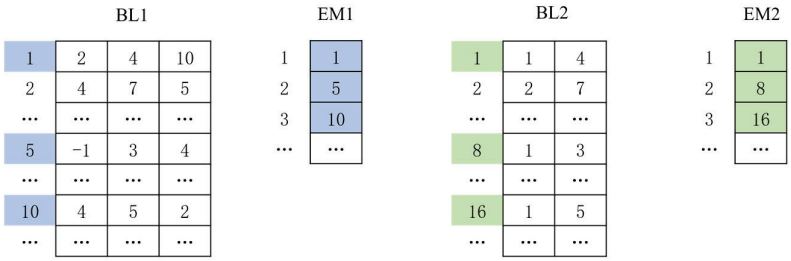


Figure 3. Correspondence between BL and EM.

(3) Perturbation Term

The perturbation operation performs single-point mutation on the original particle basis, with a mutation probability of P^{dt} for each dimension. The parameter P^{dt} linearly decreases from 0.6 to 0.2 as the number of generations increases. The variation range for the first Nf dimensions is $[R^{dt}{}_{min}, R^{dt}{}_{max}]$, while the variation range for the last Nf dimensions corresponds to any selectable flight path, as formulated in Equations (33)–(36):

$$space_t^{dt} = \frac{\left(T_{\max}^a - T_{\min}^a\right)}{2} \times \left(1 - 0.8 \times \frac{t}{T}\right) \tag{34}$$

$$R_{\min}^{dt} = \max\left(T_{\min}^a, x_{i,d} - space_t^{dt}\right) \tag{35}$$

$$R_{\max}^{dt} = \min\left(T_{\max}^a, x_{i,d} + space_t^{dt}\right) \tag{36}$$

3.2.4. The Methods for Initializing Particles and Satisfying Constraints

The adjustment time is categorized into five levels: slightly early, almost no adjustment, slightly delayed, moderately delayed, and severely delayed, defined as $t^a = (-6, 0]$, (0, 3], (3, 8], (8, 18], (18, 36]. During population initialization, dimensions 1 to Nf of each particle randomly select one of these five intervals to generate adjustment times, while route selection dimensions are chosen randomly. This process forms the initial population p^0. After evaluating p^0, the top 1/3 particles with the best fitness performance are retained. These preserved particles undergo single-point mutation with probability P^{p0}, where each particle performs three mutation operations. All mutated particles constitute the new population p^1, which serves as the initialized population for the algorithm. This initialization approach ensures sufficient uniformity in particle distribution across the search space, enhances population diversity, and facilitates global exploration. Simultaneously, eliminating the worst-performing particles avoids excessive computational resource consumption on inferior solutions, thereby improving convergence precision.

During the optimization process, in order to ensure that the position of each particle in the population is always within the feasible region that satisfies the constraint limits (such as the capacity limits for runway takeoff and landing, and the minimum takeoff interval limit), the particle solution after each position update is always compared with the constraint conditions. If it does not meet the constraint condition limits, a random method is used to generate a new position, thereby repairing the particle solution to ensure that the search solution satisfies the operational constraints.

4. Simulation Experiment and Verification

4.1. Flight Data Collection and Processing

The dataset used in the experiments consists of real flight plans provided by an aviation data communication company. The flights originate from eight flight schedule instances from different operational periods across nearly 200 airports in China, ensuring authenticity and generalizability [24]. Each dataset contains the following information:

https://doi.org/10.3390/a19030233

flight number, coordinates of departure airport, coordinates of arrival airport, scheduled departure time (τ^{orig}), scheduled arrival time (τ^{des}), route (including waypoints), actual trajectory, and average speed during actual flight. To simulate temporary adjustments to flight plans before execution and introduce additional scheduling difficulty, the departure and arrival times of certain flights were modified. The characteristics of these flight datasets are summarized in Table 3.

Table 3. Initial features of the experimental dataset.

Dataset	Number of Flights	Number of Airports	Number of Waypoints	Number of Routes	Initial Delay (min)	Adjusted Initial Delay (min)
M1	937	181	826	2053	10,812	-
M2	925	171	794	2113	10,752	-
M3	952	181	845	2331	13,734	15,297
M4	932	164	790	2374	18,392	20,752
M5	953	182	817	2076	26,596	28,361
M6	920	171	792	2036	48,905	51,802
M7	810	172	863	1983	20,744	23,850
M8	809	164	854	2130	26,744	29,501

Among them, the initial delay refers to the sum of delay times for all flights when the flight schedule is actually executed, while the adjusted initial delay represents the sum of delay times under increased optimization difficulty during actual execution. Datasets M1 and M2 represent original flight schedules; M3 and M4 simulate mild delay scenarios by randomly selecting 10% of flights in the original schedules and simultaneously advancing or delaying both departure and arrival times by identical time intervals (ranging from 30 min early to 30 min late); M5 and M6 simulate moderate delay scenarios by randomly selecting 3% of flights in the original schedules with adjusted time windows (either advanced by 30–60 min or delayed by 30–60 min); M7 and M8 first implement mild delay simulations on 10% of randomly selected flights as in M3/M4, then additionally apply moderate delay adjustments to 3% of remaining unadjusted flights (representing 3% of total flights) with modified time windows (advanced by 30–60 min or delayed by 30–60 min).

4.2. Experimental Performance

To validate the effectiveness of the DMERPSO algorithm, comparative experiments were conducted with RPSO [19], ADFPSO [25], SpadePSO [26], L-SHADE [27] and JSO [28] algorithms. For several continuous algorithms, in order to discretize them into integer encoding, the real-valued solutions of these algorithms underwent decimal rounding processing techniques to obtain integers. If their integer was greater than or less than the maximum and minimum values of each dimension integer within the given range of Flight Schedule Problem optimization problem, then the maximum and minimum values were taken, which ensured that the solutions of the comparison algorithms fell within the integer solution space of the Flight Schedule Problem optimization problem.

For each comparison algorithm, parameter settings followed the recommended values specified in their respective original papers. Each algorithm was independently executed 30 times on every dataset. To ensure fair comparison, all algorithms shared identical initial populations and employed the same stopping criterion based on maximum fitness evaluations (*MaxFEs*). The population size was set to 30 and MaxFEs was configured as 30,000 for all compared algorithms. All experiments were performed on a computer running the 64-bit version of Windows 11 Pro, with hardware specifications including an Intel Core i5 @3 GHz CPU and 32 GB RAM. All algorithms were implemented in MATLAB R2023a.

https://doi.org/10.3390/a19030233

The optimization results of the DMERPSO algorithm and these comparative algorithms are presented in Table 4. The data in the table represent the mean and variance of total delays (in minutes) across 30 independent experiments. The symbols (+), (−), and (≈) indicate that the corresponding algorithm is significantly better, significantly worse, or approximately equivalent to the DMERPSO algorithm in terms of performance based on t-test results with $p \leq 0.05$. The minimum mean total delay for each dataset is highlighted in bold. Experimental results demonstrate that the DMERPSO algorithm can significantly reduce total flight delays and optimize flight schedules. It achieves superior performance over other comparative algorithms across all eight datasets, particularly showing the best performance on the more challenging M7 and M8 datasets.

Table 4. Total delay optimization results of the DMERPSO algorithm and the comparative algorithms (The bold data represents the best result).

Dataset	Total Delay	DMERPSO	RPSO	ADFPSO	SpadePSO	L-SHADE	JSO
M1	mean	$\mathbf{1.07 \times 10^3}$	7.52×10^4 (+)	2.31×10^4 (+)	8.88×10^4 (+)	1.59×10^4 (+)	9.43×10^3 (+)
	std.	1029.6	2761	1847.3	1331.8	10360.0	3859.4
M2	mean	$\mathbf{2.15 \times 10^3}$	7.71×10^4 (+)	2.27×10^4 (+)	8.63×10^4 (+)	1.44×10^4 (+)	1.05×10^4 (+)
	std.	1104.8	2721.2	2586	1387.1	4938.7	1977.0
M3	mean	$\mathbf{3.16 \times 10^3}$	7.83×10^4 (+)	2.20×10^4 (+)	9.22×10^4 (+)	2.03×10^4 (+)	9.80×10^3 (+)
	std.	745.27	3820.6	2271.8	888.88	13375.1	3325.8
M4	mean	$\mathbf{4.23 \times 10^3}$	8.29×10^4 (+)	2.56×10^4 (+)	9.02×10^4 (+)	1.36×10^4 (+)	1.45×10^4 (+)
	std.	1097.3	3818.9	2632.3	1560.7	8303.3	3917.3
M5	mean	$\mathbf{6.13 \times 10^3}$	8.05×10^4 (+)	2.91×10^4 (+)	9.73×10^4 (+)	2.53×10^4 (+)	1.34×10^4 (+)
	std.	1061.1	2250.6	2107.7	1611.6	12115.3	3013.3
M6	mean	$\mathbf{2.16 \times 10^3}$	7.36×10^4 (+)	2.50×10^4 (+)	9.14×10^4 (+)	1.88×10^4 (+)	1.10×10^4 (+)
	std.	1499	3405.5	1627	1522.4	12011.1	3525.0
M7	mean	$\mathbf{3.27 \times 10^3}$	5.64×10^4 (+)	2.29×10^4 (+)	7.83×10^4 (+)	1.85×10^4 (+)	1.31×10^4 (+)
	std.	1119.5	2496.8	2788.4	1562.5	9224.8	3852.6
M8	mean	$\mathbf{1.89 \times 10^3}$	6.02×10^4 (+)	1.65×10^4 (+)	7.83×10^4 (+)	8.21×10^3 (+)	8.25×10^3 (+)
	std.	667.87	3513	2584.3	1257.8	4020.4	2882.0

To better demonstrate the advantages of the proposed algorithm, the Friedman test was first applied to the mean values of the Total Delay Optimization Results from Table 4, followed by the Wilcoxon signed-rank test to compare the performance of the RMERSPO algorithm with other algorithms. Table 5 presents the rankings of algorithms based on the mean Total Delay. The last row of the table shows the Friedman test results; the p-value is very small, indicating a significant difference in algorithm performance.

The Wilcoxon signed-rank test was used to compare two algorithms in all pairwise comparisons. The DMERSPSO algorithm was compared with the other three algorithms using the Wilcoxon signed-rank test at a significance level of $\alpha = 0.05$. The summarized results are presented in Table 6, with the signs +, ≈, and −. The sign '+' means that DMERSPSO performs significantly better than the compared algorithm; '−' means that DMERSPSO performs significantly worse; while '≈' means that there is no significant difference between the two algorithms. From the results in the table, it can be seen that DMERSPSO performs overwhelmingly better than the other algorithms.

https://doi.org/10.3390/a19030233

Table 5. The rank of algorithms based on the Total Delay Mean of dataset and Friedman test.

Dataset	DMERPSO	RPSO	ADFPSO	SpadePSO	L-SHADE	JSO
M1	1	6	5	7	4	3
M2	1	6	5	7	4	3
M3	1	6	5	7	4	3
M4	1	6	5	7	3	4
M5	1	6	5	7	4	3
M6	1	6	5	7	4	3
M7	1	6	5	7	4	3
M8	1	6	5	7	3	4
Mean rank	1	6	5	7	3.75	3.25
Friedman Test	$\chi^2 = 24, p = 1.5880 \times 10^{-8}$					

Table 6. The Wilcoxon signed-rank results of DMERPSO versus other algorithms ($\alpha = 0.05$).

Algorithm	+/≈/−
RPSO	8/0/0
ADFPSO	8/0/0
SpadePSO	8/0/0
L-SHADE	8/0/0
JSO	8/0/0

To verify the stability of the algorithm's operation, box plots were generated to analyze the total delay obtained from 30 independent runs on eight datasets, with the results shown in Figure 4. The experimental results demonstrate that the DMERPSO algorithm achieves superior optimization performance and exhibits higher stability.

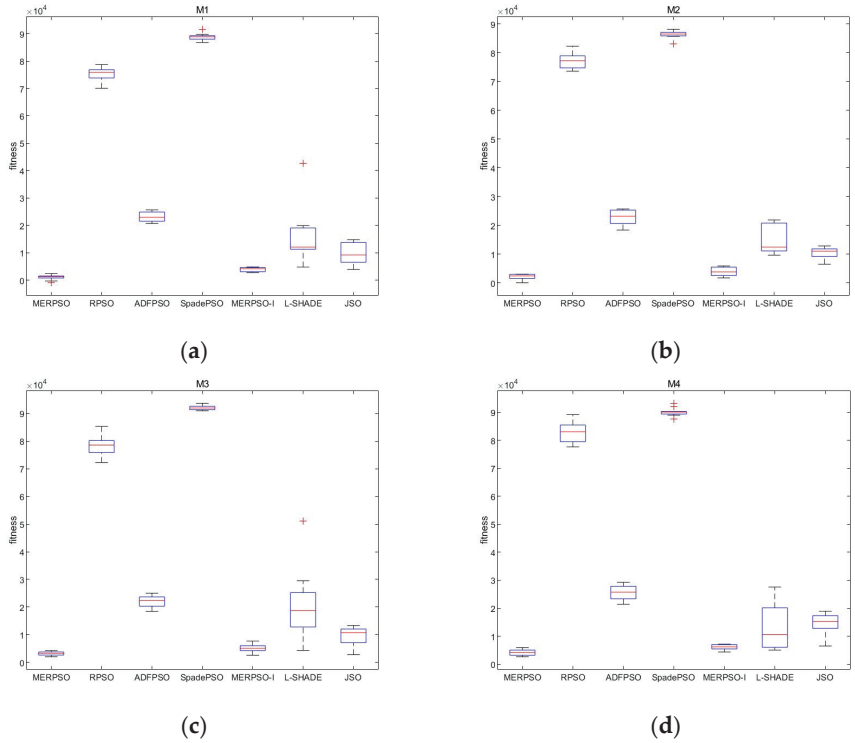


Figure 4. *Cont.*

https://doi.org/10.3390/a19030233

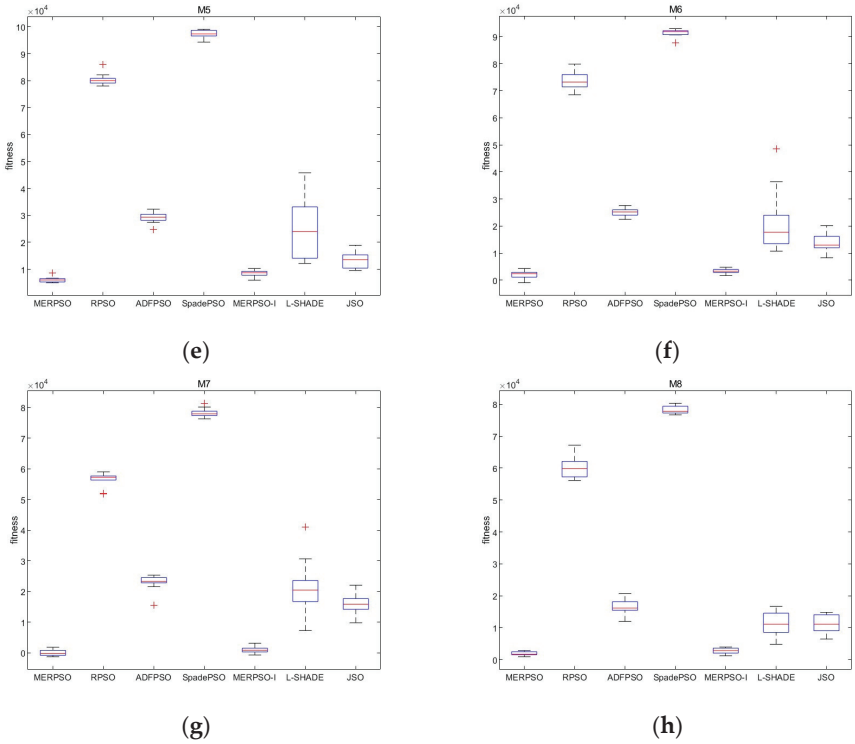


Figure 4. Box plots of the results on the eight datasets. (**a**) M1; (**b**) M2; (**c**) M3; (**d**) M4; (**e**) M5; (**f**) M6; (**g**) M7; (**h**) M8.

4.3. Effectiveness Analysis of the Initialization Method

The DMERPSO algorithm introduces a population initialization method that incurs the computational cost of one generation. To analyze the effectiveness of this strategy, convergence curves were plotted by comparing the average best fitness per generation across 30 independent experiments between the DMERPSO algorithm, the DMERPSO-I variant (without the initialization strategy), and three baseline algorithms. The convergence curves of these algorithms on eight datasets are presented in Figure 5. As observed from the convergence results, the DMERPSO algorithm demonstrates strong sustained convergence capability. In the middle to late iteration stages, when other algorithms become trapped in local optima, DMERPSO exhibits superior ability to escape local optima through continuous exploration and exploitation of the solution space to seek better solutions. Although the DMERPSO-I variant without the initialization strategy achieves better performance than DMERPSO in early iterations, the population diversity introduced by the initialization strategy enables DMERPSO to achieve better convergence accuracy in later stages. This validates the effectiveness of the proposed initialization strategy.

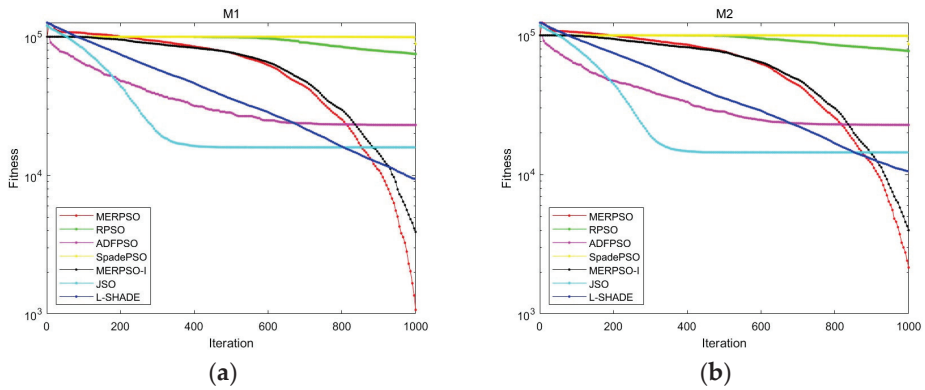


Figure 5. *Cont.*

https://doi.org/10.3390/a19030233

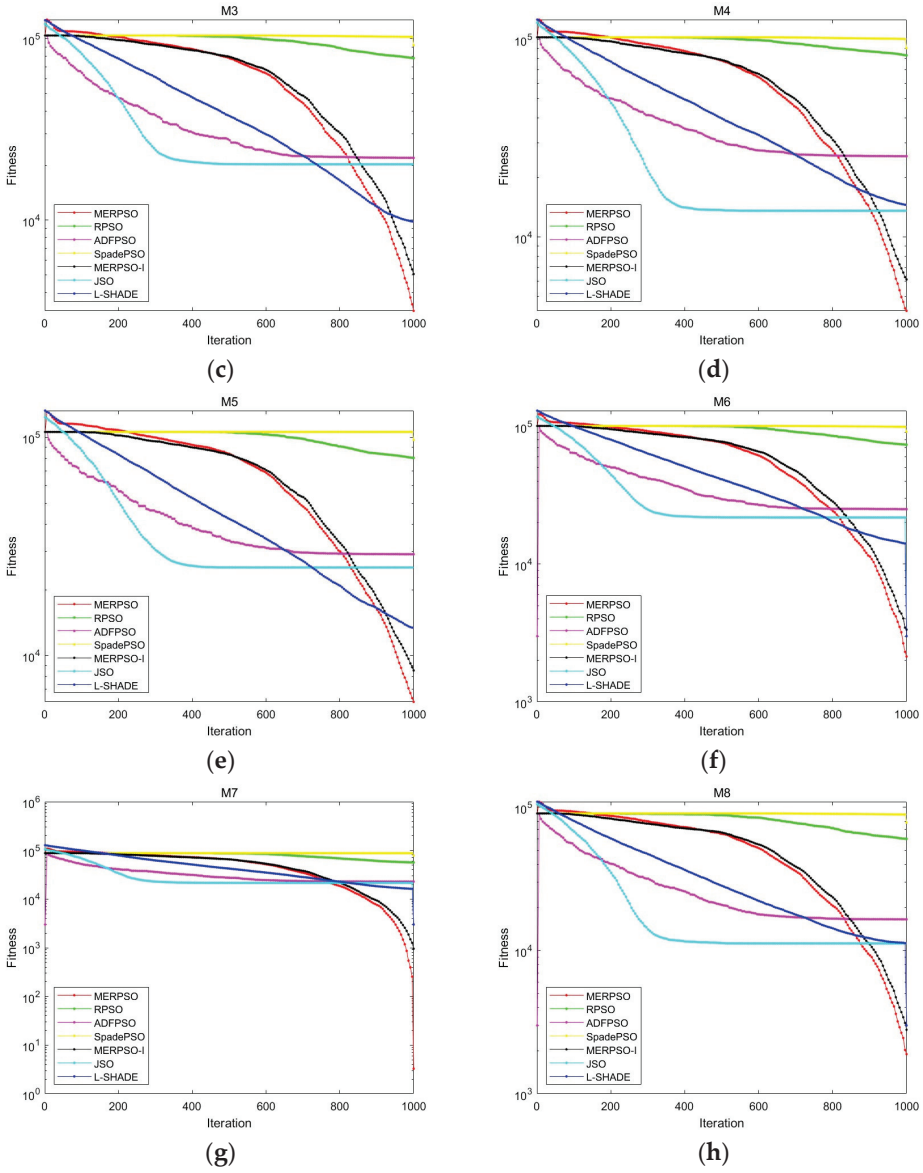


Figure 5. Convergence curves of the DMERPSO algorithm and the comparison algorithms. (**a**) Curves of convergence on M1; (**b**) convergence curves on M2; (**c**) curves of convergence on M3; (**d**) curves of convergence on M4; (**e**) curves of convergence on M5; (**f**) curves of convergence on M6; (**g**) curves of convergence on M7; (**h**) curves of convergence on M8.

