

Article

Adaptive Polar Lights Optimizer for Smart Electric Vehicle Charging Under Price Uncertainty and Battery Degradation

Abdelkrim Benmoulai ¹, Salah Kamel ² and Francisco Jurado ^{1,*}

¹ Department of Electrical Engineering, Escuela Politécnica Superior de Linares, University of Jaen, 23071 Jaen, Spain; abenmoul@ujaen.es

² Department of Electrical Engineering, Faculty of Engineering, Aswan University, Aswan 81542, Egypt; skamel@aswu.edu.eg

* Correspondence: fjurado@ujaen.es

Abstract

This paper investigates an algorithmic redesign tailored to cost minimization with degradation awareness EV charging under an uncertainty framework for coordinated grid-to-vehicle (G2V) and vehicle-to-grid (V2G) scheduling. An improved variant of the Polar Lights Optimizer (IPLO) is developed through the integration of Random Walk Exploitation (RWE) to enhance local refinement and Periodic Random Parameter Tuning (PRPT) to improve adaptability under uncertainty. In addition, an adaptive control mechanism is incorporated to adjust charging and discharging actions based on battery capacity degradation and dynamic electricity price signals. The presented framework is evaluated through simulation-based case studies and compared with several recent metaheuristic algorithms. The results demonstrate cost reductions of up to 25.42% over the original PLO and 80.78% relative to a non-optimized baseline, faster convergence, and improved robustness to price uncertainty, while mitigating adverse battery degradation effects. A statistical analysis over multiple independent runs confirms the reliability and consistency of the presented approach, highlighting its suitability for smart EV charging optimization in dynamic operating environments.

Keywords: EV; optimization; IPLO; scheduling; V2G/G2V; RWE; PRPT; battery health; cost reduction

Academic Editors: Xuguang Hu,
Yang Liu and Guangliang Liu

Received: 30 December 2025

Revised: 28 January 2026

Accepted: 4 February 2026

Published: 9 February 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and con-
ditions of the [Creative Commons At-
tribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

1.1. Literature Review

The rapid adoption of electric vehicles (EVs) has transformed the transportation sector and significantly increased the complexity of charging operations, creating an urgent need for efficient and cost-effective charging strategies capable of handling fluctuating electricity prices and battery degradation. Smart charging strategies that incorporate battery degradation and uncertainty in demand and electricity prices can significantly reduce both expected energy costs and degradation compared to uncontrolled charging frameworks [1]. The integration of V2G technology, which enables EVs to discharge stored energy back to the grid, introduces new opportunities to reduce operational costs, enhance energy efficiency, and support grid flexibility. Realizing these benefits requires a smart and coordinated scheduling strategy between G2V and V2G operations.

However, achieving effective coordination remains challenging due to the nonlinear, multimodal, and high-dimensional nature of EV charging optimization, particularly under uncertainty. Traditional optimization techniques have been applied to this problem but often suffer from limited scalability and premature convergence. Metaheuristic algorithms offer greater flexibility and search capability; nevertheless, many existing approaches lack sufficient robustness and adaptability when exposed to dynamic electricity prices and evolving battery conditions. To address these challenges, this study investigates an IPLO which integrates RWE and PRPT to enhance convergence behavior, adaptability, and robustness. The presented approach aims to reduce charging costs while preserving battery health under dynamic operating conditions.

1.2. Related Work

The rapid growth of EVs has introduced new operational challenges, most notably increased charging costs driven by grid congestion and peak demand [2]. To mitigate these issues, several strategies have been presented. Off-peak charging shifts demand to low-price periods [3], V2G systems enable EVs to discharge energy back to the grid during peak hours [4] and coordinated G2V/V2G scheduling aims to jointly maximize economic benefits and minimize charging costs [5].

However, many of these approaches overlook two critical aspects. First, frequent charging and discharging accelerate battery degradation, reducing service life and long-term economic value [6]. Second, electricity price volatility introduces significant uncertainty into operational decisions [7]. Effective EV charging scheduling must therefore balance short-term economic gains with battery longevity under dynamic conditions.

Traditional optimization techniques, including stochastic dynamic programming [8], robust stochastic programming, chance-constrained programming, and fuzzy programming [9], have been widely applied to address uncertainty and degradation. Nevertheless, these methods are highly sensitive to problem structure and often scale poorly when dealing with nonlinear, high-dimensional, and multi-modal problems. Metaheuristic algorithms—such as the improved Marine Predator Algorithm [10], Competitive Swarm Optimizer, Bat Algorithm, and Genetic Algorithm [5] offer greater flexibility but often rely on fixed algorithmic structures, limiting robustness and solution quality in dynamic environments.

Recent studies have shown that hybridization, adaptive mechanisms, and mathematical enhancement models can substantially improve metaheuristic performance by aligning algorithm behavior more closely with problem characteristics. EV charging optimization is inherently nonlinear, constrained, high-dimensional, and driven by uncertainty. Addressing such complexity requires an optimizer that combines strong exploration, effective exploitation, adaptability, and robustness.

Motivated by these requirements, this study builds upon the PLO. Inspired by the stochastic yet guided dynamics of polar lights, PLO employs random perturbations and directional movements to explore the global search space and reduce the risk of premature convergence, making it initially suitable for EV charging optimization. While recent work has begun applying recent metaheuristics to EV charging, such as the War Strategy Optimization (WSO) algorithm introduced in 2024 to minimize charging costs and queue times under realistic constraints, the application of PLO to EV charging optimization remains unexplored.

The suitability of PLO for EV charging optimization stems from the structural compatibility between its native search mechanisms and the characteristics of the problem. The aurora-inspired oval walk produces guided yet stochastic trajectories that reflect the non-stationary and time-varying nature of electricity prices in real-time charging environ-

ments. Likewise, the gyratory motion mechanism aligns well with the multi-modal structure of the EV charging cost landscape, which arises from interacting constraints such as state-of-charge limits, battery degradation effects, and bidirectional G2V/V2G decisions. These mechanisms enable PLO to navigate complex cost surfaces effectively while maintaining population diversity.

Despite these advantages, the original PLO exhibits an imbalanced exploration–exploitation trade-off that limits convergence speed and solution precision. Although gyratory motion is designed to enhance exploitation, in practice it mainly guides population movement toward promising regions rather than performing intensive local refinement. As a result, exploitation remains coarse and strongly dependent on fixed control parameters [11,12]. These limitations become more pronounced in dynamic optimization problems, where objective landscapes evolve over time due to electricity price volatility and system uncertainty.

To address these shortcomings, this study introduces IPLO through two complementary mechanisms. First, a Gaussian Random Walk-based Exploitation (RWE) strategy is incorporated to strengthen local search. By applying small, normally distributed perturbations around promising solutions, RWE enables fine-grained local refinement, accelerates convergence, and enhances solution precision while complementing PLO's global exploration capability [13,14]. Unlike external local search hybrids, RWE is embedded within PLO's native stochastic motion, preserving algorithmic coherence.

Second, the perturbation-based PRPT mechanism is adopted to introduce self-adaptation into PLO. Metaheuristic performance is strongly influenced by control parameters, and fixed parameter settings often result in inconsistent behavior in complex, real-world applications. Adaptive parameter control strategies dynamically adjust parameters during the search based on performance feedback, significantly improving robustness and efficiency [15,16]. In PRPT, key PLO parameters, specifically the gyratory weights and damping factor, are periodically perturbed within adaptive ranges. This approach directly addresses the fixed-parameter imbalance of the original PLO and is consistent with adaptive tuning frameworks validated in prior studies, such as Chemical Reaction Optimization, where perturbation-based adaptive control achieved statistically significant performance gains without requiring pilot tuning [17].

Importantly, the IPLO differs from existing hybrid or adaptive metaheuristics in how enhancement is realized. Rather than introducing additional operators or new hyper parameters, both RWE and PRPT operate on PLO's intrinsic mechanisms, enabling problem-driven, online adaptation. This design is explicitly motivated by the non-stationary nature of EV charging optimization, where electricity prices and battery health evolve continuously over time.

In the context of EV charging cost optimization, the integration of RWE and PRPT enables continuous adjustments of the exploration–exploitation balance in response to changing system conditions. As a result, IPLO achieves faster convergence, improved solution quality, and greater adaptability under uncertainty. Collectively, these enhancements mitigate the structural limitations of the original PLO and establish IPLO as an effective and well-suited optimizer for uncertainty-driven EV charging optimization.

To handle electricity price volatility and battery degradation uncertainty without resorting to detailed electrochemical aging models, this study adopts a price-responsive adjustment and degradation-aware control mechanism. Real-time pricing signals are widely used in EV charging optimization to influence charging and discharging decisions, encouraging charging during low-price periods and discharging during high-price periods [18]. Smooth and bounded price–response mappings are employed to avoid discontinuous control actions and ensure operational stability in practical energy management systems [19]. Field trials in the ERCOT region have demonstrated that such dynamic pricing

strategies can improve grid performance and reduce EV charging costs by approximately 8% compared to static tariffs [20].