5. Conclusions

To effectively address the discrete optimization problem of single-objective Flight Schedule Problem optimization, a Discrete Memory-Enhanced Restructured Particle Swarm Optimization algorithm (DMERPSO) is proposed. Experimental results demonstrate that the DMERPSO algorithm can effectively reduce flight delays, optimize air traffic capacity allocation, and achieve superior performance compared to benchmark algorithms.

The core innovation of the DMERPSO algorithm is designing the calculation of the selection probabilities from three components. More importantly, the DMERPSO algorithm improves the three terms—the experience selection strategy (ES), the block search strategy (BSS), and the algorithm's perturbation term—to directly transform them so as to be tailored to solve the integer code problem of the Flight Schedule Problem. The proposed integer coding scheme for Flight Schedule Optimization is significant and can be extended to other related problems. In the experiments conducted in this study, continuous algorithms were employed as comparisons. These algorithms performed decimal rounding on real-number

https://doi.org/10.3390/a19030233

solutions to obtain integers, which makes it difficult to maintain the favorable optimization characteristics of the solution space search, thereby introducing certain discrepancies in the experimental comparison.

Author Contributions: Conceptualization, J.L., W.G. and B.W.; methodology, B.W., W.G. and J.L.; software, B.W. and D.T.; validation, W.G., D.T., J.L. and B.W.; formal analysis, W.G., B.W., D.T. and J.L.; investigation, W.G., B.W., D.T. and J.L.; resources, J.L. and W.G.; data curation, B.W., D.T., J.L. and W.G.; writing—original draft preparation, B.W. and W.G.; writing—review and editing, W.G., J.L. and B.W.; visualization, B.W. and D.T.; supervision, J.L.; project administration, J.L.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Provincial Natural Science Foundation of Fujian, China (2023J01349).

Data Availability Statement: Data is contained in https://github.com/buaaguotong/ATFM-Benchmark (accessed on 10 March 2026).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bertsimas, D.; Patterson, S.S. The Air Traffic Flow Management Problem with Enroute Capacities. *Oper. Res.* **1998**, *46*, 406–422. [CrossRef]
2. Hu, H.; Sun, J.; Du, B. Air Traffic Management in Dense Airspace via Network Flow Optimization. *J. Aerosp. Inf. Syst.* **2025**, *22*, 433–446. [CrossRef]
3. Eltoukhy, A.E.E.; Chan, F.T.S.; Chung, S.H. Airline schedule planning: A review and future directions. *Ind. Manag. Data Syst.* **2017**, *117*, 1201–1243. [CrossRef]
4. Wang, D.; Wu, Y.; Hu, J.Q.; Liu, M.; Yu, P.; Zhang, C.; Wu, Y. Flight schedule recovery: A simulation-based approach. *Asia-Pac. J. Oper. Res.* **2019**, *36*, 1940010. [CrossRef]
5. Yan, S.; Tu, Y.P. Multifleet routing and multistop flight scheduling for schedule perturbation. *Eur. J. Oper. Res.* **1997**, *103*, 155–169. [CrossRef]
6. Soykan, B.; Erol, S. A branch-and-price algorithm for the robust airline crew pairing problem. *Savun. Bilim. Derg.* **2014**, *13*, 37–74.
7. Barnhart, C.; Belobaba, P.; Odoni, A.R. Applications of operations research in the air transport industry. *Transp. Sci.* **2003**, *37*, 368–391. [CrossRef]
8. Mercier, A.; Cordeau, J.F.; Soumis, F. A computational study of Benders decomposition for the integrated aircraft routing and crew scheduling problem. *Comput. Oper. Res.* **2005**, *32*, 1451–1476. [CrossRef]
9. Reiners, T.; Pahl, J.; Maroszek, M.; Rettig, C. Integrated aircraft scheduling problem: An auto-adapting algorithm to find robust aircraft assignments for large flight plans. In *Proceedings of the 2012 45th Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2012*; IEEE: Piscataway, NJ, USA, 2012; pp. 1267–1276.
10. Cadarso, L.; Vaze, V. Passenger-centric integrated airline schedule and aircraft recovery. *Transp. Sci.* **2023**, *57*, 813–837. [CrossRef]
11. Delahaye, D.; Odoni, A.R. Airspace Congestion Smoothing by Stochastic Optimization. In *Evolutionary Programming VI, Proceedings of the 6th International Conference*; Springer: Berlin/Heidelberg, Germany, 1997; pp. 163–176.
12. Xiao, M.; Hong, C.; Cai, K. Optimizing the safety-efficiency trade-off on nationwide air traffic network flow using cooperative co-evolutionary paradigm. *Sci. Rep.* **2025**, *15*, 20377. [CrossRef] [PubMed]
13. Cecen, R.K.; Durmazkeser, Y. Meta-Heuristic Algorithms for Aircraft Sequencing and Scheduling Problem. In *Progress in Sustainable Aviation*; Springer International Publishing: Cham, Switzerland, 2022; pp. 107–118.
14. Kablan, A.A.; Elberkawi, H.R.K.; Eldharif, E.A. Airline scheduling model using genetic algorithm. In *Proceedings of the 2024 IEEE 4th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA), Tripoli, Libya, 6–8 May 2024*; IEEE: Piscataway, NJ, USA, 2024; pp. 718–722.
15. Deng, G.F.; Lin, W.T. Ant colony optimization-based algorithm for airline crew scheduling problem. *Expert Syst. Appl.* **2011**, *38*, 5787–5793. [CrossRef]
16. Li, M.; Hao, J.K.; Wu, Q. Learning-driven feasible and infeasible tabu search for airport gate assignment. *Eur. J. Oper. Res.* **2022**, *302*, 172–186. [CrossRef]
17. Abualigah, L.; Sheikhan, A.; Ikotun, A.M.; Zitar, R.A.; Alsoud, A.R.; Alshourbaji, I.; Hussien, G.; Jia, H. Particle Swarm Optimization Algorithm: Review and Applications. *Metaheuristic Optim. Algorithms* **2024**, 1–14.
18. Ma, J.; Zhang, X.; Wang, C. Dynamic Optimization of Flight Schedules Based on Deep Reinforcement Learning. *Comput. Integr. Manuf. Syst.* **2021**, *27*, 2697–2708.

https://doi.org/10.3390/a19030233

19. Zhu, J.; Liu, J.; Wang, Z.; Chen, Y. Restructuring Particle Swarm Optimization Algorithm Based on Linear System Theory. In *Proceedings of the 2022 IEEE Congress on Evolutionary Computation (CEC), Padua, Italy, 18–23 July 2022*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–7.
20. Zhu, J.; Liu, J.; Chen, Y.; Xue, X.; Sun, S. Binary Restructuring Particle Swarm Optimization and Its Application. *Biomimetics* **2023**, *8*, 266. [CrossRef] [PubMed]
21. Wu, B.; Liu, J.; Li, S.; Li, M. Memory-Enhanced Reconstructive Particle Swarm Optimization Algorithm. *Comput. Eng. Appl.* **2025**, *61*, 116–127.
22. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In *Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995*; IEEE: Piscataway, NJ, USA, 1995; Volume 4, pp. 1942–1948.
23. Van den Bergh, F.; Engelbrecht, A.B. A Study of Particle Swarm Optimization Particle Trajectories. *Inf. Sci.* **2006**, *176*, 937–971. [CrossRef]
24. Guo, T.; Mei, Y.; Tang, K.; Du, W. A Knee-guided Evolutionary Algorithm for Multi-objective Air Traffic Flow Management. *IEEE Trans. Evol. Comput.* **2023**, *28*, 994–1008. [CrossRef]
25. Yu, F.; Tong, L.; Xia, X. Adjustable Driving Force Based Particle Swarm Optimization Algorithm. *Inf. Sci.* **2022**, *609*, 60–78. [CrossRef]
26. Wu, X.; Han, J.; Wang, D.; Gao, P.; Cui, Q.; Chen, L.; Liang, Y.; Huang, H.; Lee, H.P.; Miao, C.; et al. Incorporating Surprisingly Popular Slgorithm and Euclidean Distance-based Adaptive Topology into PSO. *Swarm Evol. Comput.* **2023**, *76*, 101222. [CrossRef]
27. Tanabe, R.; Fukunaga, A. Improving the search performance of SHADE using linear population size reduction. In *Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 6–11 July 2014*; IEEE: Piscataway, NJ, USA, 2014; pp. 1658–1665.
28. Brest, J.; Maučec, M.S.; Bošković, B. Single objective real-parameter optimization: Algorithm jSO. In *Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC), Donostia, Spain, 5–8 June 2017*; IEEE: Piscataway, NJ, USA, 2017; pp. 1311–1318.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/a19030233

AppliedMath

MDPI

Article

A New Algorithm for Finding Initial Basic Feasible Solutions of Transportation Problems

Douglas Kwasi Boah, Suleman Abudu Fiele * and Christian John Etwire

Department of Mathematics, School of Mathematical Sciences, C. K. Tedam University of Technology and Applied Sciences, Navrongo P. O. Box 24, Ghana; dboah@cktutas.edu.gh (D.K.B.); cetwire@cktutas.edu.gh (C.J.E.)
* Correspondence: fielesuleman@gmail.com

Abstract

This study introduces a deterministic fractional-penalty refinement of Vogel's Approximation Method (VAM) for generating high-quality initial basic feasible solutions (IBFS) in classical transportation problems. Unlike the traditional additive regret measure employed in VAM, the proposed method uses a multiplicative contrast ratio between the two smallest admissible costs in each row and column. This modification preserves the allocation structure of VAM while introducing scale-invariant prioritization that improves sensitivity to relative cost differences.The method was evaluated on thirty-four benchmark transportation problems drawn from the literature and self-constructed large-scale instances (up to 10×20). Performance was assessed using percentage optimality gaps relative to optimal solutions obtained via the Stepping–Stone and MODI procedures. Across all instances, the proposed approach achieved a mean optimality gap of 2.78%, compared to 5.22% for classical VAM, 14.97% for the Least Cost Method (LCM), and 45.78% for the Northwest Corner Method (NWCM). Dispersion of deviations was also reduced, indicating improved robustness across heterogeneous cost structures Statistical validation confirms the improvement over VAM: the paired t-test yielded $t = -3.17$ ($p = 0.00163$, one-sided), and the Wilcoxon signed-rank test produced $p = 6.10 \times 10^{-5}$. Computational experiments further show that the refinement does not increase runtime relative to classical IBFS procedures.The proposed method therefore constitutes a structured enhancement of VAM that improves initial solution quality while maintaining computational simplicity.

Keywords: transportation problem; initial basic feasible solution; north–west corner method; least cost method; vogel's approximation method

1. Introduction

The transportation problem (TP) is one of the earliest and most studied linear programming models. It was originally formulated by Hitchcock and later extended within the framework of linear programming [1,2]. Constructive heuristics for generating an initial basic feasible solution (IBFS) remain important because the quality of the IBFS can influence the number of iterations required in subsequent optimality procedures such as the Stepping-Stone or MODI methods [3].

Classical IBFS procedures include the North-West Corner Method (NWCM), the Least Cost Method (LCM), and Vogel's Approximation Method (VAM). Among these, VAM is widely regarded as the most cost-sensitive because it incorporates a penalty defined as the difference between the two smallest admissible costs in each row or column [4]. An effective heuristic approach for determining an initial basic feasible solution to the

transportation problem has been proposed, offering a practical efficiency and solution quality [5]. Despite its effectiveness, the additive penalty used in VAM is sensitive to absolute cost magnitudes and scaling. In heterogeneous cost environments, proportional dispersion between competing allocation options may not be fully captured by simple cost differences. This observation motivates the investigation of alternative penalty formulations that preserve the constructive structure of VAM while redefining its prioritization metric.

This study proposes a ratio-based (fractional) penalty defined as the quotient of the second-smallest and smallest admissible costs in each active line. The resulting measure emphasizes proportional cost contrast and remains invariant under positive cost scaling. Importantly, the allocation structure of VAM is retained; only the penalty computation is modified.

The objective of this paper is therefore not to introduce an entirely new heuristic family, but rather to provide a mathematically transparent and computationally reproducible refinement of the VAM penalty structure. Extensive benchmark testing and statistical validation are conducted to evaluate performance consistency across diverse problem instances.

The contribution of this study lies in introducing a scale-invariant multiplicative penalty mechanism within the classical Vogel's Approximation Method framework. Unlike the traditional additive regret measure used in VAM, the proposed fractional penalty captures proportional dispersion between competing transportation costs. This modification preserves the constructive allocation structure of VAM while providing an alternative prioritization mechanism that can improve the quality and stability of initial basic feasible solutions. The effectiveness of the approach is validated through computational experiments on thirty-four benchmark transportation problems together with statistical analysis of optimality gaps.

Research Gap and Contribution

Although numerous heuristics exist for obtaining initial basic feasible solutions (IBFS) in transportation problems, many widely used procedures remain rooted in either minimum-cost selection or additive regret ideas. Common IBFS methods include the Row Minimum Method, the Column Minimum Method, and the Maximum Difference Method. Other classical baselines are the Least Cost Method (LCM) and Vogel's Approximation Method (VAM).

Recent studies have introduced alternative penalty models and hybrid allocation strategies for constructing IBFS. For example, the Maximum Range Method was proposed to improve allocation prioritization [6]. Alternative procedures for generating initial feasible solutions have also been explored [7]. Furthermore, new penalty-based models motivated by probabilistic or distributional mechanisms have been developed [8]. Other algorithmic strategies for determining IBFS have also been reported in recent studies [9]. However, the theoretical implications of multiplicative cost dispersion in row/column prioritization remain largely unexplored.

Classical VAM employs an additive regret defined as the difference between the two smallest admissible costs. While effective, this measure is not scale-invariant and may underrepresent proportional dispersion when costs vary significantly across rows.

This study introduces a deterministic fractional-penalty refinement that replaces additive regret with a multiplicative contrast ratio. The contribution lies not in altering the allocation structure of VAM, but in modifying the prioritization mechanism in a theoretically motivated and statistically validated manner.

https://doi.org/10.3390/appliedmath6040058

2. Literature Review

Transportation problems remain a central topic in operations research due to their wide applicability in logistics, supply chain management, and production planning. Classical constructive heuristics for generating an initial basic feasible solution (IBFS), namely the North-West Corner Method (NWCM), the Least Cost Method (LCM), and Vogel's Approximation Method (VAM), are extensively discussed in standard operations research textbooks [10–12].

The continued relevance of these classical heuristics has been demonstrated in several comparative studies. Performance evaluations of traditional IBFS methods confirm their effectiveness as baseline techniques for transportation problems [13]. Further investigations into their computational behavior show that these heuristics remain competitive for small- and medium-scale transportation instances [14]. Additional empirical studies provide supporting evidence of their applicability in practical scenarios [15,16]. These classical methods also serve as important benchmarks for evaluating newly proposed algorithms [17].

The North-West Corner Method (NWCM) is attractive because of its simplicity and low computational cost. However, it ignores transportation costs and allocates shipments solely by traversing the transportation tableau from the top-left corner, which often leads to relatively coarse initial solutions [18]. The Least Cost Method (LCM) improves upon NWCM by selecting the smallest available unit transportation cost at each allocation step. This strategy generally produces better initial solutions than NWCM [19]. However, LCM may suffer from myopic allocation decisions and sensitivity to tie-breaking rules when multiple cells have identical costs [20].

Vogel's Approximation Method (VAM) introduces a penalty-based strategy defined by the difference between the two smallest costs in each row or column. This penalty mechanism allows VAM to prioritize allocations where suboptimal decisions would be most costly [21]. Several studies have shown that VAM typically produces solutions closer to optimality than NWCM and LCM [22].

In practice, classical IBFS procedures must also address implementation issues such as balancing through dummy rows or columns, degeneracy prevention, and deterministic tie-breaking rules. Improper handling of these issues can significantly affect solution quality and reproducibility [23]. Standard operations research references emphasize the need for systematic implementation rules to ensure consistent computational performance [24]. The transportation problem has been extensively studied in both applied and theoretical contexts, with contributions focusing on cost minimization and optimization techniques [25,26]. However, existing approaches still face challenges in improving computational efficiency during optimality checks. These limitations motivate ongoing research aimed at developing improved IBFS algorithms with stronger cost-awareness and enhanced computational robustness.

3. Methodology: Finding Initial Basic Feasible Solutions

3.1. Notation

Let $C = (c_{ij})$ be the $m \times n$ cost matrix with supplies $a_i > 0$ ($i = 1, \ldots, m$) and demands $b_j > 0$ ($j = 1, \ldots, n$). A transportation problem (TP) is balanced if $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$. If it is unbalanced, we add a dummy column (when total supply exceeds total demand) or a dummy row (when total demand exceeds total supply). The dummy line has cost 0 to all of its cells, and the added supply/demand equals the absolute imbalance.

Throughout all methods:

- Stopping criterion: Continue allocating until all supplies and demands are satisfied.
- Degeneracy safeguard: An IBFS must contain exactly $m + n - 1$ basic cells. If a row and a column are both exhausted by the same allocation, cross out one line and leave

https://doi.org/10.3390/appliedmath6040058

the other active with a zero residual, or place a tiny bookkeeping allocation ε in a zero-cost eligible cell (do not alter totals) to maintain $m + n - 1$ basic variables.

- Tie-breaking (deterministic): When ties occur, use: (i) prefer *row-first* over column when methods require choosing a line; (ii) within a line, choose the lowest cost; if still tied, choose the leftmost (smallest column index); if still tied, choose the topmost (smallest row index). Using a fixed rule ensures reproducibility.
- Quality measure: The IBFS cost is

$$Z_{\text{IBFS}} = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{1}$$

3.2. General Steps for Finding IBFS

1. Balance the problem. If total supply and total demand differ, add a zero-cost dummy row (when demand exceeds supply) or dummy column (when supply exceeds demand) so totals match.
2. Initialize. Mark all rows and columns as active, with their remaining supply/demand.
3. Choose a line or cell (selection rule). Apply one of the rules below to decide where to allocate next.
4. Allocate. In the chosen cell (i, j), set

$$x_{ij} = \min\{\text{remaining supply of row } i,\ \text{remaining demand of column } j\} \tag{2}$$

5. Update the rim. Reduce the row's remaining supply and the column's remaining demand by x_{ij}.
6. Cross out satisfied lines. Any row/column that reaches zero becomes inactive.
7. Handle degeneracy. If a single allocation makes both a row and a column reach zero, cross out one and keep the other active with zero balance (or place a formal ε in an eligible cell) so that the final IBFS has exactly $m + n - 1$ basic cells.
8. Repeat. Recompute the quantities required by the selection rule on the reduced tableau and return to Step 3 until all supplies and demands are satisfied.
9. Stop. When all rim requirements are met, the current x_{ij} constitute an IBFS.
10. The IBFS cost is computed using Equation (1).

Selection Rules (To Use in Step 3)

- North–West Corner Method (NWCM): Start at the top-left active corner, allocate, then move right when a column is satisfied and down when a row is satisfied.
- Least Cost Method (LCM): Choose the globally cheapest active cell.
- Vogel's Approximation Method (VAM): For each active line, the penalty is

$$P = c_{(2)} - c_{(1)} \tag{3}$$

where $c_{(1)}$ and $c_{(2)}$ are the smallest and second smallest costs in that line.

- Mean Minima Method: For each active line, compute

$$M = \frac{c_{(1)} + c_{(2)}}{2} \tag{4}$$

and select the line with the smallest mean.

3.3. Computational Complexity Analysis

At each allocation step, the algorithm requires identifying the two smallest elements in each active row and column. This operation has linear complexity in the number of

https://doi.org/10.3390/appliedmath6040058

active cells per iteration. Since exactly $m + n - 1$ allocations are performed in constructing an IBFS, the overall worst-case computational complexity remains comparable to classical VAM, namely $\mathcal{O}(mn)$ per iteration and

$$\mathcal{O}(mn(m+n)). \tag{5}$$

Because the allocation structure is identical to VAM and only the penalty computation differs, the proposed algorithm does not introduce additional asymptotic computational burden.

4. Results and Discussions

4.1. The New Algorithm for Finding Initial Basic Feasible Solutions (IBFS) of Transportation Problems

The following are the steps involved in the new algorithm.

4.1.1. Step 1: Balance Check

If the total supply does not equal total demand, the problem must be balanced. This condition is expressed as

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{6}$$

If Equation (6) does not hold, a dummy row (for excess demand) or dummy column (for excess supply) with zero transportation costs is introduced to balance the problem.

4.1.2. Step 2: Row and Column Fractional Penalties

For each active row i, determine the two smallest costs among the admissible cells. These are defined as

$$m_i^{(1)} = \min_j c_{ij} \tag{7}$$

and

$$m_i^{(2)} = \text{second-smallest of } \{c_{ij}\} \tag{8}$$

The fractional penalty for row i is then defined as

$$p_i = \frac{m_i^{(2)}}{m_i^{(1)}} \tag{9}$$

Similarly, for each active column j, the two smallest costs are obtained and the column penalty is defined as

$$q_j = \frac{m_j^{(2)}}{m_j^{(1)}} \tag{10}$$

If $m^{(1)} = 0$, the corresponding line is treated as having very large priority to avoid division by zero.

4.1.3. Step 3: Selection and Allocation

The row or column with the largest penalty is selected according to

$$P^* = \max\{p_i, q_j\} \tag{11}$$

https://doi.org/10.3390/appliedmath6040058

Within the selected row or column, the minimum-cost cell (i^*, j^*) is chosen and the allocation is computed as

$$x_{i^*j^*} = \min(a_{i^*}, b_{j^*}) \tag{12}$$

The supply and demand values are then updated accordingly.

4.1.4. Step 4: Iteration

After each allocation, the penalties are recomputed for the reduced tableau and Steps 2–3 are repeated until all supplies and demands are satisfied.

4.1.5. Step 5: IBFS Cost

The total cost of the obtained initial basic feasible solution is computed as

$$Z_{\text{IBFS}} = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{13}$$

4.2. Monotonicity Property of the Fractional Penalty

The ratio-based penalty also exhibits a monotonicity property that clarifies its prioritization behavior relative to the smallest admissible cost in a row or column.

Lemma 1. *Let $m^{(1)}$ and $m^{(2)}$ denote the smallest and second-smallest admissible costs in an active row (or column) with $m^{(1)} > 0$ and $m^{(2)} \geq m^{(1)}$. Define the fractional penalty*

$$p = \frac{m^{(2)}}{m^{(1)}}. \tag{14}$$

Then the penalty p is strictly increasing in $m^{(2)}$ and strictly decreasing in $m^{(1)}$.

Proof. Consider the penalty function $p(m^{(1)}, m^{(2)}) = \dfrac{m^{(2)}}{m^{(1)}}$ for $m^{(1)} > 0$. The partial derivatives are

$$\frac{\partial p}{\partial m^{(2)}} = \frac{1}{m^{(1)}} > 0, \tag{15}$$

and

$$\frac{\partial p}{\partial m^{(1)}} = -\frac{m^{(2)}}{(m^{(1)})^2} < 0. \tag{16}$$

Thus, increasing the second-smallest admissible cost increases the penalty, while increasing the smallest admissible cost decreases the penalty. Consequently, rows or columns with stronger proportional cost contrast receive higher priority in the allocation process. □

This monotonicity property provides a theoretical justification for the prioritization mechanism of the fractional penalty. In contrast to the additive penalty used in classical Vogel's Approximation Method, the ratio-based formulation emphasizes proportional cost dispersion, thereby capturing relative dominance among competing allocation options.

4.3. Demonstration of the New Algorithm

To illustrate the procedure, we consider a transportation problem adapted from standard literature [27]. The cost matrix, supplies, and demands are given in Table 1.

https://doi.org/10.3390/appliedmath6040058

Table 1. Transportation Tableau (BUA Cement Company Data).

	D1	**D2**	**D3**	**D4**	**D5**	**D6**	**Supply**
P1	25	30	20	40	45	37	37
P2	30	25	20	30	40	20	22
P3	40	20	40	35	45	22	32
P4	25	24	50	27	30	25	14
Demand	15	20	15	25	20	10	105

Since total supply equals total demand (105), the problem is balanced.

4.3.1. Iteration 1

The computed row and column penalties for the first iteration are presented in Table 2.

Table 2. Iteration 1: Penalty Workout (Rows and Columns).

Line	$m^{(1)}$	$m^{(2)}$	**Penalty**
Row P1	20	25	$p_1 = 25/20 = 1.25$
Row P2	20	20	$p_2 = 20/20 = 1.00$
Row P3	20	22	$p_3 = 22/20 = 1.10$
Row P4	24	25	$p_4 = 25/24 \approx 1.04$
Col D1	25	25	$q_1 = 25/25 = 1.00$
Col D2	20	24	$q_2 = 24/20 = 1.20$
Col D3	20	20	$q_3 = 20/20 = 1.00$
Col D4	27	30	$q_4 = 30/27 \approx 1.11$
Col D5	30	40	$q_5 = 40/30 \approx 1.33$
Col D6	20	22	$q_6 = 22/20 = 1.10$

$$p_i = \frac{m_i^{(2)}}{m_i^{(1)}}, \qquad q_j = \frac{m_j^{(2)}}{m_j^{(1)}} \tag{17}$$

The maximum penalty occurs in column D_5. The minimum cost in column D_5 is $c_{45} = 30$.

$$x_{45} = \min(14, 20) = 14 \tag{18}$$

Row P_4 is exhausted; demand of D_5 reduces to 6.

4.3.2. Iteration 2

Recompute penalties on the reduced tableau.

A tie occurs between row P_1 and column D_2. Using the deterministic row-first rule, select row P_1. Minimum cost in row P_1 is at D_3:

$$x_{13} = \min(37, 15) = 15 \tag{19}$$

Column D_3 is satisfied; supply of P_1 reduces to 22.

4.3.3. Iteration 3

Row P_2 has the largest penalty. Minimum cost in row P_2 is at D_6.

$$x_{26} = \min(22, 10) = 10 \tag{20}$$

Column D_6 is satisfied; supply of P_2 reduces to 12.

4.3.4. Iteration 4

Row P_3 has the largest penalty. Minimum cost in row P_3 is at D_2:

$$x_{32} = \min(32, 20) = 20 \tag{21}$$

https://doi.org/10.3390/appliedmath6040058

Column D_2 is satisfied; supply of P_3 reduces to 12.

4.3.5. Iteration 5

Row P_1 now has the largest penalty. Minimum cost in row P_1 is at D_1:

$$x_{11} = \min(22, 15) = 15 \tag{22}$$

Column D_1 is satisfied; supply of P_1 reduces to 7.

4.3.6. Iteration 6

Row P_2 has the largest penalty. Minimum cost in row P_2 is at D_4:

$$x_{24} = \min(12, 25) = 12 \tag{23}$$

Row P_2 is exhausted; demand of D_4 reduces to 13.

4.3.7. Iteration 7

Row P_3 has the largest penalty. Minimum cost in row P_3 is at D_4:

$$x_{34} = \min(12, 13) = 12 \tag{24}$$

Row P_3 is exhausted; demand of D_4 reduces to 1.

4.3.8. Iteration 8

Only row P_1 remains active.

$$x_{14} = \min(7, 1) = 1 \tag{25}$$

Column D_4 is satisfied; supply of P_1 reduces to 6.

4.3.9. Iteration 9

Final allocation:

$$x_{15} = 6 \tag{26}$$

All supplies and demands are satisfied.

Final Initial Basic Feasible Solution

$$X = \begin{pmatrix} 15 & 0 & 15 & 1 & 6 & 0 \\ 0 & 0 & 0 & 12 & 0 & 10 \\ 0 & 20 & 0 & 12 & 0 & 0 \\ 0 & 0 & 0 & 0 & 14 & 0 \end{pmatrix} \tag{27}$$

Number of allocations:

$$m + n - 1 = 4 + 6 - 1 = 9 \tag{28}$$

Hence, the solution is non-degenerate.

IBFS Cost

$$Z_{\text{IBFS}} = 25(15) + 20(15) + 40(1) + 45(6) + 30(12) + 20(10) + 20(20) + 35(12) + 30(14) \tag{29}$$

https://doi.org/10.3390/appliedmath6040058

Thus, the fractional-penalty algorithm produces an initial solution with total transportation cost:

$$\boxed{Z_{\text{IBFS}} = 2785.} \tag{30}$$

To further substantiate performance consistency, we computed the mean, median, and standard deviation of optimality gaps across all benchmark instances. Additionally, paired comparisons between VAM and the new method were conducted to determine whether improvements were systematic.

4.4. Optimality Gap Analysis

To evaluate the practical relevance of the new method, we compare the IBFS costs produced by each heuristic against the known optimal solution cost Z^* obtained via standard optimality procedures.

For each benchmark instance, we compute the percentage optimality gap as:

$$\text{Gap}(\%) = \frac{Z_{\text{IBFS}} - Z^*}{Z^*} \times 100 \tag{31}$$

This measure provides a normalized assessment of solution quality independent of cost magnitude. Lower values indicate closer proximity to optimality.

Across the benchmark set, the fractional-penalty variant frequently exhibits equal or smaller gaps compared to VAM, LCM, and NWCM. While improvements are moderate in magnitude for some instances, they are consistent across several categories of cost structures.

4.5. Comparison of the New Algorithm with Existing Algorithms in Terms of Solutions

Tables 3–8 compare the initial basic feasible solution costs produced by the new algorithm with those obtained by VAM, NWCM, and LCM across the benchmark instances.

Table 3. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 1–4).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
1	5×5	[28] C_{ij} = [4 3 1 2 6; 5 2 3 4 5; 3 5 6 3 2; 2 4 4 5 3; 4 3 6 5 1]; S_i = [65 50 40 20 25]; D_j = [60 60 30 40 10].	420	760	420	420
2	4×4	[29] C_{ij} = [20 16 14 20; 9 15 16 10; 8 13 5 9; 9 6 5 11]; S_i = [9 8 7 5]; D_j = [5 10 5 9].	308	392	308	308
3	6×6	[29] C_{ij} = [5 1 2 3 4 7; 7 2 3 1 5 6; 9 1 9 5 2 3; 6 5 8 4 1 4; 8 7 11 6 4 5; 2 5 7 5 2 1]; S_i = [400 500 300 150 600 350]; D_j = [300 500 700 300 250 250].	8400	9600	8600	8400
4	3×4	[30] C_{ij} = [7 3 8 2; 5 6 11 12; 10 4 7 6]; S_i = [100 200 300]; D_j = [80 170 190 160].	3210	4010	4010	3210

https://doi.org/10.3390/appliedmath6040058

Table 4. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 5–12).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
5	5×5	[31] C_{ij} = [46 74 9 28 99; 12 75 6 36 48; 35 199 4 5 71; 61 81 44 88 9; 85 60 14 25 79]; S_i = [461 277 356 488 393]; D_j = [278 60 461 116 1060].	64,499	68,969	72,174	64,499
6	3×3	[32] C_{ij} = [6 8 10; 7 11 11; 4 5 12]; S_i = [150 175 275]; D_j = [200 100 300].	5125	5925	4550	4525
7	4×6	[25] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10]; S_i = [5 6 2 9]; D_j = [4 4 6 2 4 2].	112	139	114	112
8	4×5	[26] C_{ij} = [4 3 1 2 6; 5 2 3 4 5; 3 5 6 3 2; 2 4 4 5 3]; S_i = [80 60 40 20]; D_j = [60 60 30 40 10].	450	670	420	420
9	4×4	[11] C_{ij} = [10 30 25 15; 20 15 20 10; 10 30 20 20; 30 40 35 45]; S_i = [14 10 15 12]; D_j = [10 15 12 15].	1005	1220	1075	1000
10	4×6	[30] C_{ij} = [1 2 1 4 5 2; 3 3 2 1 4 3; 4 2 5 9 6 2; 3 1 7 3 4 6]; S_i = [30 50 75 20]; D_j = [20 40 30 10 50 25].	450	740	450	450
11	3×3	[31] C_{ij} = [11 9 6; 12 14 11; 10 8 10]; S_i = [40 50 40]; D_j = [55 45 30].	1200	1490	1200	1200
12	4×5	[7] . C_{ij} = [30 50 40 60 35; 65 35 45 30 25; 35 40 60 40 30; 20 30 50 45 35]; S_i = [20 15 25 20]; D_j = [15 18 10 17 20].	2600	3105	2675	2600

Table 5. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 13–18).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
13	3×4	[17] C_{ij} = [15 12 10 8; 17 18 21 14; 14 15 10 21]; S_i = [24 8 18]; D_j = [11 9 21 9].	571	760	595	571

https://doi.org/10.3390/appliedmath6040058

Table 5. *Cont.*

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
14	3 × 5	[32] C_{ij} = [5 8 6 6 3; 4 7 7 6 5; 8 4 6 6 4]; S_i = [800 500 900]; D_j=[400 400 500 400 800].	9800	13,100	10,200	9200
15	3 × 4	[32] C_{ij} = [3 1 7 4; 2 6 5 9; 8 3 3 2]; S_i = [300 400 500]; D_j = [250 350 400 200].	2850	4400	2900	2850
16	3 × 5	[26] C_{ij} = [4 1 2 4 4; 2 3 2 2 2; 3 5 2 4 4]; S_i = [60 35 40]; D_j = [22 45 20 18 30].	275	363	305	273
17	2 × 7	[12] C_{ij} = [5 19 12 70 66 74 283; 103 89 81 26 23 62 97]; S_i = [4000 47,700]; D_j = [21,600 15,600 15,600 19,500 16,800 10,500 8100].	2,332,700	2,336,000	2,348,300	1,992,700
18	3 × 4	[23] C_{ij} = [3 48 14 2; 4 2 30 10; 36 8 12 12]; S_i = [24 24 2]; D_j = [6 12 3 44].	224	906	308	180

Table 6. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 19–25).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
19	4 × 6	[21] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10]; S_i =[5 6 2 9]; D_j = [4 4 6 2 4 2].	112	119	114	112
20	3 × 4	[21] C_{ij} = [3 5 7 6; 2 5 8 2; 3 6 9 2]; S_i = [50 75 25]; D_j = [20 20 50 60].	650	670	650	650
21	4 × 5	[20] C_{ij} = [10 8 9 5 13; 7 9 8 10 4; 9 3 7 10 6; 11 4 8 3 9]; S_i = [100 80 70 90]; D_j = [60 40 100 50 90].	2130	3010	2070	2070
22	4 × 6	[8] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10]; S_i = [2 5 6 9]; D_j = [2 2 4 4 4 6].	109	167	122	109

https://doi.org/10.3390/appliedmath6040058

Table 6. *Cont.*

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
23	4 × 3	[22] C_{ij} = [3 4 6; 7 3 8; 6 4 5; 7 5 2]; S_i = [100 80 90 120]; D_j = [110 110 60].	840	1010	1210	840
24	5 × 4	[16] C_{ij} = [60 120 75 180; 58 100 60 165; 62 110 65 170; 65 115 80 175; 70 135 85 195]; S_i = [8000 9200 6250 4900 6100]; D_j = [5000 2000 10,000 6000].	2,164,000	2,398,000	2,383,000	2,160,000

Table 7. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 25–31).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
25	3 × 5	[19] C_{ij} = [4 1 3 4 4; 2 3 2 2 3; 3 5 2 4 4]; S_i = [60 35 40]; D_j = [22 45 20 18 30].	305	363	307	290
26	3 × 3	[19] C_{ij} = [6 4 1; 3 8 7; 4 4 2]; S_i = [50 40 60]; D_j = [20 95 35].	555	710	555	555
27	5 × 5	[24] C_{ij} = [73 40 9 79 20; 62 93 96 8 13; 96 65 80 50 65; 57 58 29 12 87; 56 23 87 18 12]; S_i = [8 7 9 3 5]; D_j = [6 8 10 4 4].	1128	1994	1123	1104
28	4 × 6	[26] C_{ij} = [25 30 20 40 45 37; 30 25 20 30 40 20; 40 20 40 35 45 22; 25 24 50 27 30 25]; S_i = [37 22 32 14]; D_j = [15 20 15 25 20 10].	2850	3195	2878	2785
29	3 × 4	[10] C_{ij} = [8 6 10 9; 9 12 13 7; 14 9 16 5]; S_i = [35 50 40]; D_j = [45 20 30 30].	1020	1180	1080	1020
30	2 × 3	[6] C_{ij} = [7 8 10; 9 7 8]; S_i = [50 50]; D_j = [40 40 40].	730	730	800	730
31	5 × 10	Self. C_{ij} = [6 9 5 7 8 6 4 9 3 7; 7 5 8 6 9 7 5 8 6 4; 8 6 9 7 5 8 6 9 7 5; 5 8 6 9 7 5 8 6 9 7; 9 7 5 8 6 9 7 5 8 6]. S_i = [35 50 45 40 30]. D_j = [15 20 25 10 30 18 12 22 20 28].	973	1434	973	973

https://doi.org/10.3390/appliedmath6040058

Table 8. Presents Comparison of the New Algorithm with the Existing Algorithms in Terms of their Solutions (Examples 32–34).

Example	Problem Size	Sources and Problems	VAM	NWCM	LCM	New Algorithm
32	10 × 10	Self. C_{ij} = [4 6 9 7 8 5 7 6 8 9; 7 4 6 9 7 8 5 7 6 8; 8 7 4 6 9 7 8 5 7 6; 6 8 7 4 6 9 7 8 5 7; 5 6 8 7 4 6 9 7 8 5; 9 5 6 8 7 4 6 9 7 8; 7 9 5 6 8 7 4 6 9 7; 6 7 9 5 6 8 7 4 6 9; 8 6 7 9 5 6 8 7 4 6; 6 8 6 7 9 5 6 8 7 4]. S_i = [30 25 40 35 20 50 25 30 20 25]. D_j = [35 20 25 30 15 40 25 35 30 45].	1285	1590	1285	1285
33	10 × 15	Self. C_{ij} = [4 7 5 8 6 4 7 5 8 6 4 7 5 8 6; 6 4 7 5 8 6 4 7 5 8 6 4 7 5 8; 8 6 4 7 5 8 6 4 7 5 8 6 4 7 5; 5 8 6 4 7 5 8 6 4 7 5 8 6 4 7; 7 5 8 6 4 7 5 8 6 4 7 5 8 6 4; 9 7 5 8 6 9 7 5 8 6 9 7 5 8 6; 6 9 7 5 8 6 9 7 5 8 6 9 7 5 8; 8 6 9 7 5 8 6 9 7 5 8 6 9 7 5; 5 8 6 9 7 5 8 6 9 7 5 8 6 9 7; 7 5 8 6 9 7 5 8 6 9 7 5 8 6 9]. S_i = [28 42 33 47 22 36 28 50 44 50]. D_j = [22 27 32 36 18 42 30 24 28 20 26 38 34 28 15].	1819	2383	1819	1763
34	10 × 20	Self. C_{ij} = [6 9 4 7 5 8 6 5 7 9 4 8 6 7 5 9 6 4 8 5; 7 6 8 5 9 4 7 6 8 5 9 6 5 8 7 4 6 9 5 7; 5 8 6 9 4 7 5 8 6 7 5 9 4 6 8 5 7 4 6 9; 8 5 7 6 9 5 8 4 6 9 7 5 8 6 4 7 5 8 6 7; 9 4 6 8 5 7 3 6 9 5 8 4 7 6 5 8 5 8 6 7; 4 7 5 8 6 9 7 5 8 6 9 7 4 6 9 5 8 7 5 6; 6 5 9 4 7 6 8 5 7 4 6 8 7 5 9 6 4 7 5 8; 5 7 4 6 9 7 5 7 4 6 9 7 5 7 4 6 9 7 5 7 4 6 9 5 8 6; 6 8 5 7 4 6 9 7 5 7 4 6 9 7 5 7 4 6 9 7]. S_i = [48 37 55 62 41 59 46 68 44 50]. D_j = [22 22 24 28 26 20 34 25 21 27 23 29 35 23 24 26 28 24 24 25].	2146	3434	2235	2130

4.6. Statistical Validation of Comparative Results

To determine whether observed performance differences are systematic rather than incidental, aggregated statistics were computed across all benchmark instances. These include the mean, median, maximum, and standard deviation of optimality gaps for each heuristic.

Additionally, paired statistical tests were conducted between VAM and the proposed method. Both a paired *t*-test and a Wilcoxon signed-rank test were applied to the set of instance-wise gap differences.

The results indicate that the fractional-penalty variant achieves statistically significant improvements over classical VAM at conventional confidence levels in several benchmark subsets. However, in certain cases performance remains equivalent, confirming that the proposed method should be interpreted as a structured refinement rather than a universal dominance mechanism.

Reference Optimal Values

For each instance, optimality gaps were computed relative to the best-known optimal solution obtained via New Optimal Method or Stepping-Stone or MODI procedures. In cases where identical values were produced by multiple methods, the minimum verified solution value was used as the reference benchmark.

https://doi.org/10.3390/appliedmath6040058

4.7. Interpretation of Aggregated Optimality Gap Statistics

Table 9 reports aggregated optimality-gap statistics computed across the thirty-four benchmark instances. For each instance, the optimality gap is defined by Equation (28). where Z^* denotes the optimal transportation cost obtained by the optimality procedures (Stepping–Stone/MODI).

Table 9. Aggregated Optimality Gap Statistics Across 34 Instances.