Accordingly, this work employs bounded exponential mappings that scale charging and discharging increments based on dynamic electricity price levels, electricity price uncertainty is modeled using a deterministic scenario-based disturbance approach rather than a probabilistic forecast error model. Such scenario-based representations have been widely used in EV charging and scheduling literature to model external uncertainties without requiring full distributional forecasts or stochastic process assumptions. Specifically, a baseline electricity price trajectory is defined, and controlled price variations are introduced at selected time intervals to simulate unexpected price changes. These scenarios represent deviations that may arise from forecast updates, market volatility, or real-time pricing adjustments. The objective is not to estimate expected costs under a price distribution, but to evaluate the algorithm's adaptive and robust response to sudden price variations, aligning with the control-oriented nature of online EV charging scheduling serving as a surrogate control law that ensures stable and realistic operational behavior [20]. While effective, this price-responsive adjustment may concentrate demand during low-price periods, assumes a relatively high user elasticity, depends on price forecast accuracy, and does not explicitly model local grid congestion.

For battery health, the presented framework adopts a two-level degradation-aware strategy. First, a lightweight degradation indicator tracks capacity evolution based on SOC variations, inspired by throughput-based aging models widely used in scheduling studies [21]. Second, and more importantly, this information is exploited by an adaptive operational mechanism that dynamically adjusts charging and discharging rates in response to observed capacity loss. This control-oriented formulation enables a proactive mitigation of degradation effects rather than merely penalizing them, making it particularly suitable for online EV charging scheduling under uncertainty.

This approach aligns with recent studies that integrate battery health indicators, such as the state of health (SOH), directly into decision-making and optimization processes [22,23]. Health-aware adaptive scheduling frameworks have been shown to improve lifetime performance and reduce battery stress compared to cost-only strategies, particularly under V2G operation and uncertainty [24].

Despite its advantages, the presented degradation-aware adjustment mechanism has limitations. Specifically, the objective is health-aware operational adaptation rather than explicit lifetime degradation modeling. While this enables computationally efficient and responsive real-time control, it does not aim to predict long-term aging trajectories or remaining useful life. Additionally, the simplified heuristic nature of the adaptive adjustment rules, though well suited for online scheduling, may not guarantee optimality under extreme operating conditions such as highly irregular driving patterns or dominant fast-charging behavior.

1.3. Contribution

This study responds to the need for robust and effective optimization tools to address the substantial and operational challenges, including high charging costs, battery degradation, and grid stress under dynamic electricity prices, as the widespread deployment of electric vehicles EVs is reshaping both transportation and power systems by proposing an IPLO that integrates a balanced exploration–exploitation and more adaptability for better handling of the challenge. The main contributions of the study are:

- A problem-driven enhancement of the PLO for EV charging cost optimization under battery degradation and price fluctuation, due to its unique physics-inspired dual-motion strategy gyration and aurora oval walks, combined with particle collision for

escaping local optima, making it suitable for nonlinear, multi-modal, and high-dimensional continuous optimization problems.

- Performance analysis reveals that the original PLO provides coarse local exploitation, limiting the convergence precision in dynamic EV charging optimization. The RWE mechanism enhances localized search accuracy and accelerates convergence, leading to improved cost minimization. Meanwhile, PRPT improves robustness and prevents premature convergence. These enhancements collectively increase the compatibility of PLO with uncertainty-driven EV charging scheduling.
- A simplified control-oriented, price-degradation-aware mechanism that adaptively scales EV charging and discharging actions based on a deterministic scenario-based electricity price disturbance approach and a capacity-based battery degradation indicator, enabling cost reduction while mitigating degradation under uncertainty.
- The IPLO demonstrates improved performance and robustness under electricity price uncertainty, achieving consistently lower charging costs, faster and smoother convergence, and reduced sensitivity to price variations compared to the original PLO and several recent metaheuristic algorithms. Extensive comparative experiments and statistical analyses confirm the improved solution stability across multiple runs, while the adaptive mechanisms enable the algorithm to maintain a reliable performance as price signals and operating conditions vary.

The remainder of this paper is organized as follows: Section 1 introduces the research topic. Section 2 formulates the problem and describes the structure of the IPLO algorithm. Section 3 presents the experimental results, Section 4 discusses the findings. Section 5 concludes the paper.

2. Materials and Methods

2.1. Problem Formulation

Modern electric vehicles operate in two modes, G2V and V2G, enabling flexible energy use and cost reduction. This study presents an optimization framework using the PLO, improved RWE and PRPT to manage EV charging and discharging efficiently while considering battery health and electricity price fluctuations.

As illustrated in Figure 1, the system begins with a battery voltage of 384 V and a capacity of 60 Ah, ensuring smooth performance and extended range. With a 90% charging efficiency and 100% discharging efficiency, the battery supports optimal energy utilization [10].

Starting at a 30% state of charge (SOC), the battery is charged in nine intervals with varying electricity prices. The goal is to reach an 80% SOC while minimizing charging and discharging costs [10]. The PLO-based framework, supported by RWE and PRPT, ensures cost-effective and adaptable optimization. By incorporating electricity price variability and battery degradation into the cost function, the model produces robust solutions that reduce costs and support long-term battery sustainability under dynamic conditions.

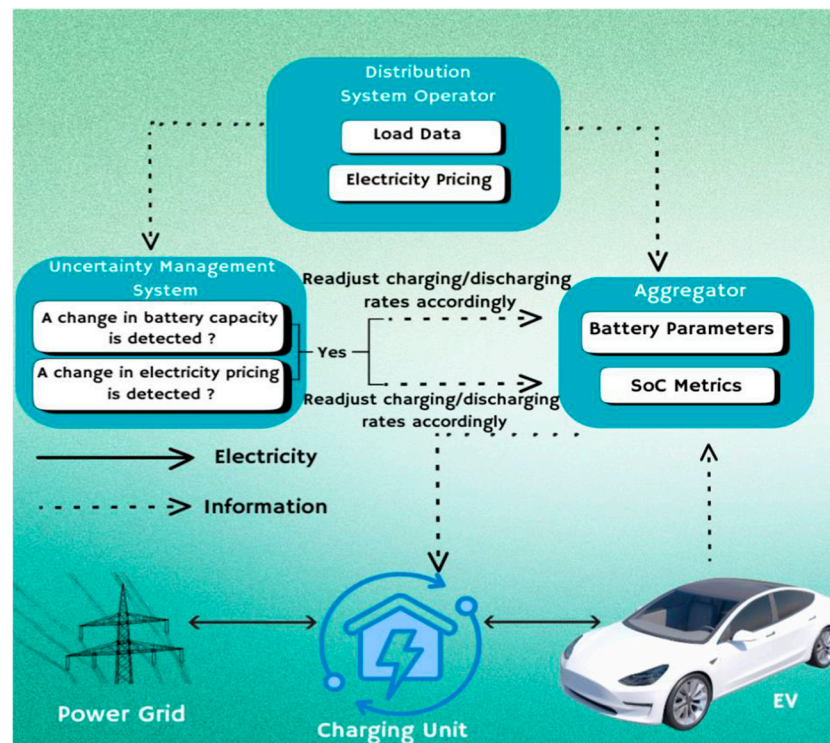


Figure 1. Illustration of the electric vehicle charging and discharging infrastructure.

$$F(k) = \min \sum_{k=1}^n J_k \quad (1)$$

$$J_k = C_k + D_k \quad (2)$$

$$C_k = \left(\frac{V(S_k - S_{k-1}) \varphi_k \phi_k Q R_k}{n} \right) \left(\frac{1 + S_k}{2} \right) S_{bk}^2 \quad (3)$$

$$D_k = (V(S_{k-1} - S_k) \varphi_k \phi_k Q R_k) \left(\frac{-1 + S_k}{2} \right) S_{bk}^2 \quad (4)$$

$$\varphi_k = \begin{cases} \gamma_{ch}(k), & \text{charging} \\ \gamma_{dis}(k), & \text{discharging} \\ 1, & \text{other} \end{cases} \quad (5)$$

$$\phi_k = \begin{cases} \lambda e^{-\rho_k}, & \text{charging} \\ \lambda e^{\rho_k}, & \text{discharging} \end{cases}, \quad \rho_k = \frac{R_k - R_{min}}{R_{max} - R_{min}} \quad (6)$$

$$S_{bk} = \begin{cases} 1, & \text{charging} \\ -1, & \text{Discharging} \end{cases} \quad (7)$$

In equations [10], V represents the voltage of the battery, indicating the electrical potential difference between its terminals. The symbol Q denotes the battery's capacity, measured in Ampere-Hours (Ah), representing n , the maximum charge the battery can store. The charging efficiency is symbolized by reflecting how effectively electrical energy from the power source is converted into stored energy within the battery. The SOC for the k_{th} interval is denoted by S_k , indicating the battery's current charge level during that specific time interval. The variable n represents the total number of time intervals, with k referring to a particular interval within that time frame. The cost of charging at the k_{th}

interval is represented by C_k , while D_k indicates the cost of discharging for the same interval. The total cost for the k_{th} interval is denoted by J_k , which is the sum of both charging and discharging costs. The selection variable for each interval, S_{bk} , is used to determine whether charging or discharging occurs during the k_{th} interval. A value of 1 corresponds to charging, G2V, while a value of -1 signifies discharging, V2G.