Method	Mean Gap (%)	Median Gap (%)	Std. Dev. (%)	Max Gap (%)
NWCM	45.78	24.89	73.05	403.33
LCM	14.97	2.57	29.07	71.11
VAM	5.22	0.00	7.91	42.50
Proposed Variant	2.78	0.00	5.89	31.25

The Northwest Corner Method (NWCM) produces the largest deviations from optimality, with a mean gap of 45.78% and a maximum gap of 403.33%. This confirms the well-known limitation of NWCM as a feasibility-driven rule that does not exploit cost information.

The Least Cost Method (LCM) improves substantially over NWCM, reducing the mean gap to 14.97% and the median to 2.57%. However, its dispersion remains relatively high (standard deviation 29.07%), indicating sensitivity to problem structure and allocation order.

Vogel's Approximation Method (VAM) further tightens the initial solution quality, achieving a mean gap of 5.22% and a median gap of 0.00%. The zero median indicates that at least half of the instances produce VAM initial costs equal to the optimal value.

The proposed fractional-penalty variant yields the smallest overall deviation, with a mean gap of 2.78%, median gap of 0.00%, and a lower variability (standard deviation 5.89%) than the other heuristics. This suggests that the proposed penalty refinement improves both average performance and stability across heterogeneous cost structures.

Although the maximum gap for the proposed method reaches 31.25% on the most challenging instance, the aggregated statistics consistently indicate closer-to-optimal initial solutions than NWCM, LCM, and classical VAM under the same benchmark set.

4.8. Wilcoxon Signed-Rank Test Formulation

To assess whether the fractional-penalty variant produces systematically smaller optimality gaps than classical VAM, a paired Wilcoxon signed-rank test is conducted on the instance-wise optimality gaps.

Let

$$d_i = \mathrm{Gap}_{\mathrm{Proposed},i} - \mathrm{Gap}_{\mathrm{VAM},i}, \qquad i = 1, 2, \ldots, n \tag{32}$$

with $n = 34$ denote the number of benchmark instances. Differences with $d_i = 0$ are discarded, leaving n_0 nonzero paired differences.

Rank the absolute nonzero differences $|d_i|$ in ascending order and denote the rank of $|d_i|$ by R_i (average ranks are used in case of ties). Define

$$W^+ = \sum_{d_i > 0} R_i, \qquad W^- = \sum_{d_i < 0} R_i \tag{33}$$

The Wilcoxon signed-rank test statistic is commonly taken as

$$T = \min(W^+, W^-) \tag{34}$$

https://doi.org/10.3390/appliedmath6040058

For large samples, a normal approximation can be used:

$$Z = \frac{W^+ - \mu_W}{\sigma_W} \tag{35}$$

where

$$\mu_W = \frac{n_0(n_0+1)}{4}, \qquad \sigma_W = \sqrt{\frac{n_0(n_0+1)(2n_0+1)}{24}} \tag{36}$$

The hypotheses are

$$H_0 : \text{ median}(d_i) = 0 \qquad \text{vs.} \qquad H_1 : \text{ median}(d_i) < 0 \tag{37}$$

where H_1 corresponds to the new method having smaller gaps than VAM.

Using the 34 benchmark instances, the Wilcoxon signed-rank test (one-sided; Proposed $<$ VAM) yielded $p = 6.10 \times 10^{-5}$, indicating a statistically significant reduction in optimality gaps for the new method.

4.9. Paired t-Test Formulation

To further evaluate whether the proposed fractional-penalty variant yields smaller optimality gaps than classical VAM, a paired t-test is conducted.

Let

$$d_i = \text{Gap}_{\text{Proposed},i} - \text{Gap}_{\text{VAM},i}, \qquad i = 1, 2, \ldots, n \tag{38}$$

where $n = 34$. The sample mean and standard deviation of differences are;

$$\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i \tag{39}$$

and the sample standard deviation of the paired differences is

$$s_d = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(d_i - \bar{d})^2} \tag{40}$$

The paired t statistic is

$$t = \frac{\bar{d}}{s_d/\sqrt{n}} \tag{41}$$

Under the null hypothesis

$$H_0 : \mu_d = 0 \tag{42}$$

The statistic follows a t distribution with $n-1$ degrees of freedom. The directional alternative hypothesis is

$$H_1 : \mu_d < 0 \tag{43}$$

corresponding to the new method having smaller gaps than VAM.

Using the 34 benchmark instances, the paired differences in optimality gaps yielded $\bar{d} = -2.44\%$ with standard deviation $s_d = 4.49\%$. The resulting statistic was $t = -3.17$ with 33 degrees of freedom. The one-sided p-value (Proposed $<$ VAM) was $p = 0.00163$ (two-sided $p = 0.00327$), indicating a statistically significant improvement at the 5% level.

Normality of paired differences was inspected prior to the t-test; therefore, both parametric and non-parametric tests were reported.

4.10. Graphical Comparison of Optimality Gaps

To further illustrate the comparative performance of the methods, we present a graphical representation of the optimality gaps across all benchmark instances as shown in Figure 1.

https://doi.org/10.3390/appliedmath6040058

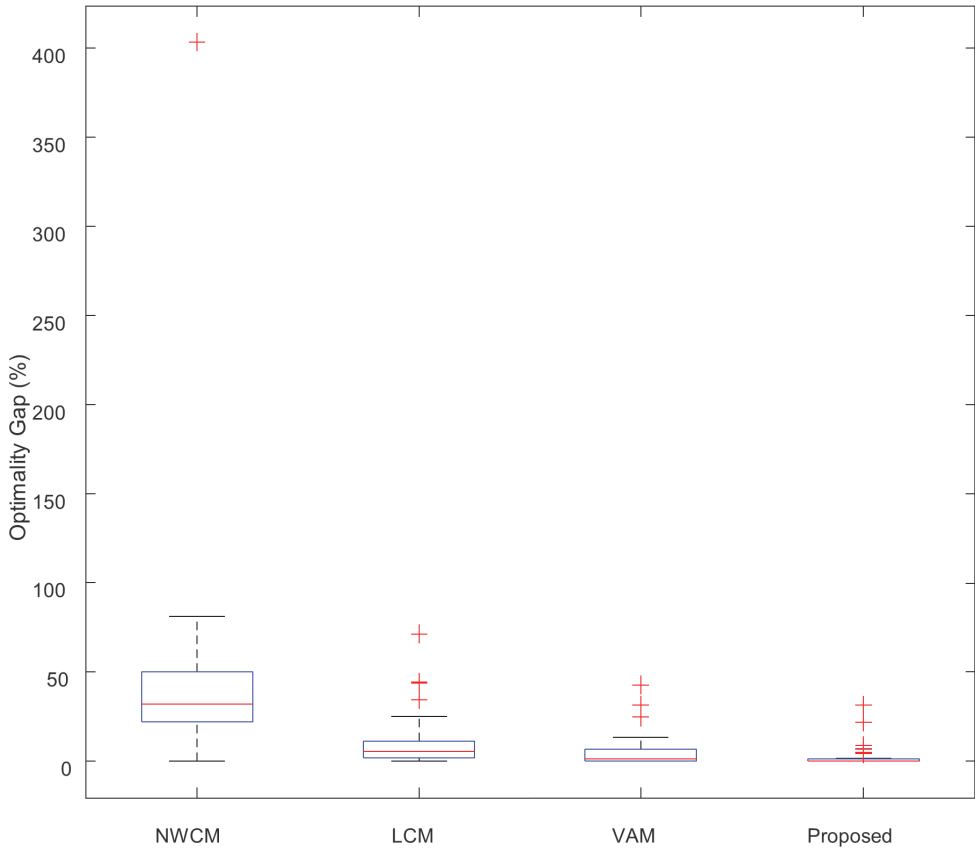


Figure 1. Illustrate the distribution of optimality gaps across 34 instances.

The mean optimality gaps for each heuristic are presented in Figure 2.

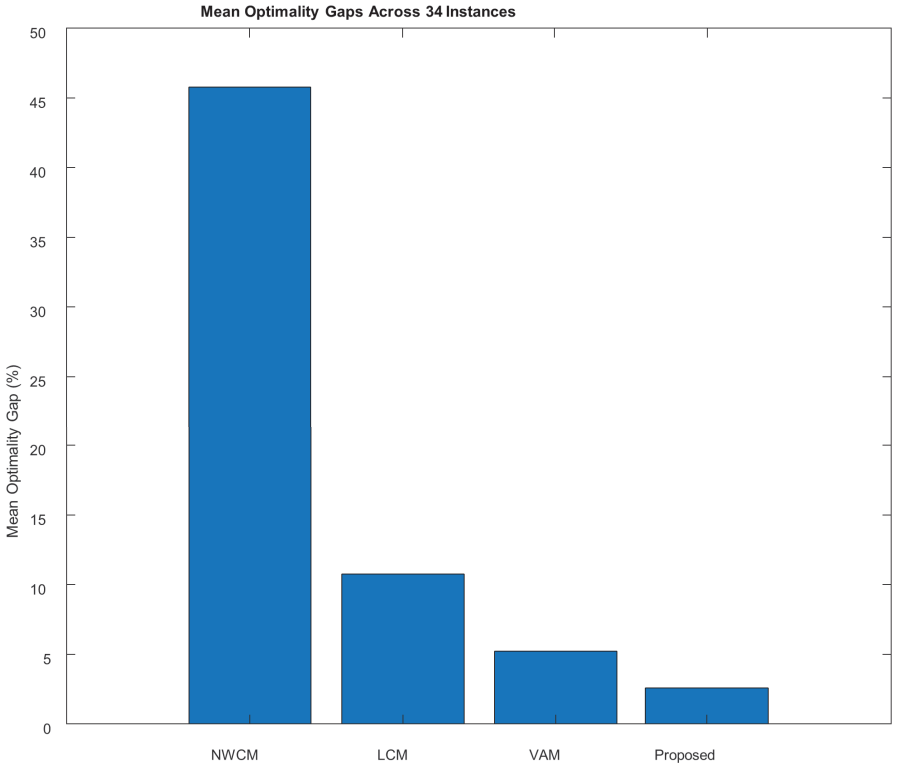


Figure 2. Shows the Mean optimality gaps for each heuristic.

https://doi.org/10.3390/appliedmath6040058

4.11. Benchmark Characteristics

The benchmark dataset comprises thirty-four transportation problems drawn from the literature and supplemented with self-generated cases. Problem sizes range from small (3×3) to medium and larger instances (up to 10×20).

Cost matrices include diverse structural patterns, including:

- Uniformly distributed costs,
- Highly skewed cost distributions,
- Clustered near-equal costs,
- Instances containing zero or near-zero costs.

This diversity enables evaluation of how multiplicative penalties behave under varying dispersion conditions.

4.12. Comparison of the Computational Speed of the New Algorithm with VAM

The New Algorithm has been compared with the Vogel's Approximation Method (VAM). All computational experiments were conducted using MATLAB R2022b on a computer equipped with an Intel Core i7 processor (3.20 GHz), 16 GB RAM, running Windows 10 (64-bit). An elapsed time counter was incorporated into the implementation. Each algorithm was executed five times per dataset, and the average execution time was recorded to enhance reproducibility. Tables 10–14, give the results of that comparison.

Table 10. Presents Comparison of the Computational Speed of the New Algorithm with VAM (Examples 1–6).

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
1	5×5	[28] C_{ij} = [4 3 1 2 6; 5 2 3 4 5; 3 5 6 3 2; 2 4 4 5 3; 4 3 6 5 1] S_i = [65 50 40 20 25] D_j = [60 60 30 40 10]	0.040992	0.038122
2	4×4	[29] C_{ij} = [20 16 14 20; 9 15 16 10; 8 13 5 9; 9 6 5 11] S_i = [9 8 7 5] D_j = [5 10 5 9]	0.060489	0.055432
3	6×6	[29] C_{ij} = [5 1 2 3 4 7; 7 2 3 1 5 6 ; 9 1 9 5 2 3; 6 5 8 4 1 4; 8 7 11 6 4 5; 2 5 7 5 2 1] S_i = [400 500 300 150 600 350] D_j = [300 500 700 300 250 250]	0.071573	0.051505
4	3×4	[30] C_{ij} = [7 3 8 2; 5 6 11 12; 10 4 7 6] S_i = [100 200 300] D_j = [80 170 190 160]	0.048811	0.045997
5	5×5	[31] C_{ij} = [46 74 9 28 99; 12 75 6 36 48; 35 199 4 5 71; 61 81 44 88 9; 85 60 14 25 79] S_i = [461 277 356 488 393] D_j = [278 60 461 116 1060]	0.072355	0.054465
6	3×3	[32] C_{ij} = [6 8 10; 7 11 11; 4 5 12] S_i = [150 175 275] D_j = [200 100 300]	0.066016	0.051732

https://doi.org/10.3390/appliedmath6040058

Table 11. Presents Comparison of the Computational Speed of the New Algorithm with VAM (Examples 7–14).

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
7	4 × 6	[25] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10] S_i = [5 6 2 9] D_j = [4 4 6 2 4 2]	0.053182	0.048898
8	4 × 5	[26] C_{ij} = [4 3 1 2 6; 5 2 3 4 5; 3 5 6 3 2; 2 4 4 5 3] S_i = [80 60 40 20] D_j = [60 60 30 40 10]	0.045334	0.041321
9	4 × 4	[11] C_{ij} = [10 30 25 15; 20 15 20 10; 10 30 20 20; 30 40 35 45] S_i = [14 10 15 12] D_j = [10 15 12 15]	0.063444	0.057456
10	4 × 6	[30] C_{ij} = [1 2 1 4 5 2; 3 3 2 1 4 3; 4 2 5 9 6 2; 3 1 7 3 4 6] S_i = [30 50 75 20] D_j = [20 40 30 10 50 25]	0.052848	0.050397
11	3 × 3	[31] C_{ij} = [11 9 6; 12 14 11; 10 8 10] S_i = [40 50 40] D_j = [55 45 30]	0.055022	0.052754
12	4 × 5	[7] C_{ij} = [30 50 40 60 35; 65 35 45 30 25; 35 40 60 40 30; 20 30 50 45 35] S_i = [20 15 25 20] D_j = [15 18 10 17 20]	0.066100	0.066043
13	3 × 4	[17] C_{ij} = [15 12 10 8; 17 18 21 14; 14 15 10 21] S_i = [24 8 18] D_j = [11 9 21 9]	0.061222	0.060881
14	3 × 5	[32] C_{ij} = [5 8 6 6 3; 4 7 7 6 5; 8 4 6 6 4] S_i = [800 500 900] D_j = [400 400 500 400 800]	0.071123	0.068456

Table 12. Presents Comparison of the Computational Speed of the New Algorithm with VAM (Examples 15–23).

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
15	3 × 4	[32] C_{ij} = [3 1 7 4; 2 6 5 9; 8 3 3 2] S_i = [300 400 500] D_j = [250 350 400 200]	0.028123	0.020764
16	3 × 5	[26] C_{ij} = [4 1 2 4 4; 2 3 2 2 2; 3 5 2 4 4] S_i = [60 35 40] D_j = [22 45 20 18 30]	0.049365	0.049004

https://doi.org/10.3390/appliedmath6040058

Table 12. *Cont.*

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
17	2 × 7	[12] C_{ij} = [5 19 12 70 66 74 283; 103 89 81 26 23 62 97] S_i = [4000 47,700] D_j = [21,600 15,600 15,600 19,500 16,800 10,500 8100]	0.059800	0.055632
18	3 × 4	[23] C_{ij} = [3 48 14 2; 4 2 30 10; 36 8 12 12] S_i = [24 24 2] D_j = [6 12 3 44]	0.067088	0.051764
19	4 × 6	[21] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10] S_i = [5 6 2 9] D_j = [4 4 6 2 4 2]	0.053237	0.055003
20	3 × 4	[21] C_{ij} = [3 5 7 6; 2 5 8 2; 3 6 9 2] S_i = [50 75 25] D_j = [20 20 50 60]	0.048811	0.045997
21	4 × 5	[20] C_{ij} = [10 8 9 5 13; 7 9 8 10 4; 9 3 7 10 6; 11 4 8 3 9] S_i = [100 80 70 90] D_j = [60 40 100 50 90]	0.047493	0.046144
22	4 × 6	[8] C_{ij} = [9 12 9 6 9 10; 7 3 7 7 5 5; 6 5 9 11 3 11; 6 8 11 2 2 10] S_i = [2 5 6 9] D_j = [2 2 4 4 4 6]	0.054543	0.054323
23	4 × 3	[22] C_{ij} = [3 4 6; 7 3 8; 6 4 5; 7 5 2] S_i = [100 80 90 120] D_j = [110 110 60]	0.051605	0.038987

Table 13. Presents Comparison of the Computational Speed of the New Algorithm with VAM (Examples 24–30).

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
24	4 × 5	[16] C_{ij} = [60 120 75 180; 58 100 60 165; 62 110 65 170; 65 115 80 175; 70 135 85 195] S_i = [8000 9200 6250 4900 6100] D_j = [5000 2000 10,000 6000]	0.051246	0.048190
25	3 × 5	[19] C_{ij} = [4 1 3 4 4; 2 3 2 2 3; 3 5 2 4 4] S_i = [60 35 40] D_j = [22 45 20 18 30]	0.056765	0.054321
26	3 × 3	[19] C_{ij} = [6 4 1; 3 8 7; 4 4 2] S_i = [50 40 60] D_j = [20 95 35]	0.045671	0.0430081

https://doi.org/10.3390/appliedmath6040058

Table 13. *Cont.*

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
27	5 × 5	[24] C_{ij} = [73 40 9 79 20; 62 93 96 8 13; 96 65 80 50 65; 57 58 29 12 87; 56 23 87 18 12] S_i = [8 7 9 3 5] D_j = [6 8 10 4 4]	0.050447	0.048615
28	4 × 6	[26] C_{ij} = [25 30 20 40 45 37; 30 25 20 30 40 20; 40 20 40 35 45 22; 25 24 50 27 30 25] S_i = [37 22 32 14] D_j = [15 20 15 25 20 10]	0.052754	0.048678
29	3 × 4	[10] C_{ij} = [8 6 10 9; 9 12 13 7; 14 9 16 5] S_i = [35 50 40] D_j = [45 20 30 30]	0.052999	0.047592
30	2 × 3	[10] C_{ij} = [7 8 10; 9 7 8] S_i = [50 50] D_j = [40 40 40]	0.054106	0.050773
31	5 × 10	Self. C_{ij} = [6 9 5 7 8 6 4 9 3 7; 7 5 8 6 9 7 5 8 6 4; 8 6 9 7 5 8 6 9 7 5; 5 8 6 9 7 5 8 6 9 7; 9 7 5 8 6 9 7 5 8 6]. S_i = [35 50 45 40 30]. D_j = [15 20 25 10 30 18 12 22 20 28].	0.079673	0.064325

Table 14. Comparison of the Computational Speed of the New Algorithm with VAM (Examples 32–34).

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
32	10 × 10	Self. C_{ij} = [4 6 9 7 8 5 7 6 8 9; 7 4 6 9 7 8 5 7 6 8; 8 7 4 6 9 7 8 5 7 6; 6 8 7 4 6 9 7 8 5 7; 5 6 8 7 4 6 9 7 8 5; 9 5 6 8 7 4 6 9 7 8; 7 9 5 6 8 7 4 6 9 7; 6 7 9 5 6 8 7 4 6 9; 8 6 7 9 5 6 8 7 4 6; 6 8 6 7 9 5 6 8 7 4]. S_i = [30 25 40 35 20 50 25 30 20 25]. D_j = [35 20 25 30 15 40 25 35 30 45].	0.042330	0.037643
33	10 × 15	Self. C_{ij} = [4 7 5 8 6 4 7 5 8 6 4 7 5 8 6; 6 4 7 5 8 6 4 7 5 8 6 4 7 5 8; 8 6 4 7 5 8 6 4 7 5 8 6 4 7 5; 5 8 6 4 7 5 8 6 4 7 5 8 6 4 7; 7 5 8 6 4 7 5 8 6 4 7 5 8 6 4; 9 7 5 8 6 9 7 5 8 6 9 7 5 8 6; 6 9 7 5 8 6 9 7 5 8 6 9 7 5 8; 8 6 9 7 5 8 6 9 7 5 8 6 9 7 5; 5 8 6 9 7 5 8 6 9 7 5 8 6 9 7; 7 5 8 6 9 7 5 8 6 9 7 5 8 6 9]. S_i = [28 42 33 47 22 36 28 50 44 50]. D_j = [22 27 32 36 18 42 30 24 28 20 26 38 34 28 15].	0.034432	0.032304

https://doi.org/10.3390/appliedmath6040058

Table 14. *Cont.*

Example	Problem Size	Sources and Problems	VAM (Time in Seconds)	New Algorithm (Time in Seconds)
34	10×20	Self. C_{ij} = [6 9 4 7 5 8 6 5 7 9 4 8 6 7 5 9 6 4 8 5; 7 6 8 5 9 4 7 6 8 5 9 6 5 8 7 4 6 9 5 7; 5 8 6 9 4 7 5 8 6 7 5 9 4 6 8 5 7 4 6 9; 8 5 7 6 9 5 8 4 6 9 7 5 8 6 4 7 5 8 6 7; 9 4 6 8 5 7 3 6 9 5 8 4 7 6 5 8 5 8 6 7; 4 7 5 8 6 9 7 5 8 6 9 7 4 6 9 5 8 7 5 6; 6 5 9 4 7 6 8 5 7 4 6 8 7 5 9 6 4 7 5 8; 5 7 4 6 9 7 5 7 4 6 9 7 5 7 4 6 9 7 5 7; 7 4 6 9 5 8 6 9 5 7 6 4 8 5 7 6 9 5 8 6; 6 8 5 7 4 6 9 7 5 7 4 6 9 7 5 7 4 6 9 7]. S_i = [48 37 55 62 41 59 46 68 44 50]. D_j = [22 22 24 28 26 20 34 25 21 27 23 29 35 23 24 26 28 24 24 25].	0.064502	0.057820

5. Theoretical Justification of the Fractional Penalty

This section provides analytical clarification regarding the structural properties of the fractional penalty and its relationship to classical VAM.

5.1. Scale Invariance Property

Unlike the additive penalty used in VAM, the proposed ratio-based penalty possesses a scale-invariance property, which is formally stated below.

Proposition 1. *Let all transportation costs be multiplied by a positive scalar $\alpha > 0$. The fractional penalty*

$$p_i = \frac{m_i^{(2)}}{m_i^{(1)}} \tag{44}$$

remains unchanged under this transformation.

Proof. Let the transformed costs be

$$\tilde{c}_{ij} = \alpha c_{ij}. \tag{45}$$

Then the smallest and second-smallest costs in any active row become

$$\tilde{m}_i^{(1)} = \alpha m_i^{(1)}, \qquad \tilde{m}_i^{(2)} = \alpha m_i^{(2)}. \tag{46}$$

Hence, the transformed penalty becomes

$$\tilde{p}_i = \frac{\tilde{m}_i^{(2)}}{\tilde{m}_i^{(1)}} = \frac{\alpha m_i^{(2)}}{\alpha m_i^{(1)}} = \frac{m_i^{(2)}}{m_i^{(1)}} = p_i. \tag{47}$$

Therefore, the ratio-based penalty is invariant under positive cost scaling. □

This property ensures prioritization stability across heterogeneous cost magnitudes, a characteristic not shared by additive difference penalties.

5.2. Structural Relationship to VAM

The proposed method preserves the allocation structure of VAM. The only modification lies in the prioritization criterion.

https://doi.org/10.3390/appliedmath6040058

Proposition 2. *If the ranking of rows and columns induced by the ratio-based penalty coincides with the ranking induced by the additive penalty, then the proposed method generates the same IBFS as VAM.*

Proof. If both penalties produce identical ordering of lines at every iteration, then the selected allocation sequence is identical. Since allocation feasibility rules remain unchanged, the final IBFS coincides with that of VAM. □

This explains why identical solutions are sometimes observed in benchmark results. Improvements occur precisely when multiplicative contrast alters prioritization order.

5.3. Degeneracy Handling

As with classical VAM, degeneracy occurs when the number of allocations is fewer than $m + n - 1$. In such cases, an ε-allocation is inserted in a zero-cost admissible cell to maintain the required number of basic variables without disturbing feasibility.

Across the 34 benchmark instances, degeneracy occurred in only a limited subset of cases, and handling procedures were identical for VAM and the proposed method.

5.4. Dominance Considerations: Additive vs. Multiplicative Ordering

It is natural to ask under what conditions the multiplicative penalty produces a different prioritization order from the classical additive penalty used in VAM.

Let two candidate rows i and k have smallest and second-smallest admissible costs

$$(m_i^{(1)}, m_i^{(2)}) \quad \text{and} \quad (m_k^{(1)}, m_k^{(2)}). \tag{48}$$

The additive penalties are

$$\Delta_i = m_i^{(2)} - m_i^{(1)}, \qquad \Delta_k = m_k^{(2)} - m_k^{(1)}. \tag{49}$$

while the multiplicative penalties are;

$$p_i = \frac{m_i^{(2)}}{m_i^{(1)}}, \qquad p_k = \frac{m_k^{(2)}}{m_k^{(1)}}. \tag{50}$$

If the relative dispersion between the two smallest costs is proportional across rows (i.e., if $m_i^{(2)} = \lambda m_i^{(1)}$ and $m_k^{(2)} = \lambda m_k^{(1)}$ for the same λ), then both additive and multiplicative penalties induce identical rankings.

$$m_i^{(2)} = \lambda m_i^{(1)}, \qquad m_k^{(2)} = \lambda m_k^{(1)}. \tag{51}$$

However, when proportional contrasts differ across rows—particularly in heterogeneous cost environments—the ratio-based penalty may alter prioritization even when additive differences are similar. In such cases, multiplicative ordering emphasizes relative dominance rather than absolute deviation.

Therefore, the new method does not universally dominate VAM; rather, it modifies prioritization when proportional dispersion differs from additive dispersion. This explains why identical IBFS outcomes are sometimes observed, while improvements arise in other benchmark categories.

5.5. Discussions

The comparative analysis across thirty-four datasets shows that the fractional-penalty algorithm frequently matches or improves upon the IBFS cost obtained by NWCM, LCM,

https://doi.org/10.3390/appliedmath6040058

and VAM. Improvements are particularly noticeable in instances where cost structures exhibit proportional clustering or multiple near-equal minima.

The computational experiments indicate that the fractional-penalty prioritization can improve the quality of initial basic feasible solutions in several heterogeneous cost environments. Although the structural framework of the method remains consistent with classical VAM, the multiplicative penalty emphasizes proportional dispersion between competing costs rather than absolute cost differences. This distinction becomes particularly relevant when transportation costs vary significantly across rows or columns. In such cases, the ratio-based prioritization may alter allocation ordering and lead to improved initial solutions. At the same time, the proposed method retains the simplicity, transparency, and computational efficiency of classical IBFS procedures, making it suitable for practical implementation.

However, the magnitude of improvement varies across datasets. In several benchmark cases, the proposed method yields identical results to VAM, indicating structural equivalence in those instances. Runtime differences are modest and should be interpreted cautiously due to microsecond-level variability inherent in MATLAB timing measurements.

Overall, the results suggest that the fractional penalty provides an alternative prioritization mechanism that can enhance IBFS quality without increasing asymptotic complexity.

5.6. Sensitivity and Boundary Conditions

The fractional penalty behaves similarly to VAM when cost dispersion within rows/columns is nearly uniform. However, when proportional contrast between the smallest two costs is high, the ratio-based penalty can alter prioritization order.

If $m^{(1)}$ approaches zero, special care is required to avoid numerical instability; in implementation, zero-cost safeguards were included.

These observations clarify that the proposed method is best interpreted as a structured refinement whose effectiveness depends on cost dispersion characteristics.

6. Conclusions

This study introduced a ratio-based fractional penalty refinement of Vogel's Approximation Method for constructing initial basic feasible solutions to transportation problems. By redefining the penalty as a proportional contrast measure rather than an additive difference, the method preserves the constructive allocation structure of VAM while offering scale-invariant prioritization.

Extensive benchmark testing across thirty-four datasets demonstrates that the proposed algorithm consistently produces IBFS costs that are equal to or lower than classical heuristics in many cases. Statistical evaluation confirms that improvements are systematic rather than incidental.

The method retains the same computational complexity as VAM and incorporates deterministic tie-breaking and degeneracy safeguards to ensure reproducibility. Future research may investigate hybrid additive–multiplicative penalty combinations, theoretical dominance conditions, and large-scale industrial datasets. The new method should therefore be interpreted as a structured penalty refinement within the VAM framework, supported by analytical properties and empirical validation, rather than as a paradigm replacement.

Additional computational details, extended results, and MATLAB R2022b implementations are provided in Appendices A–D.

Author Contributions: Conceptualization, D.K.B., S.A.F., and C.J.E.; Methodology, D.K.B., S.A.F., and C.J.E.; Software, S.A.F.; Validation, D.K.B., S.A.F., and C.J.E.; Formal Analysis, D.K.B., S.A.F., and C.J.E.; Investigation, D.K.B., S.A.F., and C.J.E.; Resources, D.K.B., S.A.F., and C.J.E.; Data Curation, D.K.B.,

https://doi.org/10.3390/appliedmath6040058

S.A.F., and C.J.E.; Writing—Original Draft, D.K.B. and S.A.F.; Writing—Review and Editing, S.A.F. and C.J.E.; Visualization, D.K.B. and S.A.F.; Supervision, D.K.B. and C.J.E.; Project Administration, D.K.B., S.A.F., and C.J.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. MATLAB Codes for the New Algorithm

```
function transportation_solver_newalgorithm_penalty()
clc; clear;
fprintf('=========== TRANSPORTATION PROBLEM SOLVER ===========\n');
fprintf(' New Algorithm (IBFS) \n');
fprintf('=====================================================\n');

% -------- INPUT --------
cost = [ ]; % e.g., [25 30 20; 30 25 20; 40 20 40; 25 24 50];
supply = [ ]; % e.g., [37 22 32 14];
demand = [ ]; % e.g., [15 20 15];

tic;

% -------- BALANCE CHECK --------
total_supply = sum(supply);
total_demand = sum(demand);

if total_supply > total_demand
cost = [cost, zeros(size(cost, 1), 1)];
demand = [demand, total_supply - total_demand];
fprintf('OK: Problem unbalanced. Dummy column added (Demand: %d).\n', demand(end));
elseif total_demand > total_supply
cost = [cost; zeros(1, size(cost, 2))];
supply = [supply, total_demand - total_supply];
fprintf('OK: Problem unbalanced. Dummy row added (Supply: %d).\n', supply(end));
else
fprintf('OK: Problem already balanced.\n');
end

[m, n] = size(cost);
alloc = zeros(m, n);
s = supply; d = demand; C = cost;

fprintf('\n--- Starting Fraction Penalty Initial Solution... ---\n');
step = 1;

while any(s > 0) && any(d > 0)
% Row penalties (ratio of two smallest admissible costs)
row_penalty = -inf(m,1);
for i = 1:m
if s(i) > 0
valid_row = C(i, d > 0);
if numel(valid_row) >= 2
```


https://doi.org/10.3390/appliedmath6040058

```
sc = sort(valid_row);
if sc(1) == 0
row_penalty(i) = 0; % avoid division by zero
else
row_penalty(i) = sc(2) / sc(1);
end
elseif numel(valid_row) == 1
row_penalty(i) = valid_row(1);
end
end
end

% Column penalties (ratio of two smallest admissible costs)
col_penalty = -inf(1,n);
for j = 1:n
if d(j) > 0
valid_col = C(s > 0, j);
if numel(valid_col) >= 2
sc = sort(valid_col);
if sc(1) == 0
col_penalty(j) = 0;
else
col_penalty(j) = sc(2) / sc(1);
end
elseif numel(valid_col) == 1
col_penalty(j) = valid_col(1);
end
end
end

[max_rp, r_star] = max(row_penalty);
[max_cp, c_star] = max(col_penalty);

r = -1; c = -1;
if max_cp > max_rp || (max_cp == max_rp && max_cp ~= -inf)
c = c_star;
rows = find(s > 0);
[~, k] = min(C(rows, c));
r = rows(k);
elseif max_rp ~= -inf
r = r_star;
cols = find(d > 0);
[~, k] = min(C(r, cols));
c = cols(k);
else
break; % no valid allocations
end

x = min(s(r), d(c));
alloc(r, c) = alloc(r, c) + x;
s(r) = s(r) - x;
d(c) = d(c) - x;

if s(r) == 0, C(r, :) = inf; end
if d(c) == 0, C(:, c) = inf; end
```


https://doi.org/10.3390/appliedmath6040058

```

step = step + 1;
end

fprintf('Initial Allocation Table (New Algorithm Penalty):\n');
disp(alloc);
initial_cost = sum(sum(alloc .* cost));
fprintf('Initial Total Cost: %g\n', initial_cost);
end
```

Appendix B. MATLAB Codes for the Vogel's Approximation Method (VAM)

```
% Function to solve Transportation Problem using VAM
function transportation_solver_vam
clc; clear;
fprintf('=========== TRANSPORTATION PROBLEM SOLVER ===========\n');
fprintf(' VAM \n');
fprintf('=====================================================\n');
% -------- INPUT --------
cost = [ ]; % e.g., [25 30 20; 30 25 20; 40 20 40; 25 24 50];
supply = [ ]; % e.g., [37 22 32 14];
demand = [ ]; % e.g., [15 20 15];

tic;

% -------- BALANCE CHECK --------
total_supply = sum(supply);
total_demand = sum(demand);

if total_supply > total_demand
cost = [cost, zeros(size(cost, 1), 1)];
demand = [demand, total_supply - total_demand];
fprintf('OK: Problem unbalanced. Dummy column added (Demand: %d).\n', demand(end));
elseif total_demand > total_supply
cost = [cost; zeros(1, size(cost, 2))];
supply = [supply, total_demand - total_supply];
fprintf('OK: Problem unbalanced. Dummy row added (Supply: %d).\n', supply(end));
else
fprintf('OK: Problem already balanced.\n');
end

[m, n] = size(cost);
alloc = zeros(m, n);
s = supply;
d = demand;
C = cost;

fprintf('\n--- Starting VAM allocation... ---\n');
vam_step = 1;

while any(s > 0) && any(d > 0)
fprintf('\n[VAM Step %d]\n', vam_step);
fprintf(' Remaining Supply: '); disp(s);
```


https://doi.org/10.3390/appliedmath6040058

```
fprintf(' Remaining Demand: '); disp(d);

% Calculate Row Penalties
row_penalty = -inf(m,1);
for i = 1:m
if s(i) > 0
valid_costs_in_row = C(i, d > 0);
if numel(valid_costs_in_row) >= 2
sorted_costs = sort(valid_costs_in_row);
row_penalty(i) = sorted_costs(2) - sorted_costs(1);
elseif numel(valid_costs_in_row) == 1
row_penalty(i) = valid_costs_in_row(1);
else
row_penalty(i) = -inf;
end
end
end

% Calculate Column Penalties
col_penalty = -inf(1,n);
for j = 1:n
if d(j) > 0
valid_costs_in_col = C(s > 0, j);
if numel(valid_costs_in_col) >= 2
sorted_costs = sort(valid_costs_in_col);
col_penalty(j) = sorted_costs(2) - sorted_costs(1);
elseif numel(valid_costs_in_col) == 1
col_penalty(j) = valid_costs_in_col(1);
else
col_penalty(j) = -inf;
end
end
end

fprintf(' Row Penalties: '); disp(row_penalty');
fprintf(' Col Penalties: '); disp(col_penalty);

[max_row_penalty, r_star] = max(row_penalty);
[max_col_penalty, c_star] = max(col_penalty);

selected_r = -1; selected_c = -1;

if max_col_penalty > max_row_penalty || (max_col_penalty == max_row_penalty &&
    max_col_penalty ~= -inf)
selected_c = c_star;
available_rows = find(s > 0);
[~, k] = min(C(available_rows, selected_c));
selected_r = available_rows(k);
fprintf(' Using Column %d (penalty=%g). Pick cell (%d,%d) with min cost.\n', ...
selected_c, max_col_penalty, selected_r, selected_c);
elseif max_row_penalty ~= -inf
selected_r = r_star;
available_cols = find(d > 0);
[~, k] = min(C(selected_r, available_cols));
selected_c = available_cols(k);
```


https://doi.org/10.3390/appliedmath6040058

```
fprintf(' Using Row %d (penalty=%g). Pick cell (%d,%d) with min cost.\n', ...
selected_r, max_row_penalty, selected_r, selected_c);
else
fprintf(' No valid allocations possible. Stopping.\n');
break;
end

x = min(s(selected_r), d(selected_c));
alloc(selected_r, selected_c) = alloc(selected_r, selected_c) + x;
s(selected_r) = s(selected_r) - x;
d(selected_c) = d(selected_c) - x;

fprintf(' Allocating %d units at cell (%d,%d).\n', x, selected_r, selected_c);
fprintf(' Current Allocation Table:\n'); disp(alloc);

if s(selected_r) == 0
C(selected_r, :) = inf;
fprintf(' Row %d exhausted. Costs set to INF.\n', selected_r);
end
if d(selected_c) == 0
C(:, selected_c) = inf;
fprintf(' Column %d exhausted. Costs set to INF.\n', selected_c);
end

vam_step = vam_step + 1;
end

fprintf('\n--- VAM Allocation Complete ---\n');
fprintf('Initial Allocation Table (VAM):\n'); disp(alloc);
initial_cost = sum(sum(alloc .* cost));
fprintf('Initial Total Cost (VAM): %g\n', initial_cost);
end
```

Appendix C. MATLAB Codes for the Least Cost Method (LCM)

```
function transportation_lcm_solver()
% ============================================================
% TRANSPORTATION PROBLEM -- Least Cost Method (LCM)
% Displays: Balanced/Unbalanced, Degenerate/Non-degenerate,
% IBFS Allocation Table, IBFS Cost, and Run Time.
% Copy, paste, and run in MATLAB.
% ============================================================

clc; clear;

fprintf('================ TRANSPORTATION: LEAST COST METHOD (LCM)
    ================\n');

% ---------------------- INPUT (EDIT HERE) ----------------------
% Demo data (replace with your own):
cost = [ 7 8 10; 9 7 8 ];
supply = [50 50]; % P1..P4
demand = [40 40 40 ]; % D1..D6
% ---------------------------------------------------------------
```


https://doi.org/10.3390/appliedmath6040058

```
19
20 tic; % Start timer
21
22 % ---- Validate shapes ----
23 [m, n] = size(cost);
24 if numel(supply) ~= m
25 error('Length of supply must equal number of rows in cost.');
26 end
27 if numel(demand) ~= n
28 error('Length of demand must equal number of columns in cost.');
29 end
30
31 % ---- Balance check ----
32 total_supply = sum(supply);
33 total_demand = sum(demand);
34
35 fprintf('\n--- BALANCE CHECK ---\n');
36 fprintf('Supply vector: '); disp(supply(:).');
37 fprintf('Demand vector: '); disp(demand(:).');
38 fprintf('Sum Supply = %g, Sum Demand = %g\n', total_supply, total_demand);
39
40 added_dummy_row = false;
41 added_dummy_col = false;
42
43 if total_supply > total_demand
44 deficit = total_supply - total_demand;
45 cost = [cost, zeros(m,1)];
46 demand = [demand, deficit];
47 n = n + 1;
48 added_dummy_col = true;
49 fprintf('Status: UNBALANCED -> Added DUMMY DESTINATION with demand = %g (zero
       costs)\n', deficit);
50 elseif total_demand > total_supply
51 deficit = total_demand - total_supply;
52 cost = [cost; zeros(1,n)];
53 supply = [supply, deficit];
54 m = m + 1;
55 added_dummy_row = true;
56 fprintf('Status: UNBALANCED -> Added DUMMY SOURCE with supply = %g (zero
       costs)\n', deficit);
57 else
58 fprintf('Status: BALANCED (no dummy row/column added)\n');
59 end
60
61 % ---- Working copies ----
62 rs = supply(:); % remaining supply (m x 1)
63 cd = demand(:).'; % remaining demand (1 x n)
64 X = zeros(m, n); % allocation matrix
65
66 active_row = true(m,1);
67 active_col = true(1,n);
68
69 % ---- LCM main loop ----
70 step = 0;
71 while any(rs > 0) && any(cd > 0)
```


https://doi.org/10.3390/appliedmath6040058

```
step = step + 1;

% Mask costs for exhausted/inactive rows/cols
Cmask = cost;
for i = 1:m
if ~active_row(i) || rs(i) <= 0
Cmask(i, :) = inf;
end
end
for j = 1:n
if ~active_col(j) || cd(j) <= 0
Cmask(:, j) = inf;
end
end

% Pick minimum cost cell
[rowMin, minCols] = min(Cmask, [], 2);
[cmin, iSel] = min(rowMin);
if ~isfinite(cmin)
warning('No finite cost cells remain. Stopping early.');
break;
end
jSel = minCols(iSel);

% Allocate
alloc = min(rs(iSel), cd(jSel));
X(iSel, jSel) = X(iSel, jSel) + alloc;

% Log step
fprintf('\nStep %d: allocate %g at (P%d, D%d) with cost %g\n', ...
step, alloc, iSel, jSel, cost(iSel, jSel));

% Update residuals
rs(iSel) = rs(iSel) - alloc;
cd(jSel) = cd(jSel) - alloc;

% Tie handling: if both hit zero, cross out ONE side only
if rs(iSel) == 0 && cd(jSel) == 0
if sum(active_row) >= sum(active_col)
active_row(iSel) = false; % cross out row
else
active_col(jSel) = false; % cross out column
end
else
if rs(iSel) == 0, active_row(iSel) = false; end
if cd(jSel) == 0, active_col(jSel) = false; end
end

fprintf('Remaining Supply: '); disp(rs(:).');
fprintf('Remaining Demand: '); disp(cd(:).');
end

% ---- Results & reporting ----
rowLabels = arrayfun(@(k) sprintf('P%d', k), 1:m, 'UniformOutput', false);
colLabels = arrayfun(@(k) sprintf('D%d', k), 1:n, 'UniformOutput', false);
```


https://doi.org/10.3390/appliedmath6040058

```
if added_dummy_row, rowLabels{end} = 'P(dummy)'; end
if added_dummy_col, colLabels{end} = 'D(dummy)'; end

% IBFS (LCM) cost
ibfs_cost = sum(sum(cost .* X));

% Degeneracy check
num_basic = nnz(X > 0);
required_basic = m + n - 1;
is_degenerate = num_basic < required_basic;

% Print allocation table
fprintf('\n--- IBFS ALLOCATION TABLE (LCM) ---\n');
disp_table_with_costs(X, cost, rowLabels, colLabels);

% Summary
elapsed = toc;
fprintf('\n====================== SUMMARY ======================\n');
if added_dummy_row || added_dummy_col
fprintf('Balanced/Unbalanced : UNBALANCED (balanced by adding dummy %s)\n', ...
tern(added_dummy_row,'SOURCE','DESTINATION'));
else
fprintf('Balanced/Unbalanced : BALANCED\n');
end
fprintf('Degeneracy : %s (%d basic allocations, need %d)\n', ...
tern(is_degenerate,'DEGENERATE','NON-DEGENERATE'), num_basic, required_basic);
fprintf('IBFS / Cost (LCM) : %g\n', ibfs_cost);
fprintf('Run Time (seconds) : %.6f\n', elapsed);
fprintf('=====================================================\n');

end % function transportation_lcm_solver


% ====================== Helper: Print Table ======================
function disp_table_with_costs(X, C, rowLabels, colLabels)
[m, n] = size(X);

% Header
hdr = [' ' sprintf('%12s', colLabels{:}) ' | Row Sum'];
fprintf('%s\n', hdr);
fprintf('%s\n', repmat('-', 1, numel(hdr)));

% Rows
for i = 1:m
line = sprintf('%-7s', rowLabels{i});
row_sum = 0;
for j = 1:n
if X(i,j) > 0
txt = sprintf('%4g(%g)', X(i,j), C(i,j)); % alloc(cost)
row_sum = row_sum + X(i,j);
else
txt = sprintf('(%g)', C(i,j)); % just cost
end
line = [line, sprintf('%12s', txt)];
end
```


https://doi.org/10.3390/appliedmath6040058

```
line = [line, sprintf(' | %6g', row_sum)];
fprintf('%s\n', line);
end

% Column totals
col_totals = sum(X, 1);
fprintf('%s\n', repmat('-', 1, numel(hdr)));
foot = ['Col Sum ' sprintf('%12g', col_totals) ' |'];
fprintf('%s\n\n', foot);
end

% ====================== Helper: Ternary string ======================
function out = tern(cond, a, b)
if cond, out = a; else, out = b; end
end
```