The proposed formulation adopts a control-oriented, degradation-aware operational model for EV charging scheduling. The objective is not to predict battery lifetime but to mitigate degradation in real time by adapting charging and discharging behavior based on observed battery health evolution. This choice is motivated by the requirements of online EV charging optimization under uncertainty, where computational efficiency, adaptability, and responsiveness are essential. Battery degradation is therefore represented using a lightweight capacity-based indicator, which enables health-aware decision making without relying on detailed electrochemical aging models that rely on Arrhenius kinetics and SOC-dependent acceleration to accurately quantify calendar and cycle aging under controlled conditions [24], as they require more chemistry-specific parameters that are beyond the scope of real-time EV charging optimization and are therefore not adopted in this study. A comparison between the degradation-aware cost formulation and classical battery degradation models is presented in Table 1. Battery health degradation is represented using a capacity-based degradation indicator, defined as:

$$Q(i+1) = Q(i) - \sigma (|\Delta S_{ch} + \Delta S_{dis}|) \quad (8)$$

where σ is a degradation sensitivity coefficient and ΔS_{ch} and ΔS_{dis} denote the charging and discharging SOC variations, respectively. The coefficient σ is selected to reproduce the order of magnitude of the capacity fade observed in real EV batteries, which typically lose approximately 1.8–2.3% of capacity per year under normal usage. This corresponds to a capacity loss on the order of 10^{-4} per SOC cycle, making σ an effective degradation proxy suitable for scheduling-level optimization rather than an exact electrochemical aging parameter. Battery health is further quantified through a normalized degradation indicator:

$$\delta_{deg} = \frac{Q - Q(k)}{Q} \quad (9)$$

representing the relative state of health (SOH) loss and providing a continuous and interpretable measure of degradation severity. Rather than penalizing degradation directly in the objective function, the proposed approach introduces an adaptive operational response that reshapes SOC trajectories when degradation is detected. The charging and discharging increments are adjusted as follows:

$$\Delta S_{adj} = \gamma(k) \Delta S \quad (10)$$

with SOH-aware SOC regulation factors defined by:

$$\gamma_{ch}(k) = 1 - k_{ch} \delta_{deg} \quad (11)$$

$$\gamma_{dis}(k) = 1 - k_{dis} \delta_{deg} \quad (12)$$

where $k_{dis} > k_{ch}$ enforces a stronger attenuation on discharging actions. This asymmetry reflects the higher degradation impact of discharging, particularly under V2G operation, which increases lithium plating risk, internal resistance growth, and depth-of-discharge stress, thereby altering the battery aging pattern and increasing the contribution of cyclic aging to total capacity loss [25]. Through this mechanism, the optimizer proactively limits aggressive SOC transitions as battery health deteriorates, achieving degradation mitigation while preserving responsiveness to electricity price signals.

Table 1. Comparison between the presented degradation-aware cost formulation and classical battery degradation models.

Aspect	Cycle-Life/DoD-Based Degradation Models	Presented Degradation-Aware Cost Formulation
Aging representation	Explicit modeling of battery degradation as a function of depth-of-discharge (DoD), equivalent full cycles, or empirically derived cycle-life curves.	Control-oriented degradation indicator based on observable capacity evolution and SOC variations.
Primary objective	Quantifies long-term capacity fade and remain useful life.	Enables degradation-aware operational decision making in real time.
Model output	Absolute degradation metrics (e.g., capacity loss, lifetime estimation).	Relative degradation awareness embedded directly into the cost function.
Responsiveness to uncertainty	Limited; typically assumes static or slowly varying operating conditions.	High; dynamically adapts charging/discharging behavior under price and capacity fluctuations.
Computational complexity	Moderate to high due to empirical fitting or physics-based modeling.	Low; suitable for online and metaheuristic optimization.
Integration into optimization	Well suited for offline planning and long-term scheduling.	Well suited for real-time and uncertainty-driven EV charging optimization.
Adaptability during operation	Passive (degradation is evaluated, not actively mitigated).	Active (charging/discharging rates are adjusted to mitigate degradation).
Strengths	Physically interpretable, chemistry-aware, accurate for lifetime analysis.	Simple, robust, computationally efficient, compatible with dynamic pricing and metaheuristics.
Limitations	Require detailed battery parameters; limited scalability for real-time use.	Does not explicitly predict long-term battery lifetime; provides relative aging assessment.

To keep up with price fluctuations, an adaptive model is incorporated with an exponential term that supports charging during low-price periods and discharging during high-price periods, improving economic efficiency; ϕ_k normalizes current price within the dynamic range of $[0, 1]$ to control the intensity of adjustment, R_k is the electricity price at the current interval, $R_{\min}$ and $R_{\max}$ are the minimum and maximum observed prices, and ϕ_k is the price scaling factor that adjusts the decision based on whether the interval corresponds to charging or discharging. To preserve battery health and extend its lifespan, several constraints must be considered during the optimization process. The following constraints ensure that the battery operates within safe and efficient parameters, balancing energy costs and battery longevity [10]:

Constraint 1: (Limit values for the SOC): The SOC of the battery should be kept within certain limits to avoid overcharging or deep discharging, which can shorten the life of the battery. In this case, the minimum SOC value should not fall below a certain threshold (30%), while the maximum SOC value should not exceed a predefined value (80%).

$$SOC_{\min} \leq S_k \leq SOC_{\max} \quad (13a)$$

Constraint 2 states that the SOC must be sufficient to cover the energy requirements of the next journey and avoid the restriction that the state of charge is above the minimum SOC value.

$$SOC_{n+1} = SOC_{final} + SOC_{min} \quad (13b)$$

Constraint 3: The charging decisions can be adjusted so that the SOC during each interval does not exceed the maximum chargeable SOC (40%).

$$|S_k - S_{k-1}| \leq \Delta SOC_{charge}(charging) \quad (13c)$$

Constraint 4: The discharge decisions can be controlled to prevent the SOC of the battery from reaching the maximum dischargeable value (40%) and to avoid over-discharge.

$$|S_k - S_{k-1}| \leq \Delta SOC_{discharge}(discharging) \quad (13d)$$

Constraint 5 states that there should be a minimum limit for the deviation of the SOC from one interval to the next, which is referred to as ΔSOC_{min} . This means that the SOC should not change by less than the value of ΔSOC_{min} (20%,10%) between successive intervals.

$$|S_k - S_{k-1}| \geq \Delta SOC_{min} \quad (13e)$$

2.2. Algorithm

2.2.1. Natural Phenomenon Behind Polar Lights

The polar lights (Auroras) are caused by the interaction of charged solar wind particles (mainly electrons and protons) with the Earth's magnetosphere [26]. These particles are directed toward the polar regions by the magnetic field lines, following a spiraling path; the phenomenon is illustrated in Figure 2.

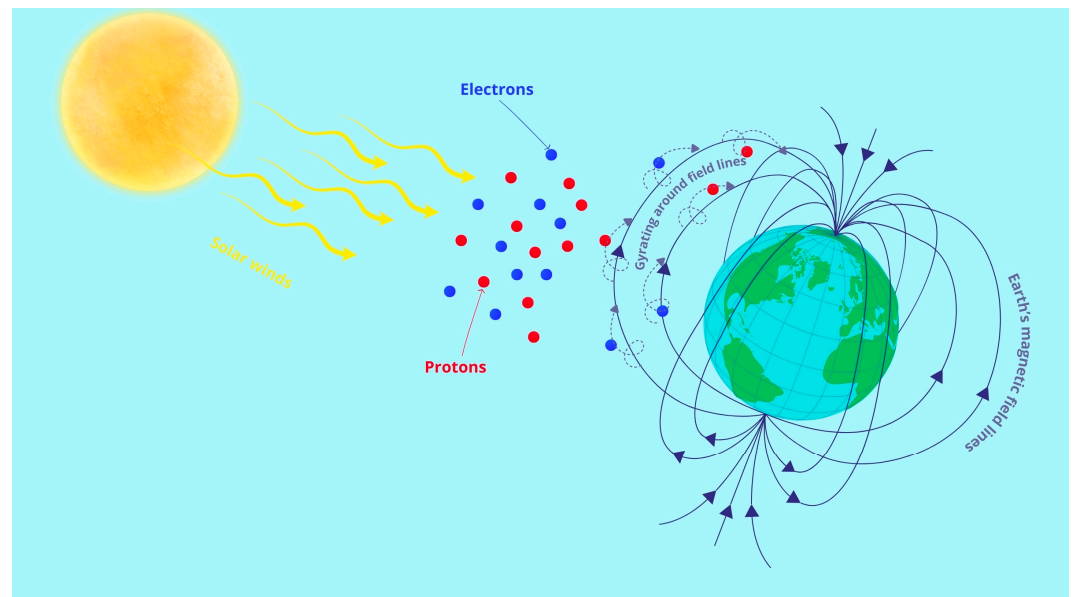


Figure 2. Solar wind particles interacting with Earth's magnetic field.