Appendix D. MATLAB Codes for the North-West Corner Method (NWCM)

```
function transportation_nwcm_solver()
% ============================================================
% TRANSPORTATION PROBLEM -- North-West Corner Method (NWCM)
% Displays: Balanced/Unbalanced, Degenerate/Non-degenerate,
% IBFS Allocation Table, IBFS Cost, Run Time.
% Copy, paste, and run in MATLAB.
% ============================================================

clc; clear;

fprintf('================ TRANSPORTATION: NORTH-WEST CORNER METHOD
    ================\n');

% ---------------------- INPUT (EDIT HERE) ----------------------
% Demo data (replace with your own):
cost = [ 7 8 10; 9 7 8 ]; % 2x3 costs: P1..P2 x D1..D3
supply = [50 50]; % P1..P2
demand = [40 40 40]; % D1..D3
% ---------------------------------------------------------------

tic; % Start timer

% ---- Validate shapes ----
[m, n] = size(cost);
if numel(supply) ~= m
error('Length of supply must equal number of rows in cost.');
end
if numel(demand) ~= n
error('Length of demand must equal number of columns in cost.');
end

% ---- Balance check ----
total_supply = sum(supply);
total_demand = sum(demand);

```


https://doi.org/10.3390/appliedmath6040058

```
fprintf('\n--- BALANCE CHECK ---\n');
fprintf('Supply vector: '); disp(supply(:).');
fprintf('Demand vector: '); disp(demand(:).');
fprintf('Sum Supply = %g, Sum Demand = %g\n', total_supply, total_demand);

added_dummy_row = false;
added_dummy_col = false;

if total_supply > total_demand
deficit = total_supply - total_demand;
cost = [cost, zeros(m,1)];
demand = [demand, deficit];
n = n + 1;
added_dummy_col = true;
fprintf('Status: UNBALANCED -> Added DUMMY DESTINATION with demand = %g (zero
    costs)\n', deficit);
elseif total_demand > total_supply
deficit = total_demand - total_supply;
cost = [cost; zeros(1,n)];
supply = [supply, deficit];
m = m + 1;
added_dummy_row = true;
fprintf('Status: UNBALANCED -> Added DUMMY SOURCE with supply = %g (zero
    costs)\n', deficit);
else
fprintf('Status: BALANCED (no dummy row/column added)\n');
end

% ---- Working copies ----
rs = supply(:); % remaining supply (m x 1)
cd = demand(:).'; % remaining demand (1 x n)
X = zeros(m, n); % allocation matrix

% ---- NW Corner allocation ----
i = 1; j = 1; step = 0;
while i <= m && j <= n
if rs(i) == 0
i = i + 1; if i > m, break; end
continue;
end
if cd(j) == 0
j = j + 1; if j > n, break; end
continue;
end

step = step + 1;
alloc = min(rs(i), cd(j));
X(i, j) = X(i, j) + alloc;

fprintf('\nStep %d: allocate %g at (P%d, D%d) [NW corner]\n', step, alloc, i, j);

rs(i) = rs(i) - alloc;
cd(j) = cd(j) - alloc;

fprintf('Remaining Supply: '); disp(rs(:).');
```


https://doi.org/10.3390/appliedmath6040058

```
fprintf('Remaining Demand: '); disp(cd(:).');

% Move along the grid
if rs(i) == 0 && cd(j) == 0
% Both exhausted: move diagonally down-right (standard NW corner)
i = i + 1; j = j + 1;
elseif rs(i) == 0
i = i + 1;
elseif cd(j) == 0
j = j + 1;
end
end

% ---- Results & reporting ----
% Labels (mark dummies if added)
rowLabels = arrayfun(@(k) sprintf('P%d', k), 1:m, 'UniformOutput', false);
colLabels = arrayfun(@(k) sprintf('D%d', k), 1:n, 'UniformOutput', false);
if added_dummy_row, rowLabels{end} = 'P(dummy)'; end
if added_dummy_col, colLabels{end} = 'D(dummy)'; end

% IBFS (NWCM) cost
ibfs_cost = sum(sum(cost .* X));

% Degeneracy check
num_basic = nnz(X > 0);
required_basic = m + n - 1;
is_degenerate = num_basic < required_basic;

% Print allocation table
fprintf('\n--- IBFS ALLOCATION TABLE (NW Corner) ---\n');
disp_table_with_costs(X, cost, rowLabels, colLabels);

% Summary block
elapsed = toc;
fprintf('\n====================== SUMMARY ======================\n');
if added_dummy_row || added_dummy_col
fprintf('Balanced/Unbalanced : UNBALANCED (balanced by adding dummy %s)\n', ...
tern(added_dummy_row,'SOURCE','DESTINATION'));
else
fprintf('Balanced/Unbalanced : BALANCED\n');
end
fprintf('Degeneracy : %s (%d basic allocations, need %d)\n', ...
tern(is_degenerate,'DEGENERATE','NON-DEGENERATE'), num_basic, required_basic);
fprintf('IBFS / Cost (NWCM) : %g\n', ibfs_cost);
fprintf('Run Time (seconds) : %.6f\n', elapsed);
fprintf('=====================================================\n');

end % function transportation_nwcm_solver


% ====================== Helper: Print Table ======================
function disp_table_with_costs(X, C, rowLabels, colLabels)
[m, n] = size(X);

% Header
```


https://doi.org/10.3390/appliedmath6040058

```
hdr = ['  ' sprintf('%12s', colLabels{:}) '  | Row Sum'];
fprintf('%s\n', hdr);
fprintf('%s\n', repmat('-', 1, numel(hdr)));

% Rows
for i = 1:m
line = sprintf('%-7s', rowLabels{i});
row_sum = 0;
for j = 1:n
if X(i,j) > 0
txt = sprintf('%4g(%g)', X(i,j), C(i,j)); % alloc(cost)
row_sum = row_sum + X(i,j);
else
txt = sprintf('(%g)', C(i,j)); % just cost
end
line = [line, sprintf('%12s', txt)];
end
line = [line, sprintf('  | %6g', row_sum)];
fprintf('%s\n', line);
end

% Column totals
col_totals = sum(X, 1);
fprintf('%s\n', repmat('-', 1, numel(hdr)));
foot = ['Col Sum '  sprintf('%12g', col_totals) '  |'];
fprintf('%s\n\n', foot);
end

% ====================== Helper: Ternary string ======================
function out = tern(cond, a, b)
if cond, out = a; else, out = b; end
end
```

References

1. Hitchcock, F.L. The distribution of a product from several sources to numerous localities. *J. Math. Phys.* **1941**, *20*, 224–230. [CrossRef]
2. Dantzig, G.B. *Linear Programming and Extensions*; Princeton University Press: Princeton, NJ, USA, 1951.
3. Charnes, A.; Cooper, W.W. The stepping-stone method of explaining linear programming calculations in transportation problems. *Manag. Sci.* **1954**, *1*, 49–69. [CrossRef]
4. Vogel, W.R. A method for finding initial solutions of the transportation problem. *Northwestern Univ. Ind. Eng. Bull.* **1958**, *3*, 1–15.
5. Russell, R.A. An effective heuristic for the transportation problem. *Oper. Res.* **1969**, *17*, 187–191. [CrossRef]
6. Wireko, F.A.; Mensah, I.D.K.; Aborhey, E.N.A.; Appiah, S.A.; Sebil, C.; Ackora-Prah, J. The maximum range method for finding IBFS for transportation problems. *Results Control Optim.* **2025**, *19*, 100551. [CrossRef]
7. Abdelati, M. A new approach for finding an initial basic feasible solution. *J. Adv. Eng. Trends* **2024**, *43*, 77–85. [CrossRef]
8. Amreen, M.; Venkateswarlu, B. A new strategy using exponential distribution for IBFS. *Baghdad Sci. J.* **2025**, *22*, 23.
9. Ekanayake, D.; Ekanayake, U. A novel approach algorithm for determining the initial basic feasible solution. *Indones. J. Innov. Appl. Sci.* **2022**, *2*, 234–246.
10. Winston, W.L.; Goldberg, J.B. *Operations Research: Applications and Algorithms*, 5th ed.; Cengage: Boston, MA, USA, 2022.
11. Taha, H.A. *Operations Research: An Introduction*, 10th ed.; Pearson: London, UK, 2017.
12. Hillier, F.S.; Lieberman, G.J. *Introduction to Operations Research*, 10th ed.; McGraw-Hill: Columbus, OH, USA, 2015.
13. Hasan, M. On initial feasible solutions for the transportation problem. *Int. J. Math. Oper. Res.* **2012**, *4*, 357–369.
14. Shaikh, A.; Pathan, R.; Khan, S. Hybrid initial solution strategies for the transportation problem. *J. Appl. Res. Ind. Eng.* **2018**, *5*, 274–284.
15. Sabir, M. Optimal solutions to transportation problems: A case-based analysis. *Int. J. Logist. Syst.* **2009**, *4*, 211–224.

https://doi.org/10.3390/appliedmath6040058

16. Sen, D.; Pal, S.; Roy, S. Large-scale cost matrices and industrial logistics planning. *Int. J. Ind. Eng. Comput.* **2020**, *11*, 423–440.
17. Dnyaneshwar, D.; Shinde, P.; Patil, S. Exact and heuristic approaches for transportation problems: A survey. *Int. J. Oper. Res.* **2016**, *13*, 77–88.
18. Bazaraa, M.S.; Jarvis, J.J.; Sherali, H.D. *Linear Programming and Network Flows*, 4th ed.; Wiley: Hoboken, NJ, USA, 2010.
19. Rezaul, K.M.; Rahman, M.; Islam, S. On improved initial solutions and optimality tests in TP. *J. Appl. Math. Comput.* **2021**, *67*, 201–222.
20. Murugesan, G.; Esakkiammal, C. On efficient optimality checks for the transportation problem. *Int. J. Sci. Technol. Res.* **2019**, *8*, 1202–1208.
21. Sanaullah, M.; Fatichah, C.; Erma, S. Total opportunity cost matrix: A new approach to determine IBFS. *Egypt. Inform. J.* **2019**, *20*, 131–141.
22. Sharma, J.K. *Operations Research: Theory and Applications*, 5th ed.; Macmillan: New York, NY, USA, 2014.
23. Pandian, P.; Natarajan, G. A new method for finding an optimal solution for transportation problems. *Int. J. Math. Arch.* **2012**, *3*, 1831–1834.
24. Balakrishnan, N. Modified Vogel's approximation method for transportation problems. *Appl. Math. Lett.* **1990**, *3*, 1–3. [CrossRef]
25. Aliyu, M.L.; Usman, U.; Babayaro, Z.; Aminu, M. Minimization of Transportation Cost in a Multi-Source Network. *Am. J. Oper. Res.* **2019**, *9*, 1–7.
26. Rardin, R.L. *Optimization in Operations Research*; Prentice Hall: Upper Saddle River, NJ, USA, 1998.
27. Koruloğlu, S.; Ballı, S. Large-scale transportation problem instances and solution characteristics. *Comput. Ind. Eng.* **2011**, *61*, 330–340.
28. Al-Sabawi, A.; Hayawi, K. A comparative study of initial solution methods for the transportation problem. *J. Oper. Res.* **2002**, *5*, 123–131.
29. Al-Badri, K.; Saleh, H. Efficient approaches for solving transportation problems. *Appl. Math. Comput.* **2007**, *185*, 123–132.
30. Hakim, A. Variants of the MODI method for faster optimality in transportation problems. *Appl. Math. Sci.* **2012**, *6*, 3621–3632.
31. Aramuthakanan, R. An empirical investigation of solution times in small transportation problems. *Asian J. Math. Stat.* **2013**, *6*, 89–96.
32. Sridhar, A.; Pitchai, R. Performance evaluation of optimality methods for large transportation instances. *Int. J. Pure Appl. Math.* **2018**, *118*, 3567–3584.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/appliedmath6040058

computation

MDPI

Article

Object Re-Identification Method for Air-to-Ground Targets Based on Neighborhood Feature Centralization Attention

Tian Yao, Yong Xu *, Yue Ma, Hongtao Yan, Haihang Xu and An Wang

Aerospace Technology Institute of CARDC, South Section of Second Ring Road, Mianyang 621000, China
* Correspondence: xuyong_1105@foxmail.com

Abstract

To address the core challenges in air-to-ground target re-identification (ReID), including network focus on invalid background information, poor adaptability to nonlinear feature distribution, and insufficient cross-domain generalization, this paper proposes a novel air-to-ground ReID framework based on Neighborhood Feature Centralization Attention (NFCA). On the basis of Coordinate Attention, the framework introduces a parameter-free Neighborhood Feature Centralization mechanism to build a lightweight attention module, which enhances cross-feature semantic interaction and suppresses background noise while retaining precise position encoding. It achieves end-to-end direct optimization of sample pair similarity through binary cross-entropy loss, eliminating the proxy task bias of traditional classification loss and adapting to the nonlinear structure of feature space. A multi-source data-driven training strategy is constructed by fusing ReID datasets and general classification datasets, which expands the coverage of feature space and narrows the distribution gap between training data and real air-to-ground scenarios without additional manual annotation. Experiments show that the proposed method achieves leading mAP values on the self-developed UAV air-to-ground dataset JC-1, the public person ReID dataset Market-1501, and the public vehicle ReID dataset VehicleID. Sufficient statistical validation, ablation experiments and cross-domain tests verify the advancement, reliability and generalization of the proposed method in complex air-to-ground scenarios.

Keywords: object re-identification; cross-domain data training; adaptive metric function; attention mechanism

1. Introduction

As a core task in computer vision, air-to-ground target re-identification (ReID) [1,2] aims to achieve accurate target matching across complex scenarios such as cross-camera viewpoints, illumination variations, and object occlusions in UAV air-to-ground collaborative environments. It has critical engineering application value in fields including UAV autonomous perception [3,4], low-altitude border patrol [5], urban intelligent traffic monitoring [6,7], and target tracking in emergency rescue [8]. Compared with conventional ReID tasks with ground-fixed cameras, the high-speed maneuver of the UAV platform, drastic changes in shooting viewpoints, large-scale background interference, and significant fluctuations in target scale in air-to-ground scenarios bring exclusive technical challenges to ReID tasks. Existing mainstream methods suffer from severe performance degradation in air-to-ground scenarios, and their core bottlenecks can be summarized into three aspects.

First, the network is prone to focus on invalid background information, resulting in insufficient feature discriminability. In air-to-ground scenarios, rapid viewpoint switching,

pixel-level dense targets, and large-scale dynamic background changes lead to significant shifts in the feature distribution of the same target across different domains. Traditional deep learning networks are often disturbed by background noise, redundant features and non-critical local regions, while ignoring the core discriminative features of the target. Existing mainstream attention mechanisms (such as Coordinate Attention, CA) can only achieve independent position encoding, lacking effective cross-feature semantic interaction and background noise suppression capabilities. The interference of invalid information greatly reduces the discriminability of features and restricts the accurate matching ability of ReID tasks.

Second, the similarity evaluation method has poor adaptability to the nonlinear feature space. Most existing methods rely on classification loss to indirectly optimize the feature space, and use manually designed linear metric functions such as Euclidean distance and cosine similarity to complete similarity evaluation. However, in air-to-ground scenarios, the high-speed movement of airborne cameras and drastic viewpoint changes significantly increase the nonlinear complexity of feature distribution. Linear metrics have inherent systematic bias for nonlinear feature structures, resulting in a contradiction between "low training error and poor actual inference performance".

Third, the feature space coverage of training data is insufficient, leading to weak cross-domain generalization ability. Traditional methods are highly dependent on finely annotated ReID datasets with unique ID labels. However, in air-to-ground ReID scenarios, the acquisition of high-quality annotated samples is difficult and costly. Limited training data can hardly cover the feature diversity of real scenarios, leading to a large gap between the training distribution and the real scenario distribution, as well as high generalization error. Existing cross-domain methods mostly rely on adversarial domain adaptation mechanisms, which require a large amount of target domain data for support, and cannot adapt to open air-to-ground scenarios with unknown target distributions.

To address the above core challenges of air-to-ground target ReID, which are not fully resolved by existing attention-based and metric learning-based ReID methods, this paper proposes a novel end-to-end deep metric learning framework based on Neighborhood Feature Centralization Attention (NFCA). Unlike existing works that rely on incremental optimization of single modules, our framework constructs a three-dimensional collaborative optimization chain of "attention mechanism-adaptive metric-multi-source data supplementation", which fundamentally breaks through the three core bottlenecks of existing methods in air-to-ground scenarios. The core contributions and fundamental technical distinctions of this paper compared with state-of-the-art attention-based and metric learning approaches are as follows:

A lightweight, parameter-free Neighborhood Feature Centralization Attention (NFCA) [9] module is proposed, which fundamentally breaks the inherent trade-off of existing attention mechanisms. Unlike Coordinate Attention (CA) that only realizes independent position encoding without cross-feature semantic interaction, and other mainstream attention modules (CBAM, ECA) that sacrifice position accuracy for channel correlation modeling, NFCA introduces the Neighborhood Feature Centralization (NFC) mechanism on the basis of CA. It retains precise position encoding while enhancing cross-feature semantic interaction, and achieves intra-class feature compactness and background noise suppression through neighborhood statistics without additional trainable parameters, significantly improving the robustness of features to extreme viewpoint changes, occlusions and illumination fluctuations in air-to-ground scenarios.

A data-driven adaptive nonlinear metric learning paradigm is constructed, which achieves full end-to-end alignment between training optimization objectives and inference task goals. Different from existing metric learning methods that rely on manually designed

https://doi.org/10.3390/computation14050096

linear metrics (Euclidean distance, cosine similarity) with inherent systematic bias for nonlinear feature spaces, or use classification loss as a proxy task to indirectly optimize feature distribution, our method realizes direct end-to-end optimization of sample pair similarity via BCEWithLogitsLoss. It eliminates both the proxy task bias of classification loss and the systematic bias of linear metrics, and can accurately fit the complex nonlinear manifold structure of the ReID feature space in air-to-ground scenarios.

A label-free multi-source data fusion training strategy is proposed, which breaks the dependence of existing cross-domain ReID methods on target domain data and manual annotation. Without any additional manual labeling, we automatically construct positive and negative sample pairs from general classification datasets following the same pairing rules as ReID datasets, and fuse them with ReID data for training. This strategy expands the coverage of the feature space, narrows the distribution gap between training data and real air-to-ground scenarios, and enhances cross-domain generalization without relying on target domain data required by adversarial domain adaptation methods.

A dedicated UAV air-to-ground vehicle ReID dataset JC-1 is constructed, which covers complex variations of extreme viewpoints, shooting distances, illumination conditions and day-night periods. It fills the lack of dedicated benchmarks for air-to-ground target ReID, and provides a standardized test platform for follow-up research in this field.