1. **Solar Wind Interaction:** As solar particles travel from the Sun toward Earth, they encounter the Earth's magnetic field, which traps and channels them along magnetic lines toward the poles. This motion is governed by the Lorentz force and is represented as a gyratory or spiral trajectory.
2. **Energy Transfer and Collisions:** Upon entering the atmosphere, the particles collide with oxygen and nitrogen atoms, transferring energy and exciting the atoms to higher energy states.
3. **Emission of Light:** When these atoms return to their normal states, they release photons. The types of gas and altitude determine the emitted light's color (e.g., oxygen emits green or red, while nitrogen emits blue or purple). This stage is probabilistic, as collision frequency and energy levels vary.

4. **Spatial Distribution:** Auroras typically appear in arcs or spirals due to magnetic field dynamics and particle movements. This is influenced by magnetic re-connection and plasma behavior, creating complex, variable patterns across the sky.

These phenomena showcase a combination of deterministic behavior (particles guided by magnetic lines) and randomness (energy exchange and photon emission).

2.2.2. Polar Lights Optimization Algorithm

Inspired by this phenomenon, the PLO [26], represents search agents as high-energy particles (solar wind) moving through the magnetic field (solution space) toward the poles, creating brilliant, dynamic patterns.

Gyration motion models local exploitation through the spiral motion of particles as they align with magnetic field lines.

Auroral Oval Walk: This is a global exploration technique based on the elliptical paths of auroras, factoring in velocity, energy, and atmospheric conditions to model movement across the solution space [26].

A Particle Collision Strategy captures the continuous collisions between charged particles to enhance escaping local optima, reflecting real-world energy exchange and aurora formation dynamics.

2.2.3. Mathematical Model

1. **Initialization:** PLO starts by initializing a population of candidate solutions (particles) in a matrix with defined rows and columns. These candidates are randomly distributed to simulate the randomness of solar wind directions [25].

$$X(N, D) = LB + R \times (UB - LB) = \begin{bmatrix} X(1,1) & X(1,2) & \cdots & X(1,D) \\ X(2,1) & X(2,2) & \cdots & X(2,D) \\ \vdots & \vdots & \ddots & \vdots \\ X(N,1) & X(N,2) & \cdots & X(N,D) \end{bmatrix} \quad (14)$$

2. **Gyration Motion**

The spiraling trajectory of charged particles is modeled using the Lorentz force. As particles approach Earth, the magnetic field causes them to spiral. This force is described by a differential equation [26]:

$$m \frac{dv}{dt} = qvB \quad (15)$$

To reflect atmospheric resistance, a damping factor (α) is introduced, modifying the motion to simulate kinetic energy loss as particles slowdown in denser atmospheric layers:

$$m \frac{dv}{dt} = qvB - \alpha v \quad (16)$$

Solving this gives a velocity function with decaying energy behavior:

$$v(t) = C e^{\frac{qB - \alpha}{m}t} \quad (17)$$

The charge q , magnetic field strength B , and constant C are simplified to a value of 1, while the mass m is set to 100. The damping factor α is randomly chosen within the range [1, 1.5]. Time t is simulated per iteration. This strategy enhances local search, allowing for a precise exploitation of regions while preserving exploration capabilities.

3. **Auroral Oval Walk**

Mimicking the dynamic elliptical band of auroras, this phase enables global exploration. The changing shape of the auroral oval due to geomagnetic activity mirrors the unpredictable nature of wide search areas [26].

To model this behavior, Levy flight (LF) is integrated. LF is a random walk with step sizes following a Levy distribution, enabling particles to explore far-reaching zones in the search space effectively.

$$Levy(d) \sim |d|^{-1-\beta}, 0 < \beta \leq 2 \quad (18)$$

Here, β controls the stability of the step size d . The movement of particles in PLO is influenced by geomagnetic activity simulating the contraction and expansion of the auroral oval. The change in particle position Ao is expressed as

$$Ao = Levy(d) \times (X_{avg}(j) - X(i, j)) + LB + r_1 \times (UB - LB)/2 \quad (19)$$

$$X_{avg} = \frac{1}{N} \times \sum_{i=1}^N X(i) \quad (20)$$

where X_{avg} is the average position of all particles, $X(i, j)$ is the current position, and r_1 is a random factor. PLO combines two search patterns: the gyration motion and the auroral oval walk. The gyration motion simulates particles spiraling along the Earth's magnetic field, focusing on local search and fine adjustments to improve solution quality. The auroral oval walk, in contrast, emphasizes global exploration, allowing particles to make larger moves and explore broader regions of the solution space.

The combined motion is represented by

$$X_{new}(i, j) = X(i, j) + r_2 \times (W_1 \times v(t) + W_2 \times Ao) \quad (21)$$

where r_2 is a random interference factor, and W_1 and W_2 are adaptive weights controlling the balance between local and global search. These weights evolve with each iteration:

$$W_1 = \frac{2}{(1 + e^{-2(t/T)^4})} - 1 \quad (22)$$

$$W_2 = e^{-(2t/T)^3} \quad (23)$$

As iterations progress, W_1 increases, giving more weight to local search, while W_2 decreases, reducing the influence of global exploration. This dynamic adjustment helps balance both search phases, enabling PLO to efficiently navigate the solution space and locate optimal solutions.

4. Collision and Energy Exchange

Auroras result from high-energy particle collisions that alter speed and direction, causing energy transfer, excitation, and ionization processes that amplify auroral brightness and complexity. Inspired by this, PLO introduces a Particle Collision Strategy to simulate these chaotic interactions; Figure 3 illustrates the collision.

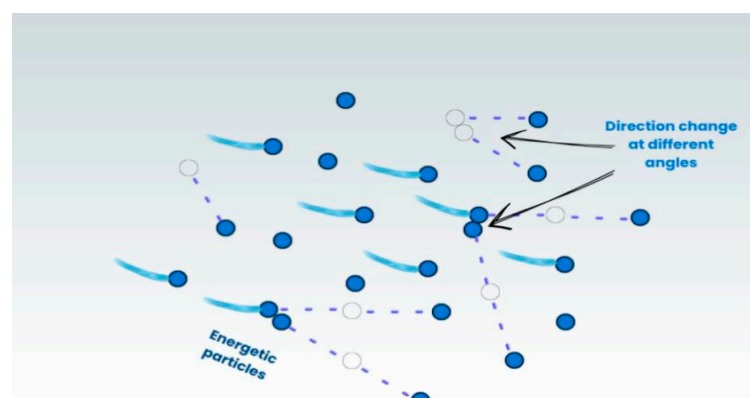


Figure 3. Particles collision.

In this strategy, particles randomly collide like in a billiard game, generating new positions and enhancing the optimizer's ability to escape local optima. As in nature, collision frequency increases over time, especially within the auroral oval, allowing for a broader solution space exploration. Mathematically, the local search term is calculated using

$$X_{new}(i, j) = X(i, j) + \sin(r_3 \times \pi) \times (X(i, j) - X(a, j)), \quad (24)$$

$$r_4 < K \text{ and } r_5 < 0.05$$

$$K = \sqrt{(t/T)} \quad (25)$$

$X(a, j)$ represents any particle in the particle cluster. Collisions between particles become more frequent as the algorithm proceeds and are therefore controlled by the collision probability K , calculated using Equation (20). r_3 and r_4 are random values, taking values in $[0, 1]$.

PLO replicates the natural phenomena of polar lights through stages of initialization (solar wind interaction), local refinement (gyratory motion and particle collisions), global exploration (Levy flight and oval walk guidance), and convergence (aurora formation), all orchestrated by equations that mirror the balance between exploration and exploitation found in nature.

2.2.4. IPLO

This paper introduces the PLO, inspired by the physical dynamics of auroras [26]. In this model, search agents represent high-energy solar particles guided by Earth's magnetic fields and atmospheric interactions as they spiral unpredictably toward the poles. Initially influenced by geomagnetic forces, these particles slow down due to atmospheric collisions, forming the auroral oval, an elliptical region that fluctuates with magnetic activity. A modified velocity equation, derived from the Lorentz force and Newton's second law, incorporates a damping factor to simulate this behavior. To balance global and local search, PLO integrates gyration motion for local exploitation and the auroral oval walk for global exploration. A Particle Collision Strategy further enhances the optimizer's ability to escape local optima through dynamic interactions. The algorithm's workflow is detailed in Algorithm 1 and visualized in Figure 4.

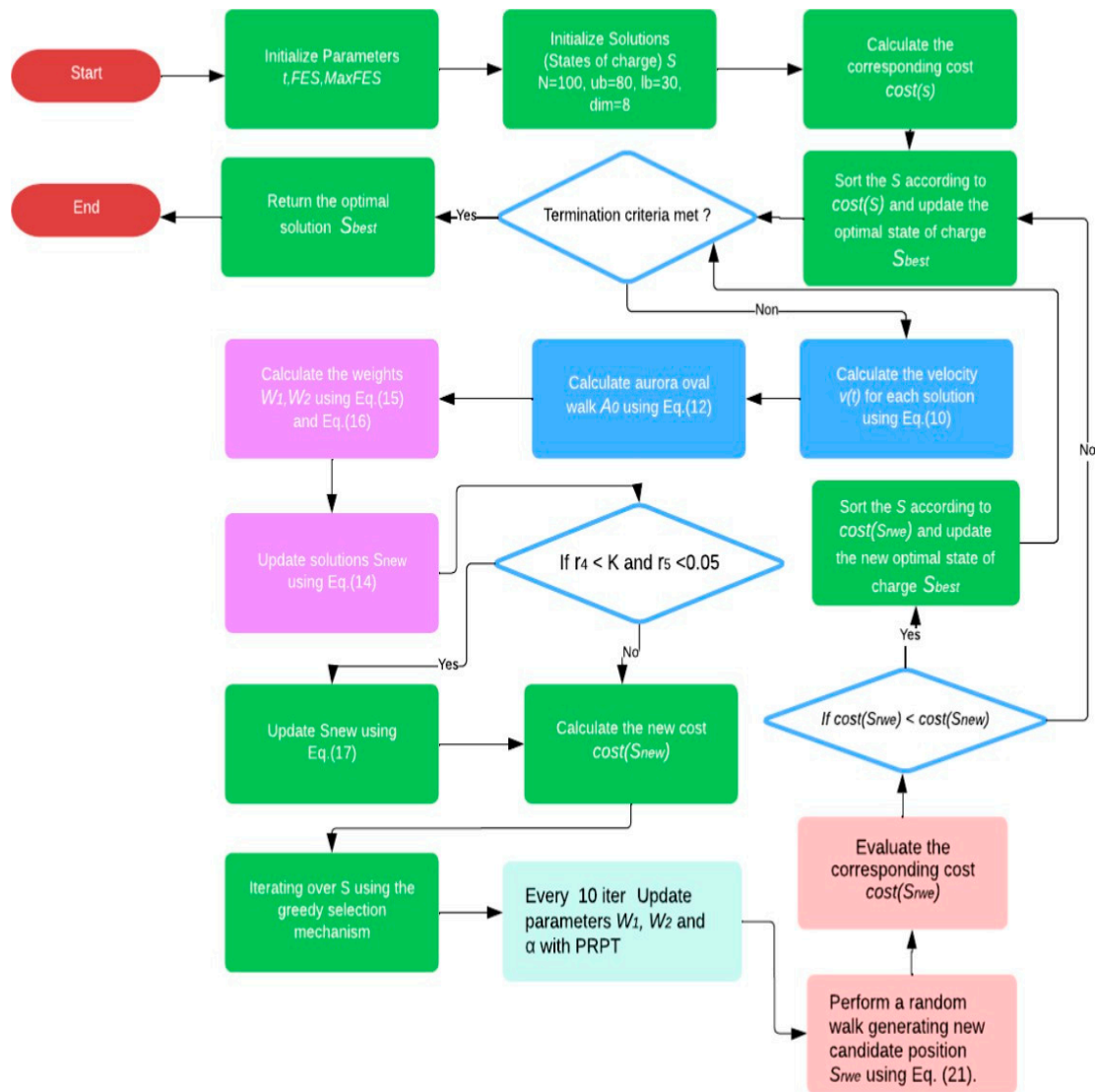


Figure 4. Flowchart of EV charging optimization using the Improved Polar Lights Optimizer.

Algorithm 1. Pseudo Code for IPLO

Parameters initializing: $FES = 0$, $MaxFES$, $t = 0$

Initialize high-energy particle cluster X .

Calculate the fitness value $f(X)$.

Sort X according to $f(X)$.

Update the current optimal solution X_{best} .

While $FES < MaxFES$

Calculate the velocity $v(t)$ for each particle, according to Equation (17).

Calculate aurora oval walk Ao for each particle, according to Equation (19).

Calculate weights W_1 and W_2 according to Equations (22) and (23).

For each energetic particle **do**

Updating particles X_{new} using Equation (21).

If $r_4 < K$ and $r_5 < 0.05$

Particle collision strategy: update particle X_{new} using Equation (24).

End If

Calculate the fitness $f(X_{new})$.

$FES = FES + 1$.

End For

If $f(X_{new}) < f(X)$

Iterating over X using the greedy selection mechanism.

End If

Parameter Refinement and Tuning (PRPT)

Every $k = 10$ iteration

Update parameters W_1 , W_2 , and α using adaptive tuning based on current fitness values using Equations (28)–(30)

Random Walk Exploitation (RWE)

For each energetic particle **do**

With a probability p (e.g., 0.2), perform a random walk to generate a new candidate position X_{RWE} using equation (27).

Evaluate the fitness $f(X_{RWE})$ and update particle X if the new candidate is better.

End For

Sort X according to $f(X)$.

Update the optimal solution X_{best} .

$t = t + 1$.

End While

Return the X_{best} .

2.2.5. Improvement

In the IPLO, two key improvements have been integrated: a random walk strategy, which has proven its effectiveness in many scenarios such as Random walk exploration for swarm mapping [27], Gaussian random walk salp swarm algorithm [13] and Random walk grey wolf optimizer [28], side by side with Parameter Tuning, which is employed to improve algorithm performance significantly.

a. Random Walk Exploitation

The RWE model is an improvement technique designed to improve the algorithm's convergence and exploitation ability. It introduces a targeted, controlled random walk phase into the optimization process, allowing the particles or agents to perform fine-grained exploitation around promising regions identified by the algorithm. The concept of a random walk as a search mechanism goes back to early metaheuristic and optimization theory. A random walk changes a solution by adding a small random step [18]. This allows particles to exploit promising areas more effectively, thereby refining the search within high-potential regions and improving convergence toward optimal solutions. Key components and equations of RWE are the following:

(1) Random step size generation

$$\Delta X = \alpha \cdot (ub - lb) \cdot \eta \quad (26)$$

Here, α is a scaling factor that equals 0.05, controlling the step size. This value ensures localized exploration rather than global disruption, enabling fine-grained adjustments of charging schedules. From a physical perspective, this corresponds to minor operational variations in EV charging and discharging rates that are feasible within practical energy management systems. From an economic standpoint, it prevents abrupt schedule changes that could negate cost benefits or violate operational constraints. The variable η is a random variable drawn from a standard normal distribution, $\eta \sim N(0, 1)$, with zero mean and unit variance. This term introduces unbiased stochastic perturbations that enhance local exploitation while preserving numerical stability.

(2) Position Update with RWE

Each particle's position is updated by adding the random walk step to the current position. The equation for the position update is defined as:

$$X_{RWE} = X_i + \Delta X \quad (27)$$

This update mechanism implements a Gaussian local search, which is widely used in metaheuristic optimization to intensify exploitation near high-quality solutions. The use of a zero-mean normal distribution biases the search toward local neighborhoods, improving solution precision and convergence speed without introducing directional bias.

(3) Evaluation and Acceptance of New Positions

The updated position X_{RWE} is then evaluated. If the evaluation of the function with X_{RWE} provides a lesser cost than with X_i , then $X_i = X_{RWE}$ and $f(X_i) = f(X_{RWE})$; otherwise, the particle retains its previous position.

b. Periodic Random Parameter Tuning Method

1. PRPT: This method introduces lightweight and periodic adjustments to the key parameters of the IPLO, helping to maintain a balance between exploration and exploitation throughout iterations. PRPT accomplishes this by slightly perturbing parameters within controlled ranges to ensure flexibility and adaptability without substantial computational overhead.