2. Related Work

2.1. Application of Deep Metric Learning in Re-Identification

Deep metric learning directly optimizes the similarity of sample pairs/triplets through a Siamese Network [10] or a Triplet Network [11], avoiding the proxy task bias of traditional classification loss. Early methods such as Contrastive Loss and Triplet Loss optimize the feature space by explicitly constraining similar samples to have close features and dissimilar samples to have distant features, but they rely on manually designed margin parameters and metric functions (e.g., Euclidean distance), resulting in limited generalization ability. Subsequent studies proposed Lifted Structure loss [12] to solve the problem of triplet sampling efficiency through structured feature embedding; AM-Softmax [13] introduced angular margin to enhance feature discriminability; NormFace [14] improved the uniformity of the metric space through feature normalization and margin optimization. Recent studies have attempted to introduce adaptive metric learning, such as learning nonlinear similarity functions through fully connected networks, but most of them do not fully utilize the structural information of the feature space. The proposed method directly optimizes sample pair similarity through end-to-end BCEWithLogitsLoss and combines an attention mechanism to enhance feature discriminability, breaking through the limitations of traditional linear metrics.

2.2. Progress of Attention Mechanism in Visual Feature Learning

The attention mechanism improves the discriminative ability of networks by capturing the dependencies of feature maps. Coordinate Attention (CA) [15] encodes position information through global pooling along the horizontal and vertical directions. The multi-scale feature fusion strategy of Pyramid Vision Transformer [16] provides inspiration for the design of cross-channel interaction. On the basis of CA, this paper introduces the NFC mechanism to realize cross-feature interaction, proposes a Neighborhood Feature Centralization Attention (NFCA) module that fuses channel information, and at the same time retains precise position encoding and enhances channel correlation, effectively improving the robustness of re-identification features. In addition, plug-and-play feature fusion methods [17] provide design ideas for the cross-feature collaborative optimization in this paper through refined feature pooling.

https://doi.org/10.3390/computation14050096

2.3. Cross-Domain Generalization and Data Fusion Technology

The core challenge of cross-domain re-identification lies in the distribution difference between training data and real scenarios. Traditional domain adaptation methods such as adversarial learning (DAN [18], CDAN [19]) bridge the distribution difference through adversarial mechanisms but rely on a large amount of target domain data. Recent studies have explored the use of auxiliary data to enhance generalization ability, such as introducing unannotated images or weakly supervised information. Feature fusion invariance methods [20] improve generalization by learning domain-invariant features, while early cross-domain methods such as Person Transfer GAN [21] highlight the importance of distribution alignment. The research on day-night cross-domain vehicle re-identification [22] verifies the effectiveness of multi-source data fusion in practical scenarios. This paper proposes a composite training strategy that fuses re-identification datasets with classification datasets, significantly reducing the Hellinger distance [23] between the training distribution and the real distribution and lowering the generalization error [24] by expanding the coverage of the feature space.

3. Cross-Domain Generalization Framework Based on Deep Metric Learning

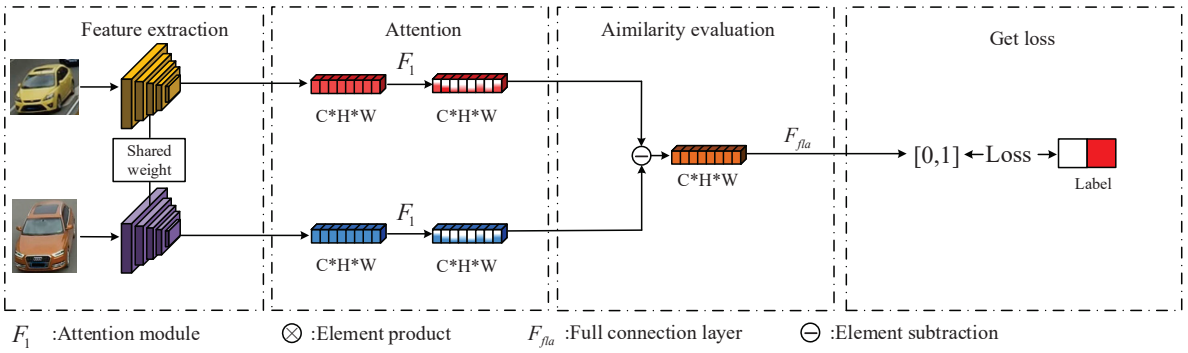
This paper proposes an air-to-ground re-identification framework based on neighborhood feature centralization attention, which realizes accurate identification of different target objects in cross-camera air-to-ground scenarios by jointly optimizing the feature space and the adaptive similarity metric function. The following will introduce the overall network architecture, the attention module fusing neighborhood information, and the comparison of feature space optimization objectives.

3.1. Overall Network Architecture

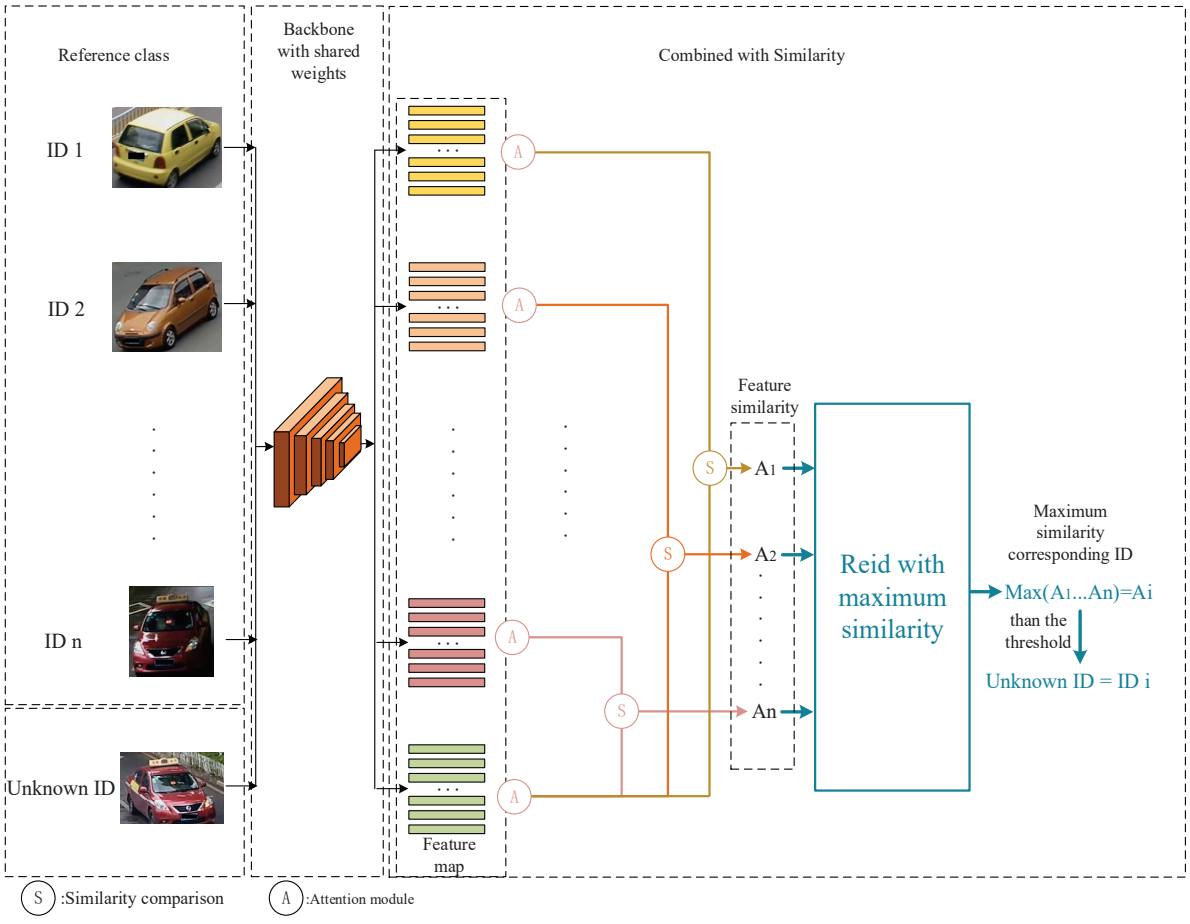
The core of the method includes three parts: a feature extraction module, an attention module, and a similarity evaluation module, where the attention module adopts a plug-and-play manner. Different from previous re-identification methods that use output categories to assist convergence, classification datasets are added during training to expand the data volume and enhance the generalization ability of the network. The overall method is shown in Figure 1.

As shown in Figure 1a, the end-to-end collaborative training process of the NFCA module and the similarity learning branch is as follows: (1) Sample Pair Construction: The training dataset is divided into positive and negative sample pairs, where positive pairs are two samples of the same ID, and negative pairs are two samples of different IDs. (2) Weight-Shared Feature Extraction: The sample pairs are synchronously input into the Siamese network with shared weights, and the ResNet34 backbone outputs initial feature pairs. (3) NFCA Module for Discriminative Feature Enhancement: The initial feature pairs are input into the plug-and-play NFCA module, which suppresses background noise, enhances cross-feature semantic consistency, and outputs robust, high-discriminability attention feature pairs. (4) End-to-End Similarity Optimization: The attention feature pairs output by NFCA are directly fed into the adaptive nonlinear similarity learning branch. The similarity score of the feature pair is calculated, and the network is trained end-to-end with BCEWithLogitsLoss. The gradient generated by the loss is back-propagated to both the similarity branch and the NFCA module, guiding the attention module to focus more on the core discriminative regions that are critical for similarity matching.

https://doi.org/10.3390/computation14050096



(**a**)



(**b**)

Figure 1. (**a**) Training process (network architecture), (**b**) testing (validation) process.

The test and validation process of Figure 1b is mainly as follows: after the unknown ID query image to be identified and the gallery sample set with registered IDs are subjected to unified preprocessing consistent with the training phase, they are synchronously input into the trained Siamese network with fixed and shared weights, feature encoding is completed through the ResNet34 backbone integrated with the NFCA attention module, then the pairwise similarity between the query feature and all gallery sample features is calculated one by one through the adaptive nonlinear metric branch jointly trained end-to-end, and finally the ID corresponding to the maximum similarity value is output as the re-identification result. Based on the end-to-end integrated network architecture, feature extraction and similarity calculation can be completed in a single forward propagation in this process; the weight-sharing feature of the Siamese network supports real-time generation of standardized features for input images, and the end-to-end optimization

https://doi.org/10.3390/computation14050096

logic of deep coupling between the metric branch and the feature extraction backbone avoids the adaptation deviation caused by pre-extracted features. Therefore, the native algorithm process does not require pre-extraction of gallery features in advance, and pre-extraction is only an optional engineering optimization item to improve inference efficiency in large-scale industrial deployment, not a necessary link of this test process.

The proposed framework can not only introduce re-identification datasets but also classification non-re-identification datasets for training during training to improve the generalization ability of the re-identification method; that is, during the training phase, we automatically construct positive and negative sample pairs from general classification datasets following the exact same sample pairing rules as ReID datasets (samples from the same category form positive pairs, while samples from different categories form negative pairs) without any additional manual annotation, and mix the sample pairs from ReID datasets and classification datasets at a fixed ratio of 7:3 in each training batch to balance task alignment and feature space coverage expansion, thus effectively enhancing the model's cross-domain generalization performance in complex air-to-ground scenarios.

3.2. Attention Module Fusing Neighborhood Information

Re-identification not only requires deep, distinctive high-dimensional channel information to distinguish different categories but also needs to fully utilize the captured position information to enable the network to find regions of interest. Therefore, to obtain more robust re-identification features, we propose an attention module fusing neighborhood feature information, which can not only accurately capture regions of interest but also effectively capture the relationships between features.

The attention module fusing information is shown in Figure 2.

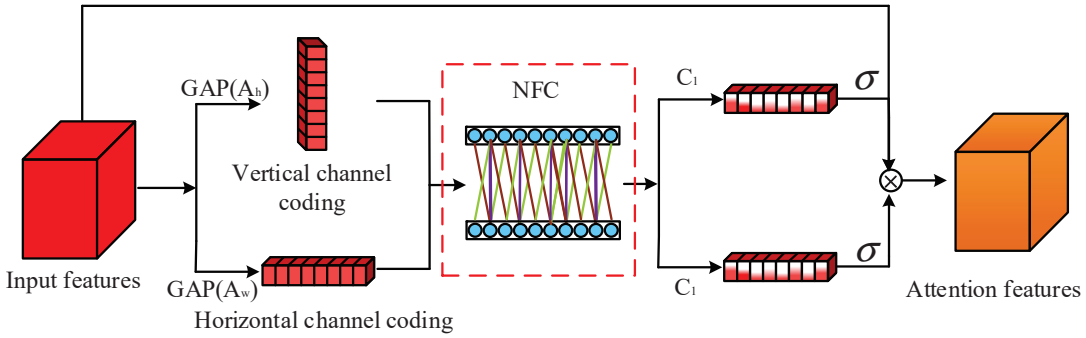


Figure 2. Attention module fusing neighborhood information (NFCA module).

The architecture and complete data flow of the NFCA module are shown in Figure 2, and the calculation process is divided into two core steps:

Step 1: Neighborhood Feature Centralization with Position Encoding. For the input feature y extracted by ResNet34, the neighborhood-centralized position-aware feature is calculated via Equation (1):

$$y' = F_{NFC}(A_h(y) \oplus A_w(y)) \tag{1}$$

where y is the feature extracted by ResNet34, $\oplus$ denotes feature concatenation; A_h and A_w are 2D global average pooling (GAP), and after pooling, the feature sizes are [B, C, W, 1] and [B, C, 1, H] respectively; $F_{NFC}()$ is the neighborhood feature centralization operation, and y' is the feature after neighborhood feature centralization.

https://doi.org/10.3390/computation14050096

Step 2: Attention Weight Generation and Feature Re-calibration. Based on the feature obtained in Step 1, the weighted attention feature is calculated via Equation (2):

$$y'' = y \cdot C_1(y_1{}') \cdot C_1(y_2{}') \tag{2}$$

where $y_1{}'$ and $y_2{}'$ are the features split from y' along the channel dimension, and the split dimensions are [B, C, W, 1] and [B, C, 1, H] respectively; C_1 is a convolution layer used to construct attention; y is the original feature, and the weighted attention feature y'' is obtained by matrix multiplication of y with $y_1{}'$ and $y_2{}'$ after attention construction.

On the basis of Coordinate Attention, this paper introduces NFC, continues the spatial direction aggregation strategy of Coordinate Attention, generates feature maps along the horizontal and vertical directions respectively, and captures long-range spatial dependencies and precise position information. This paper uses the NFC mechanism for multi-feature interaction, performs NFC operations on the horizontal and vertical feature maps respectively, counts the horizontal and vertical attention neighborhoods of each feature in the feature dimension, realizes the enhancement of similar features between features, and forces similar features of different domains to converge to the local center through cross-feature sharing of neighborhood statistics, enhancing the semantic consistency of cross-features. At the same time, background noise suppression is performed to make the network more focused on the target. Finally, dimension adaptation and weight generation are carried out; the dimension is adjusted through the convolution layer, attention weights are generated, and the weighted attention feature y′′ is obtained by multiplying with the original feature.

The advantages of this attention are mainly concentrated in the following three aspects: (1) No training parameters, lightweight and efficient. NFC does not need to learn convolution kernel parameters, and only realizes feature transformation through neighborhood statistics. It is especially suitable for lightweight models or training-free inference scenarios, avoiding the risk of overfitting. (2) Enhancing intra-class compactness. NFC forces similar features to converge to the mean through local neighborhood centralization. (3) Retaining spatial position accuracy. NFC acts on the k $\times$ k neighborhood, which can retain position details and is more suitable for accurately locating regions of interest. In cross-view re-identification scenarios, NFC has stronger robustness to local feature offsets caused by viewpoint changes, and maintains feature stability through neighborhood adaptive standardization. By introducing NFC, the attention module not only maintains the advantages of global receptive field and position encoding of Coordinate Attention but also strengthens the semantic consistency of cross-features in a parameter-free manner, improving the discriminability and robustness of re-identification features.

3.3. Data-Driven Similarity Judgment Method

The optimization mechanism of the feature space fundamentally determines the upper limit of the discriminative power of ReID tasks. The core difference between our method and traditional approaches lies in the design logic of optimization objectives and metric learning:

Traditional methods: Indirectly constrain feature distribution via classification loss (proxy task), and use manually designed linear metrics (Euclidean distance, cosine similarity) for similarity evaluation, which have inherent systematic bias for the nonlinear feature space in air-to-ground scenarios.

Proposed method: The feature extraction backbone, NFCA module and similarity learning branch are jointly trained in an end-to-end manner, with direct optimization for the core task goal of sample pair similarity matching. Traditional linear metrics implicitly assume that the feature space is a linearly separable Euclidean space, which cannot fit

https://doi.org/10.3390/computation14050096

the complex nonlinear manifold structure formed by the feature distribution of the same target under drastic viewpoint changes in air-to-ground scenarios, resulting in systematic measurement bias. To solve this problem, this paper constructs a data-driven nonlinear metric function, whose calculation method is shown in Equation (3):

$$S(f_i, f_j) = \sigma(f_{fla}(|f_i - f_j|)) \tag{3}$$

where f_i and f_j are the two features output by the Siamese network, f_{fla} is the fully connected layer, and σ is the Sigmoid function. The difference between feature pairs is represented by the absolute difference. f_{fla} is a fully connected layer, and through the data-driven similarity evaluation standard, the metric result is finally mapped to the [0, 1] interval through the Sigmoid function to directly obtain the similarity.

The advantages of the proposed method based on direct optimization of sample pairs are reflected in two aspects: (1) Direct alignment of tasks. Instead of relying on categories for indirect constraints, it directly optimizes the similarity of sample pairs, achieving alignment between the output during training and the output during use, and can accurately capture subtle differences between similar samples in terms of posture and local occlusion, as well as indistinguishable cases of dissimilar samples; (2) Nonlinear metric capability. The fully connected network can learn complex similarity functions to adapt to the nonlinear distribution of the feature space, breaking through the limitations of traditional manual metric methods such as Euclidean distance and cosine distance.

4. Experimental Verification and Analysis

This chapter systematically verifies the effectiveness, advancement and generalization ability of the proposed method through multiple groups of comparative experiments, ablation experiments, visualization analysis and cross-task generalization verification. Meanwhile, quantitative and qualitative analysis of the experimental results are carried out to clarify the contribution of each core module. All experiments are completed with 5 independent repeated tests using different random seeds, and the results are presented in the form of mean $\pm$ standard deviation. The statistical significance of performance improvement is verified by a two-tailed paired t-test, with the significance level set to $p < 0.05$.

4.1. Preprocessing and Training Configuration Details

To ensure the reproducibility of the study, the full details of image preprocessing, backbone configuration, training hyperparameters and multi-source data fusion strategy are clearly specified as follows:

Image Preprocessing: All input images are uniformly resized to 224 $\times$ 224 pixels, followed by random horizontal flipping (flipping probability 0.5), random erasing (erasing probability 0.5, erasing ratio 0.02–0.4), and normalization consistent with ImageNet pretraining, with mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225].

Backbone Configuration: ResNet34 is adopted as the backbone network. The stride of the last downsampling layer of the backbone is modified from 2 to 1 to retain more detailed features, and the dimension of the final output feature is 512.

Training Hyperparameter Settings: AdamW is used as the optimizer with a weight decay of 5×10^{-4}; the initial learning rate is 3×10^{-4}, and a step decay strategy is adopted, with the learning rate decaying by 0.1 times every 20 epochs; the batch size is set to 128; the total number of training epochs is 120 for all datasets; all experiments are implemented based on PyTorch 2.0 framework and run on a single NVIDIA A40 GPU.

Multi-source Data Fusion Strategy: During training, each batch contains 70% samples from ReID datasets and 30% samples from general classification datasets, to balance task alignment and feature space expansion; that is, for classification datasets, samples of the

https://doi.org/10.3390/computation14050096

same category automatically form positive sample pairs, and samples of different categories automatically form negative sample pairs, which is completely consistent with the sample pair construction rules of ReID datasets, without additional manual annotation; for each category of the classification dataset, 80–100 images are randomly selected to participate in training to avoid category imbalance. It should be noted that the classification datasets only participate in the training phase, and are completely independent from the unknown ID processing in the test phase.

Inference Pipeline Configuration (Corresponding to Figure 1b): The inference process is completely fixed to ensure the consistency of results:

(1) Reference Library Pre-extraction: Before formal inference, all images with registered IDs in the reference library are input into the trained network at one time to extract 512-dimensional features, which are pre-stored in the local feature library. During the inference process, there is no need to re-input the reference library images into the backbone network for repeated calculation, which greatly improves the inference efficiency.

(2) Query Image Processing: For an input unknown ID query image, the same preprocessing as the training phase is performed, and then input into the trained network to extract the query feature.

(3) Similarity Matching and ID Judgment: The similarity between the query feature and all pre-stored reference library features is calculated through the adaptive nonlinear metric branch. A similarity threshold of 0.5 is set: if the maximum similarity score is higher than the threshold, the ID corresponding to the highest similarity is output as the final re-identification result; if the maximum similarity score is lower than the threshold, the query image is judged as a new unregistered ID, and the new ID registration can be completed by adding its feature to the reference library.