The selected parameters include the following:

- Intensity Weights (w_1 , w_2) manage exploration and exploitation trade-offs, adapting over iterations.
 - The damping factor (α) adjusts the movement magnitude of candidate solutions, encouraging exploration in early phases and fine-tuning later.
2. Perturbation Equations

Each parameter is periodically updated based on simple perturbations as follows:

- (1) Intensity Weights Update (w_{1new} and w_{2new})

$$w_{1new} = w_1 + \delta_{w1} \quad (28)$$

$$w_{2new} = w_2 + \delta_{w2} \quad (29)$$

where δ_{w1} and δ_{w2} are small, independent perturbation steps with ranges of $\delta_{w1} \in [-0.03, 0.08]$ and $\delta_{w2} \in [-0.05, 0.02]$. Since w_1 primarily governs local exploitation while w_2 controls global exploration, asymmetric perturbation ranges are adopted. Larger positive perturbations are allowed for w_1 to intensify local refinement, whereas perturbations of w_2 are more restricted to limit unnecessary global exploration once promising regions have been identified. These ranges are intentionally narrow to preserve algorithm stability while enabling gradual adaptation across different search phases. From an optimization perspective, this prevents premature convergence caused by static parameter settings, while economically it maintains a smooth evolution of charging decisions across time.

- (2) Damping Factor Update α

$$\alpha_{new} = \alpha \times (1 + \delta_{\alpha}) \quad (30)$$

The damping factor regulates the persistence and magnitude of particle motion, effectively controlling the aggressiveness of the search process. In the context of EV charging optimization, overly aggressive motion corresponds to abrupt and unrealistic SOC changes that may increase battery stress and lead to unstable charging schedules. Therefore, the damping factor is maintained within a narrow perturbation range, δ_w , scaled

between $[-0.1, 0.1]$ to ensure smooth and physically feasible solution trajectories. Furthermore, linking damping adaptation to battery degradation enables a gradual search momentum as health deteriorates, promoting conservative refinement near optimal schedules and preserving economically smooth charging behavior under uncertainty.

3. Results

This section of the paper discusses the results of the study. The main points addressed are the following:

1. Analysis of the test case: the test case includes the evaluation of two ΔSOC_{min} values, namely 20% and 10%. This analysis helps to investigate the impact of different limiting cycle levels on the optimization process [10].
2. Dealing with uncertainty: The paper addresses how the presented approach deals with uncertainties related to battery performance degradation and changes in electricity prices. This aspect highlights the algorithm's ability to adapt and optimize under uncertain conditions [10].
3. Comparison with state-of-the-art algorithms: The performance of the IPLO is compared with other metaheuristic algorithms [29], including the PLO [26], Dung Beetle Optimizer (DBO) [30], Moth Flame Optimizer (MFO) [31], Hippopotamus Optimizer (HO) [32], Sea Horse (SH) [33], Whale optimization algorithm (WOA) [34], and Coati optimization algorithm (COA) [35]; in addition, a non-optimized approach is included as part of the comparison process where a customer performs the charging/discharging operations based on their own choice to highlight the pivotal role of optimization in the process. Several trial runs are performed with a population size P of 100 and iteration number T of 3000 for each algorithm to ensure a fair comparison within certain parameter values.
4. Control parameters and robustness: Table 2 shows the additional control parameters of the selected algorithms. To ensure reliability and robustness, each algorithm is subjected to 20 test runs so that its performance can be comprehensively evaluated. Overall, this section provides a thorough analysis of the test case, a comparison with other algorithms, and an evaluation of the IPLO algorithm's performance in managing uncertainties and optimizing battery usage in response to changing conditions.

Table 2. Control parameters of all algorithms.

Algorithm	Parameters	Values
IPLO, PLO	$\alpha, W1, W2, m$	$[1, 1.5], [0, 0.8], [0, 1], 100$
DBO	α, b, k	$\{-1, 1\}, (0, 1)$ $(0, 0.2]$
MFO	b, r	1, decrease linearly from -1 to -2
HO	$r1, r2, I1, I2$	$[0, 1], [0, 1]$ $[1, 2], [1, 2]$
SH	$\alpha, r1, r2, r3$	Decrease linearly from 1 to 0 $[0, 1], [0, 1], [0, 1]$
WOA	r, l, α	$[0, 1], [-1, 1]$, decrease linearly from 2 to 0
COA	r, I	$[0, 1], \{1, 2\}$

The analysis includes three targeted test case studies, each addressing distinct aspects of the optimization problem:

1. Test Case 1: This case evaluates the optimized scheduling results produced by various algorithms. The objective is to identify the most effective algorithm for the problem, focusing on battery lifespan preservation by implementing two limiting cycle values.
2. Test Case 2: This study examines battery performance degradation factors, such as frequent cycles of charging and discharging and extreme battery levels (either fully charged or discharged). It analyzes the impact of these factors on the cost function, with the necessity of compensation strategies to maintain profit goals during battery degradation conditions.
3. Test Case 3: This study assesses the influence of fluctuating electricity prices during operation, incorporating real-time pricing to minimize costs and strengthen the model's robustness.

Together, these test cases target key cost function factors to optimize EV charging schedules by maximizing profit while considering battery health and dynamic pricing. Comparative analysis with existing algorithms highlights the effectiveness of the presented approach and offers valuable insights to guide future research in EV charging optimization.

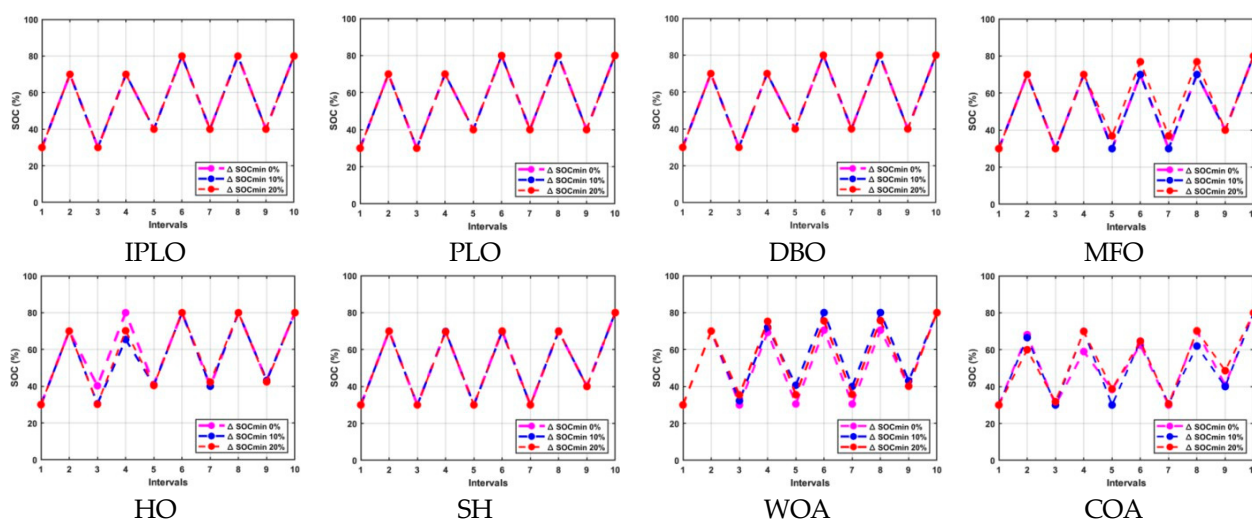
3.1. Test Case Study 1

This paper introduces a scheduling operation for a 23 kWh, 384 V, and 60 Ah battery [10] over a nine-period time frame to ensure more robust results with the minimum V2G and G2V cycles. The main objective is to minimize the total charging cost of electric vehicles, EVs, by optimizing a cost function with five constraints. The initial SOC of the battery is 30%, and the target is to reach 80% SOC; the maximum chargeable and dischargeable energy amount is set to 40% during any interval. The battery has a charging C-rating of 0.9 C and a discharging C-rating of 1 C. The price of electricity is $R_k = \{7, 12, 8, 11, 9, 13, 7, 12, 10\}$; it varies over the nine intervals, and the customer receives benefits for supplying energy back to the grid and paying for charging their EV battery [10].

The performance and efficiency of the IPLO were evaluated against various metaheuristic algorithms and a non-optimized version, focusing on cost and efficiency. As shown in Table 3, IPLO achieved the lowest cost at Rs 1050.23, significantly outperforming the other methods. DBO and PLO followed, with Rs 2816 and Rs 2817.94, while MFO and SH resulted in costs of Rs 5120 and Rs 5361, respectively. The WOA followed, with Rs 5418, and the HO exhibited relatively higher costs. Meanwhile, both the COA and the non-optimized version recorded the highest costs in this category. These results underscore IPLO's ability to efficiently schedule charging and discharging operations under normal, unrestricted conditions. In Table 4, under limited cycle (l.c.) conditions of 20% and 10%, IPLO maintained its top performance, with a consistent cost of Rs 1050.23. DBO and PLO also performed reliably, achieving in sequence Rs 2816 and Rs 2819.9 for both limited cycles. MFO followed, with Rs 3534.54 under 20% l.c. and Rs 5120 without limitation. WOA ranked next, with Rs 4174.12 under 10% l.c., while HO recorded Rs 6154.5 (20% l.c.) and Rs 6145.45 (10% l.c.). SH produced costs of Rs 5538 and Rs 5554 for 20% and 10% l.c., respectively. COA showed high costs of Rs 24,113.35 (20% l.c.) and Rs 20,772.99 (10% l.c.), while the non-optimized version was the least efficient, with Rs 38,912 (20% l.c.) and Rs 43,520 (10% l.c.). Figure 5 illustrates these variations across discharge and charge cycles with and without limitations. Overall, in this initial case study, IPLO demonstrated an outstanding cost-effectiveness and adaptability across all conditions, setting a new benchmark with Rs 1050.23, remarkably exceeding the PLO performances, while DBO also proved to be highly effective. MFO and WOA followed as reliable alternatives under limited-cycle scenarios.

Table 3. Scheduling without limiting cycles for nine intervals.

Algorithm	k	0	1	2	3	4	5	6	7	8	9	J_k
IPLO	SOC	30%	69.99%	30%	69.99%	40%	79.99%	39.99%	79.99%	40.03%	80%	
	Cost		71,677.97	−110,588.8	81,916.93	−76,027.58	92,137.33	−119,779.75	71,678.22	−110,493.05	100,529.03	1050.23
PLO	SOC	30%	70%	30%	69.99%	40%	80%	40%	80%	40%	80%	
	Cost		71,680	−110,592	81,917.952	−76,029.46	92,160	−119,808	71,680	−110,580.94	102,389.76	2817.94
DBO	SOC	30%	70%	30%	70%	40%	80%	40%	80%	40%	80%	
	Cost		71,680	−110,592	81,920	−76,032	92,160	−119,808	71,680	−110,592	102,400	2816
MFO	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
	Cost		71,680	−110,592	81,920	−101,380	92,160	−119,810	71,680	−82,940	102,400	5120
HO	SOC	30%	70%	40.24%	80%	40.97%	80%	40%	80%	43.16%	80%	
	Cost		71,679.11	−82,263.20	81,416.71	−98,918	89,925.46	−119,805.66	71,678.60	−101,831.26	94,288.21	6169.96
SH	SOC	30%	70%	30%	69.8%	30%	70%	30%	69.9%	40.1%	80%	
	Cost		71,612	−110,487	81,535	−100,900	92,072	−119,694	71,546	−82,557	102,234	5361
WOA	SOC	30%	70%	30%	69.27%	30.53%	70.49%	30.50%	70.50%	40.21%	80%	
	Cost		716,780	−110,590.4	80,422.92	−98,165.52	92,062.32	−119,762.61	71,679.99	−83,763.51	101,855.56	5418.7
COA	SOC	30%	68.14%	30%	58.97%	38.40%	62.43%	30%	69.99%	40.79%	80%	
	Cost		68,357.24	−105,465.4	59,329.80	−52,107.8	55,354.29	−97,148.48	71,672.36	−80,728.76	100,359.77	19,622.97
Non-Optimized	SOC	30%	70%	34%	75%	36%	57%	31%	72%	41%	80%	
	Cost		71,680	−99,532.8	83,968	−98,841.6	48,384	−77,875.2	73,472	−85,708.8	99,840	15,385.6

**Figure 5.** Discharging and charging operations with 0% 10% and 20% limiting cycles for nine intervals.**Table 4.** Scheduling with limiting cycles for nine intervals.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
IPLO	20%	SOC	30%	69.99%	30%	69.99%	40%	79.99%	39.99%	79.99%	40.03%	80%	
		Cost		71,677.97	−110,588.87	81,916.93	−76,027.58	92,137.33	−119,779.75	71,678.22	−110,493.05	100,529.03	1050.23
	10%	SOC	30%	69.99%	30%	69.99%	40%	79.99%	39.99%	79.99%	40.03%	80%	
		Cost		71,677.97	−110,588.87	81,916.93	−76,027.58	92,137.33	−119,779.75	71,678.22	−110,493.05	100,529.03	1050.23
PLO	20%	SOC	30%	69.99%	30%	70%	40%	80%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,913.85	−76,016.79	92,153.08	−119,805.00	71,678.20	−110,583.70	102,392.32	2819.96
	10%	SOC	30%	69.99%	30%	70%	40%	80%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,913.856	−76,016.79	92,153.088	−119,805.00	71,678.208	−110,583.70	102,392.32	2819.96
DBO	20%	SOC	30%	70%	30%	70%	40%	80%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,920	−76,032	92,160	−119,808	71,680	−110,592	102,400	2816
	10%	SOC	30%	70%	30%	70%	40%	80%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,920	−76,032	92,160	−119,808	71,680	−110,592	102,400	2816
MFO	20%	SOC	30%	70%	30%	70%	36.88%	76.88%	36.88%	76.88%	40%	80%	
		Cost		71,680	−110,592	81,920	−83,935.96	92,160	−119,808	71,680	−101,969.48	102,400	3534.54

HO	10%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	5120
		Cost		71,680	−110,592	81,920	−101,380	92,160	−119,810	71,680	−82,940	102,400	
	20%	SOC	30%	70%	30.18%	70.18%	40.38%	80%	42.25%	80%	42.40%	80%	
		Cost		71,679.73	−110,073.82	81,919.36	−75,529.2	91,272.15	−113,054.28	67,639.28	−103,933.49	96,234.77	
SH	10%	SOC	30%	69.99%	30.47%	65.42%	40.55%	80%	40%	80%	43.07%	80%	6145.45
		Cost		71,676.32	−109,276.92	71,569.88	−63,012.93	90,877.20	−119,804.72	71,678.03	−102,079	94,517.59	
	20%	SOC	30%	69.9%	30%	69.6%	30%	70%	30%	69.9%	40%	80%	
		Cost		71,440	−110,222	81,158	−100,433	92,093	−119,721	71,528	−82,651	102,346	
WOA	10%	SOC	30%	70%	30%	69.8%	30%	69.6%	30%	70%	40%	80%	5554
		Cost		71,598	−110,465	81,539	−100,905	91,294	−118,682	71,610	−82,828	102,392	
	20%	SOC	30%	70%	35.45%	75.27%	35.51%	75.51%	35.71%	75.71%	40.12%	80%	
		Cost		71,680	−95,509.87	81,544.11	−100,764.71	92,160	−119,212.83	71,679.31	−98,385.59	102,077.53	
COA	10%	SOC	30%	70%	32.25%	72.05%	40.67%	80%	40%	80%	43.04%	80%	4174.12
		Cost		71,679.85	−103,661.60	81,920	−80,665.13	90,596.80	−119,807.99	71,679.99	−102,165.24	94,597.45	
	20%	SOC	30%	59.96%	31.80%	69.87%	38.74%	64.56%	30.49%	70.26%	48.61%	80%	
		Cost		53,693.13	−77,850.51	77,970.70	−78,889.07	59,473.66	−102,033.60	71,268.74	−59,878.65	80,358.94	
Non-Optimized	10%	SOC	30%	66.57%	30%	69.99%	30%	64.65%	30.68%	61.98%	40%	80%	20,772.99
		Cost		65,538.81	−101,117.03	81,909.15	−101,362.57	79,839.19	−101,755.29	56,090.73	−60,756.86	102,386.86	
	20%	SOC	30%	70%	40%	60%	30%	70%	40%	60%	40%	80%	
		Cost		71,680	−82,944	40,960	−76,032	92,160	−89,856	35,840	−55,296	102,400	
	10%	SOC	30%	70%	40%	60%	50%	70%	40%	60%	40%	80%	43,520
		Cost		71,680	−82,944	40,960	−25,344	46,080	−89,856	35,840	−55,296	102,400	

3.2. Test Case Study 2

Battery Degradation Uncertainty

Battery performance degradation refers to the gradual decline in capacity over time, primarily influenced by factors such as frequent charging, a deep low state of charge, full-capacity charging, and excessive fast charging [36]. These factors introduce uncertainties in maintaining the desired SOC and ensuring cost stability. This study assesses the IPLO algorithm's ability to manage such uncertainties by adapting its optimization strategy based on the real-time detection of battery performance changes. Degradation detection operates at two levels. At the charging infrastructure level, advanced battery health monitoring techniques such as impedance spectroscopy [37], incremental capacitance analysis [38], and the sensor-based tracking of temperature and voltage enable the prediction and detection of degradation by comparing collected data (e.g., cycles, temperature, voltage) against healthy benchmarks. Techniques like Coulomb counting [39], voltage-based SOC estimation, and Incremental Capacity Analysis (ICA) further contribute precise insights by tracking capacity changes and analyzing voltage capacity derivatives. Recent advances in battery health modeling demonstrate that battery aging is a complex, data-dependent phenomenon influenced by operating conditions, cycling behavior, and cell-to-cell variability. Deep learning frameworks such as BatLiNet enable an accurate lifetime prediction across diverse aging regimes [40], while platforms like BatteryML highlight the growing need for unified data-driven degradation modeling [41]. While such approaches are highly effective for lifetime prediction and offline analysis, they rely on large datasets and chemistry-specific parameters that limit their suitability for real-time EV charging optimization. In contrast, this work adopts a control-oriented degradation-aware formulation that leverages lightweight health indicators to enable adaptive charging decisions under uncertainty, prioritizing responsiveness and computational efficiency. At the experimental level, the presented degradation-aware cost formulation introduces a capacity-based health indicator and an adaptive operational mechanism that dynamically adjusts

charging and discharging actions in response to observed battery degradation. By dampening SOC transitions proportionally to health loss, the approach reduces deep cycling and high-stress operating conditions while preserving economic responsiveness. This control-oriented formulation is computationally efficient, physically consistent, and well suited for real-time EV charging optimization under uncertainty.