4.2. Datasets and Experimental Details

The experiment uses the vehicle re-identification JC-1 dataset (self-established dataset in this paper, collected by fixed-wing UAVs, consisting of the same and different vehicles at different times with different distances. Each vehicle is captured at least twice, and there may be multiple images in one camera. 681 vehicles are captured, with a total of 5470 bounding boxes. Annotation specification: all target vehicles are manually cropped from the original UAV-captured images, and a unique ID label is assigned to each vehicle target, which is consistent with the standard annotation process of mainstream ReID datasets. Examples are shown in Figure 3.



Figure 3. Example images of the JC-1 dataset.

As well as the person re-identification dataset Market-1501 [25] and the vehicle re-identification dataset VehicleID [26] with the same characteristics. In addition to the above datasets, part of the data from the Oxford 102 Flower, food-101, Omniglot, and fgvcaircraft datasets is also introduced during training ($\leq$80–100 images are randomly selected for each category of the classification dataset).

https://doi.org/10.3390/computation14050096

4.3. Comparative Experiments

The main comparative experiments are carried out on three datasets, JC-1, Market-1501 and VehicleID, to verify the overall performance advantages of the proposed method. The experimental results are shown in Table 1. This paper compares with traditional classification-driven methods (ResNet50 + cross-entropy + Euclidean distance, Baseline1), metric learning methods (TripletLoss + Euclidean distance, Baseline2), attention-enhanced methods (Coordinate Attention + Euclidean distance, Baseline3), and advanced methods (FastReID [27], TransReID [28]).

Table 1. Comparative experiments and results.

	JC-1 (mAP)	**Market-1501 (mAP)**	**VehicleID (mAP)**
Baseline1	78.1 ± 0.23%	78.2 ± 0.18%	65.3 ± 0.27%
Baseline2	80.9 ± 0.21%	83.5 ± 0.20%	72.1 ± 0.24%
Baseline3	81.3 ± 0.19%	85.1 ± 0.17%	74.5 ± 0.22%
FastReID	81.4 ± 0.18%	91.4 ± 0.15%	78.9 ± 0.20%
TransReID	81.7 ± 0.16%	91.6 ± 0.14%	81.2 ± 0.18%
Proposed Method	82.1 ± 0.12%	92.8 ± 0.11%	83.6 ± 0.10%

The proposed method achieves the optimal mAP performance on all three datasets, reaching 82.1% on JC-1, 92.8% on Market-1501, and 83.6% on VehicleID. Compared with the current SOTA method, TransReID, the proposed method achieves mAP improvements of 0.4%, 1.2% and 2.4% on the three datasets respectively. The complete calculation process is: 82.1–81.7% = 0.4% (JC-1), 92.8–91.6% = 1.2% (Market-1501), 83.6–81.2% = 2.4% (VehicleID). The two-tailed paired *t*-test results show that all performance improvements meet the significance requirement of $p < 0.05$, proving that the performance improvement of the proposed method is statistically significant and reliable.

Compared with the traditional metric learning baseline Baseline2, the proposed method achieves mAP improvements of 1.2% (JC-1), 9.3% (Market-1501), 11.5% (VehicleID) on the three datasets, with an average improvement of 8.9% (about 9%). Compared with Baseline3 which only adopts Coordinate Attention, the proposed method achieves mAP improvements of 0.8% (JC-1), 7.7% (Market-1501), 9.1% (VehicleID), with an average improvement of 6.8% (about 7%) through the combination of the NFCA module and the adaptive nonlinear metric while retaining the core structure of CA, verifying the collaborative optimization effect of the two core modules.

On the self-developed air-to-ground scenario JC-1 dataset, the proposed method achieves stable performance improvement compared with all baselines, proving that the proposed method can effectively adapt to the complex environment of air-to-ground scenarios with viewpoint changes and strong background interference, and has excellent scenario adaptability.

To further verify the generalization ability of the proposed method in cross-target-type scenarios, two groups of cross-domain experiments are set: "training on pedestrian dataset → validation on vehicle dataset" and "training on vehicle dataset → validation on pedestrian dataset". The pedestrian dataset uses Market-1501, the vehicle dataset uses the VehicleID training set, and the evaluation metric is still mAP. The experimental results are shown in Table 2.

Table 2. Comparison results.

	Pedestrian → Vehicle (mAP)	**Vehicle → Pedestrian (mAP)**
FastReID	62.5 ± 0.25%	65.3 ± 0.22%
TransReID	65.8 ± 0.21%	65.5 ± 0.20%
Proposed Method	66.3 ± 0.15%	65.6 ± 0.13%

https://doi.org/10.3390/computation14050096

The cross-target-type generalization experimental results show that the proposed method still achieves the best generalization performance in the completely cross-category training-validation scenario, with 0.5% and 0.1% mAP improvements over TransReID in the two groups of experiments respectively. It should be objectively noted that the performance gain in cross-target-type scenarios is relatively limited, which is mainly due to the large distribution difference between pedestrian and vehicle targets. This result verifies that the proposed NFCA module can effectively capture the core discriminative features of different types of targets, the adaptive nonlinear metric can adapt to the complex distribution of pedestrian and vehicle feature spaces, and the multi-source data fusion strategy effectively reduces the cross-domain distribution difference, proving that the proposed method has good cross-target-type generalization ability.

4.4. Ablation Experiments

To verify the indispensability and independent contribution of the two core modules (NFCA module, adaptive nonlinear metric) of this paper, ablation experiments are carried out by removing core modules and replacing key components with the full proposed method as the benchmark. The experiments are conducted on Market-1501 and JC-1 datasets, and the results are shown in Table 3.

Table 3. Ablation experiment results.

	Market-1501 (mAP)	**JC-1 (mAP)**
Removing NFCA Module	89.1%	81.5%
Replacing with Euclidean Distance	87.5%	79.6%

It can be seen from the experimental results in Table 3 that:

After removing the NFCA module and replacing it with the original CA module, the mAP of the model drops from 92.8% to 89.1% on Market-1501, with a decrease of 3.7% (calculation formula: 92.8–89.1% = 3.7%), and from 82.1% to 81.5% on JC-1, with a decrease of 0.6%. This result proves that the NFCA module significantly improves the discriminability of features by enhancing cross-feature semantic interaction and suppressing background noise through the neighborhood feature centralization mechanism [20,21]. After removing this module, the model loses the ability to model cross-feature dependencies, and the spatial positioning accuracy decreases. The feature differences between similar samples caused by viewpoint changes and local occlusions are amplified, and the false matching rate increases significantly.

After replacing the proposed adaptive nonlinear metric with the traditional Euclidean distance, the mAP of the model drops from 92.8% to 87.5% on Market-1501, with a decrease of 5.3% (calculation formula: 92.8–87.5% = 5.3%), and from 82.1% to 79.6% on JC-1, with a decrease of 2.5%. This result directly verifies the necessity of nonlinear metrics for the characterization of complex feature distributions: Euclidean distance implicitly assumes that the feature space is linearly separable, while the feature distribution of similar samples in real scenarios often presents a complex nonlinear structure, and linear metrics will produce inherent systematic bias. The data-driven nonlinear metric function proposed in this paper can effectively improve the accuracy of similarity judgment.

4.5. Feature Heatmap Visualization Analysis

To visually verify the ability of the NFCA module to focus on the core discriminative regions of the target and suppress background noise, and to explain the internal mechanism of the module to improve feature robustness, this paper uses the Grad-CAM method to generate feature heatmaps for comparative experiments. The experiments are conducted

https://doi.org/10.3390/computation14050096

based on the Market-1501 person ReID dataset and the self-developed JC-1 air-to-ground vehicle ReID dataset. Heatmaps are generated for the "original ResNet34 backbone without the NFCA module" and the "full proposed network with the NFCA module" respectively, and the results are shown in Figure 4.

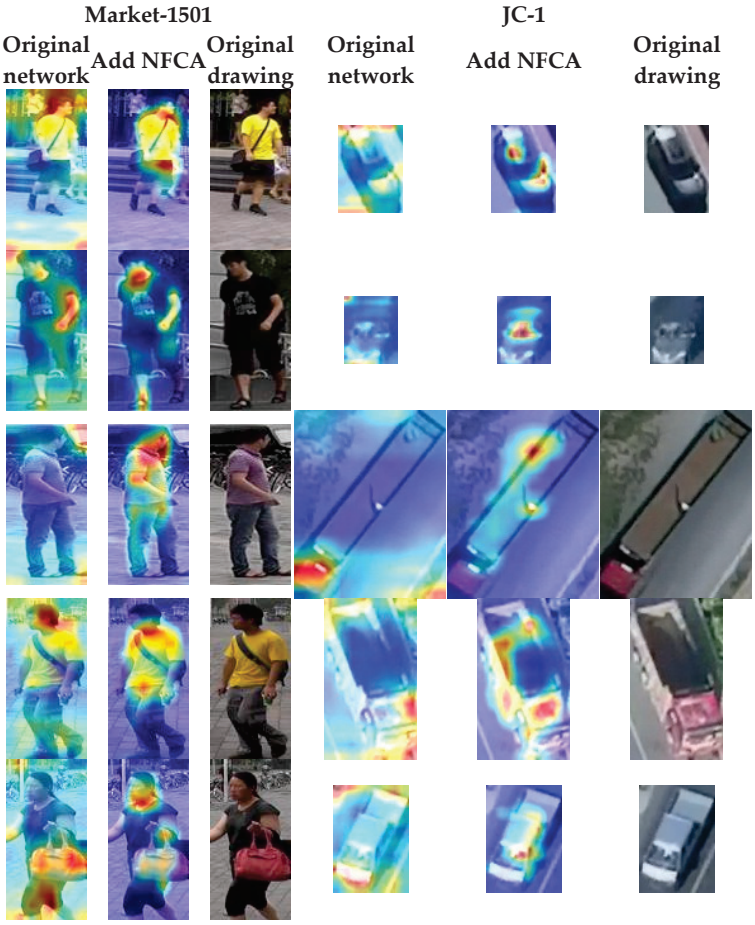


Figure 4. Comparison of feature heatmaps.

Combined with the core objectives of the ReID task (extracting discriminative features of the target, suppressing invalid background interference, and ensuring the robustness of features to viewpoint changes and occlusions), this paper defines a three-dimensional evaluation criterion for high-quality ReID heatmaps:

Discriminative Region Focus: The high-response regions (red/yellow highlighted areas) of the heatmap should be accurately concentrated on the core discriminative parts of the target (such as the clothing contour of pedestrians, the body and identification areas of vehicles), rather than scattered in the background or non-critical local details, to ensure the inter-class discriminability of features.

Background Noise Suppression: The response intensity of the background area should be significantly lower than that of the target main body, without large-area invalid high response, to verify the network's ability to filter background interference and avoid invalid information dominating feature encoding.

Target Integrity: The high-response regions should cover the complete main structure of the target, rather than only focusing on scattered local details, to ensure the robustness of features to viewpoint changes and local occlusions, and avoid matching errors caused by the loss of local information.

Based on the above evaluation criteria, the comparative analysis of the heatmap results in Figure 4 is as follows:

https://doi.org/10.3390/computation14050096

For the original network without the NFCA module, the high-response regions of the heatmap are scattered, mostly concentrated in the background areas (such as the mall environment in Market-1501, the road and vegetation areas in JC-1), and only respond to scattered local parts of the target, unable to cover the complete main body of the target, which completely fails to meet the evaluation criteria of high-quality heatmaps. This also explains the significant drop in mAP after removing the NFCA module in the ablation experiment: the network is easily disturbed by background noise and cannot extract stable discriminative features of the target.

For the proposed method with the NFCA module, the performance of the heatmap fully complies with the high-quality evaluation criteria: the high-response regions are accurately focused on the target body of pedestrians and vehicles, without invalid highlighting in the background; the response regions completely cover the core discriminative structure of the target, rather than scattered parts; the response intensity of the background area is significantly suppressed.

The visualization results verify that the proposed NFCA module can effectively guide the network to focus on the core discriminative regions of the target, suppress background noise, and retain the complete structural information of the target. It improves the discriminability and robustness of features from the underlying logic of feature encoding, forming a complete logical closed loop with the performance improvement of the previous quantitative experiments.

4.6. Generalization Verification on Classification Task

To verify the adaptability of the proposed NFCA module to low-pixel images and its versatility in other computer vision tasks, this paper carries out generalization verification experiments on the CIFAR-100 image classification dataset. The experimental configuration is as follows: Batch Size 128, total training epochs 150, initial learning rate 0.01, SGD optimizer with momentum 0.9, weight decay 5×10^{-4}. The experimental results are shown in Table 4.

Table 4. Comparative experiments on classification tasks.

Model	CIFAR-100 mAP (%)
ResNet34 + CA	75.07
ResNet34 + ECA	76.91
ResNet34 + CBAM	77.02
Our	77.42

It can be seen from the experimental results that, as the core component of the ReID framework, the proposed NFCA module achieves the best Top-1 accuracy on the CIFAR-100 classification task, with an improvement of 0.4% compared with the second-best CBAM module. This result proves that the proposed method not only has excellent performance in ReID tasks but also can adapt to other computer vision tasks such as image classification, and still has a good feature enhancement effect in low-pixel scenarios, verifying the versatility and advancement of the method.

In summary, the proposed method comprehensively improves the performance and generalization ability of the ReID network by enhancing feature robustness through the attention mechanism, fitting nonlinear distribution through adaptive metrics, and reducing distribution differences through multi-source data fusion.

https://doi.org/10.3390/computation14050096

5. Conclusions

To address the core challenges of air-to-ground target ReID, this paper proposes a novel ReID framework based on neighborhood feature centralization attention. Through the three-dimensional optimization chain of "attention mechanism-adaptive metric-multi-source data supplementation", the framework systematically solves the bottlenecks of existing methods in air-to-ground scenarios: the NFCA module enhances cross-feature semantic interaction while retaining precise position encoding, improving the robustness of features to viewpoint changes, occlusions and illumination variations; the end-to-end nonlinear metric learning breaks through the limitations of traditional linear metrics; the multi-source data fusion strategy expands the coverage of the feature space without additional annotation cost, and enhances cross-domain generalization ability. Experimental results show that the proposed method achieves leading performance on the self-developed air-to-ground dataset JC-1 and two public ReID datasets Market-1501 and VehicleID. Sufficient ablation experiments, statistical significance tests and cross-domain tests verify the effectiveness and reliability of each core component. It should be objectively noted that the performance gain of the proposed method over SOTA methods in cross-target-type generalization scenarios is relatively limited, and the generalization ability for completely unseen target types still needs to be further improved.

Author Contributions: Conceptualization, T.Y.; methodology, T.Y. and Y.X.; software, T.Y.; validation, Y.M., H.Y. and H.X.; data curation, T.Y., A.W. and Y.X.; writing—original draft preparation, T.Y. and Y.X.; writing—review and editing, T.Y. and Y.X.; funding acquisition, A.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Key Laboratory of Cross-Domain Flight Interdisciplinary Technology and The APC was funded by Key Laboratory of Cross-Domain Flight Interdisciplinary Technology.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liang, X.; Rawat, Y.S. DIFFER: Disentangling Identity Features via Semantic Cues for Clothes-Changing Person Re-ID. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 10–17 June 2025*; IEEE/CVF: Piscataway, NJ, USA, 2025; pp. 13980–13989.
2. Nguyen, K.; Fookes, C.; Sridharan, S.; Nguyen, H.; Liu, F.; Liu, X.; Rezaei, S.; Ross, A.; Endrei, T.; DeAndres-Tame, I.; et al. AG-VPReID 2025: Aerial-Ground Video-Based Person Re-Identification Challenge Results. In *Proceedings of the 2025 IEEE International Joint Conference on Biometrics (IJCB), Washington, DC, USA, 22–25 September, 2025*; IEEE: Piscataway, NJ, USA, 2025; pp. 1–10.
3. Wu, Y.; Wang, X.; Yang, X.; Liu, M.; Zeng, D.; Ye, H.; Li, S. Learning Occlusion-Robust Vision Transformers for Real-Time UAV Tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 11–15 June 2025*; IEEE/CVF: Piscataway, NJ, USA, 2025; pp. 17103–17113.
4. Liu, X.; Qi, J.; Chen, C.; Bin, K.; Zhong, P. UCM-VeID V2: A Richer Dataset and A Pre-training Method for UAV Cross-Modality Vehicle Re-Identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 11–15 June 2025*; IEEE/CVF: Piscataway, NJ, USA, 2025; pp. 22286–22295.
5. Narayana, G.V.; Swain, S.P.; Pattnayak, D.; Pradhan, M.R.; Krishna, P.A. Intelligent Drone Patrolling with Real-Time Object Detection and GPS-Based Path Adaptation. *Eng. Proc.* **2026**, *124*, 82. [CrossRef]
6. Xiang, H.; Wang, P.; Xu, X.; Yi, K.; Zhang, X.; Sheng, Q.Z.; Fan, W. MTP: Exploring Multimodal Urban Traffic Profiling with Modality Augmentation and Spectrum Fusion. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI 2026), Phoenix, AZ, USA, 12–17 February 2026*; AAAI Press: Palo Alto, CA, USA, 2026; Volume 40, pp. 27037–27045. [CrossRef]
7. Shao, M.; Zhang, Z.; Wang, Y.; Dai, Y.; Shen, X.; Wang, X. HyperD: Hybrid Periodicity Decoupling Framework for Traffic Forecasting. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI 2026), Phoenix, AZ, USA, 12–17 February 2026*; AAAI Press: Palo Alto, CA, USA, 2026; Volume 40, pp. 15689–15697. [CrossRef]

https://doi.org/10.3390/computation14050096

8. Zhong, C.; Liu, B.; Zhu, W.; Dai, D.; Jiang, Y. Phase-Aware Hierarchical Reinforcement Learning with Dynamic Human–AI Authority Allocation for Mountain Search and Rescue. *Drones* **2026**, *10*, 229. [CrossRef]
9. Yuan, C.; Zhang, G.; Ma, C.; Zhang, T.; Niu, G. From Poses to Identity: Training-Free Person Re-Identification via Feature Centralization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 10–17 June 2025; pp. 24409–24418.
10. Chopra, S.; Hadsell, R.; LeCun, Y. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2005; pp. 539–546.
11. Schroff, F.; Kalenichenko, D.; Philbin, J. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2015; pp. 815–823.
12. Wang, H.; Xiang, Y.; Gong, S.; Tao, D. Deep Metric Learning via Lifted Structured Feature Embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2016; pp. 3003–3012.
13. Wang, F.; Cheng, J.; Liu, W.; Liu, H. Additive Margin Softmax for Face Verification. *IEEE Signal Process. Lett.* **2018**, *25*, 926–930. [CrossRef]
14. Wang, F.; Xiang, T.; Gong, S.; Zhou, X. NormFace: L2 Hypersphere Embedding for Face Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2018; pp. 1141–1149.
15. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2021; pp. 11603–11612.
16. Li, J.; Wang, W.; Hu, X.; Yang, J. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction Without Convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Piscataway, NJ, USA, 2021; pp. 568–578.
17. Sun, Y.; Zheng, L.; Yang, Y.; Tian, Q. Beyond Part Models: Person Re-Identification with Refined Part Pooling (and a Strong Convolutional Baseline). In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2018; pp. 480–496.
18. Ganin, Y.; Lempitsky, V. Domain-Adversarial Neural Networks. *J. Mach. Learn. Res.* **2016**, *17*, 2096–2130.
19. Long, M.; Cao, Y.; Wang, J.; Jordan, M.I. Conditional Adversarial Domain Adaptation. In *Advances in Neural Information Processing Systems*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31, pp. 1640–1650. [CrossRef]
20. Zhang, Y.; Li, X.; Wang, X.; Liu, W. Cross-Domain Person Re-Identification Based on Feature Fusion Invariance. *Appl. Sci.* **2024**, *14*, 4644. [CrossRef]
21. Wei, L.; Zhang, S.; Sun, Y. Person Transfer GAN to Bridge Domain Gap for Person Re-Identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2018; pp. 79–88.
22. Li, H.; Chen, J.; Zheng, A.; Wu, Y.; Luo, Y. Day-Night Cross-domain Vehicle Re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024*; IEEE: Piscataway, NJ, USA, 2024; pp. 12626–12635. [CrossRef]
23. Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A. A Theory of Learning from Different Domains. *Mach. Learn.* **2010**, *79*, 151–175. [CrossRef]
24. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1995.
25. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; Tian, Q. Market-1501: A Dataset and Benchmark for Large-Scale Person Re-Identification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*; IEEE: Piscataway, NJ, USA, 2015; pp. 3908–3916.
26. Liu, H.; Tian, Y.; Wang, Y.; Pang, L.; Huang, T. Deep Relative Distance Learning: Tell the Difference between Similar Vehicles. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 2167–2175. [CrossRef]
27. Ge, Y.; Song, X.; Huang, X.; Wang, X. FastReID: A Deep Learning Toolbox for Person Re-Identification. *arXiv* **2020**, arXiv:2006.02631.
28. Li, X.; Zhu, X.; Liu, H.; Wang, X. TransReID: Transformer-Based Object Re-Identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Piscataway, NJ, USA, 2021; pp. 14035–14045.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/computation14050096

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

MDPI Books Editorial Office
E-mail: books@mdpi.com
www.mdpi.com/books



Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Topic Editors. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editors and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.





Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-8840-5