To evaluate this, the algorithm was tested over three key intervals ($k = 1, 4$, and 7) and under two degradation constraints: 20% and 10% cycle limitations. Tables 5–7 show the scheduling results of this case study. In the first interval ($k = 1$), IPLO achieved a highly favorable outcome of Rs −1107.59 for both cycle limits, entering profitability; on the other hand, PLO achieved Rs 668.7 for both cycle limits. This result illustrates IPLO’s strength in early-stage degradation management, applying compensatory adjustments to minimize costs while maintaining system stability. Comparatively, DBO recorded Rs 665.6 and Rs 780.09, while MFO achieved Rs 669.23 and Rs 665.6 for the 20% and 10% limits, respectively. WOA followed with a higher Rs 1954.45 under 10% l.c., while HO and SH ranged from Rs 3200 to Rs 4200, proving less reliable in this phase. COA and the non-optimized version resulted in significantly higher costs, reinforcing their inefficiency. In the fourth interval ($k = 4$), IPLO maintained its effectiveness, with a cost of Rs 3329.54 under both cycle limitations, while the PLO jumped to Rs 5106.32 and 5101.32 for 20% and 10%, respectively, showing a remarkable difference. This mid-stage result outperformed DBO, which scored Rs 5495.21 (20%) and Rs 5899.06 (10%), while MFO followed closely with slightly higher costs. HO produced Rs 7528 (20%) and Rs 6477 (10%), and SH and WOA ranged from Rs 8200 to Rs 9200, confirming their comparatively weaker performance. Again, COA and the non-optimized version generated the highest costs. This stage posed greater uncertainties due to mid-cycle degradation, which IPLO successfully mitigated by recalibrating its schedule through its compensatory mechanism. In the seventh interval ($k = 7$), IPLO again demonstrated profitability, with Rs −1118.2419 under both limitations, showing a strong adaptability to late-stage degradation, again outperforming the PLO, which achieved Rs 669.13 and Rs 668.9 for 20% and 10%, respectively. DBO ranked second, with Rs 818.1 for 20% l.c., followed by MFO, with Rs 856.91 and Rs 985.65 for 20% and 10%, respectively. In contrast, HO, SH, and WOA incurred higher operational expenses between Rs 3000 and 7000. COA yielded a cost of Rs 88,142.05 for 20%, and the non-optimized version reached Rs 30,156.8 for 10%, demonstrating their significant inefficiency. These findings confirm the IPLO algorithm’s robustness and flexibility in managing degradation while ensuring profitability and system resilience. By detecting and improving performance drops and applying a compensation factor, IPLO dynamically adapts its schedule, outperforming PLO and other contemporary algorithms across multiple intervals and degradation conditions. Figures 6–8 illustrates the resulting charge/discharge operations for this case study.

Table 5. Scheduling with limiting cycles for nine intervals with battery degradation uncertainty at $k = 1$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
IPLO	20%	SOC	30%	69.99%	30%	69.99%	40.02%	79.99%	40%	80%	40%	80%	
		Cost		69,522.35	−110,580.47	81,916.39	−75,974.73	92,111.25	−119,805.96	71,679.36	−110,591.27	100,615.49	−1107.59
	10%	SOC	30%	69.99%	30%	69.99%	40.02%	79.99%	40%	80%	40%	80%	
		Cost		69,522.35	−110,580.47	81,916.39	−75,974.73	92,111.25	−119,805.96	71,679.36	−110,591.27	100,615.49	−1107.59
PLO	20%	SOC	30%	69.99%	30%	69.99%	40.02%	79.99%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,913.856	−76,019.32	92,155.39	−119,808	71,680	−110,586.47	102,394.88	668.7031
	10%	SOC	30%	69.99%	30%	69.99%	40.02%	79.99%	40%	80%	40%	80%	
		Cost		71,680	−110,592	81,913.85	−76,019.32	92,155.39	−119,808	71,680	−110,586.47	102,394.88	668.7031

DBO	20%	SOC	30%	70%	30%	70%	40%	80%	40%	80%	40%	80%	665.6
		Cost		69,529.6	−110,592	81,920	−76,032	92,160	−119,808	71,680	−110,592	102,400	
	10%	SOC	30%	70%	30%	70%	39.5%	79.5%	39.5%	79.5%	40%	80%	780.09
		Cost		69,529.6	−110,592	81,920	−77,291.45	92,160	−119,808	71,680	−109,218.05	102,400	
MFO	20%	SOC	30%	70%	30.01%	70.01%	39.99%	79.99%	39.99%	79.99%	40%	80%	669.23
		Cost		69,529.6	−110,560.78	81,919.99	−76,072.02	92,159.99	−119,808	71,679.99	−110,579.54	102,400	
	10%	SOC	30%	70%	30%	70%	40%	80%	40%	80%	40%	80%	665.6
		Cost		69,529.6	−110,592	81,920	−76,032	92,160	−119,808	71,680	−110,592	102,400	
HO	20%	SOC	30%	70%	35.92%	75.92%	43.21%	80%	40%	80%	43.91%	80%	3594.94
		Cost		69,529.55	−94,218.82	81,919.87	−82,884.17	84,745.13	−119,804.7	71,678.03	−99,494.32	92,124.37	
	10%	SOC	30%	70%	30.01%	70.01%	40.60%	80%	40%	80%	56.32%	80%	4153.15
		Cost		69,529.58	−110,556.46	81,919.74	74,530.62	90,765.79	−119,805.40	71,678.45	−65,447.16	60,599.23	
SH	20%	SOC	30%	70%	30%	70%	30%	69.7%	30%	70%	40.1%	80%	3249
		Cost		69,451	−110,468	81,900	−101,351	91,453	−118,888	71,680	−82,723	102,195	
	10%	SOC	30%	69.93%	30%	69.8%	30%	69.9%	30%	69.9%	40%	80%	3241
		Cost		69,408	−110,398	81,578	−100,953	91,992	−119,589	71,572	−82,648	102,280	
WOA	20%	SOC	30%	69.99%	35.46%	75.46%	39.96%	79.96%	39.98%	79.96%	40%	80%	1954.45
		Cost		69,528.09	−95,472.34	81,920	−89,968.7	92,159.94	−119,760.06	71,651.32	−110,482.32	102,378.55	
	10%	SOC	30%	70%	30%	65.16%	33.03%	73.01%	35.23%	75.23%	42.01%	80%	5555.44
		Cost		69,529.6	−110,592	72,019.4	−81,445.82	92,125.17	−113,146.62	71,679.94	−91,850.95	97,236.73	
COA	20%	SOC	30%	56.33%	31.29%	68.19%	44.83%	77.92%	55.79%	35.79%	58.37%	80%	84,819.34
		Cost		45,774.92	−69,236.20	75,578.39	−59,210.15	76,240.82	−66,281.27	−35,843.1	62,440.69	55,355.23	
	10%	SOC	30%	66.63%	37.96%	74.35%	38.75%	65.89%	40.74%	79.42%	64.9%	80%	23,716.61
		Cost		63,687.62	−79,282.94	74,519.18	−90,215.18	62,531.96	−75,340.48	69,330.27	−40,170.41	102,400	
Non-Optimized	20%	SOC	30%	70%	40%	80%	50%	70%	30%	60%	40%	80%	19,609.6
		Cost		69,529.6	−82,944	81,920	−76,032	46,080	−119,808	53,760	−55,296	102,400	
	10%	SOC	30%	70%	30%	60%	50%	70%	40%	70%	40%	80%	24,473.6
		Cost		69,529.6	−110,592	61,440	−25,344	46,080	−89,856	53,760	−82,944	102,400	

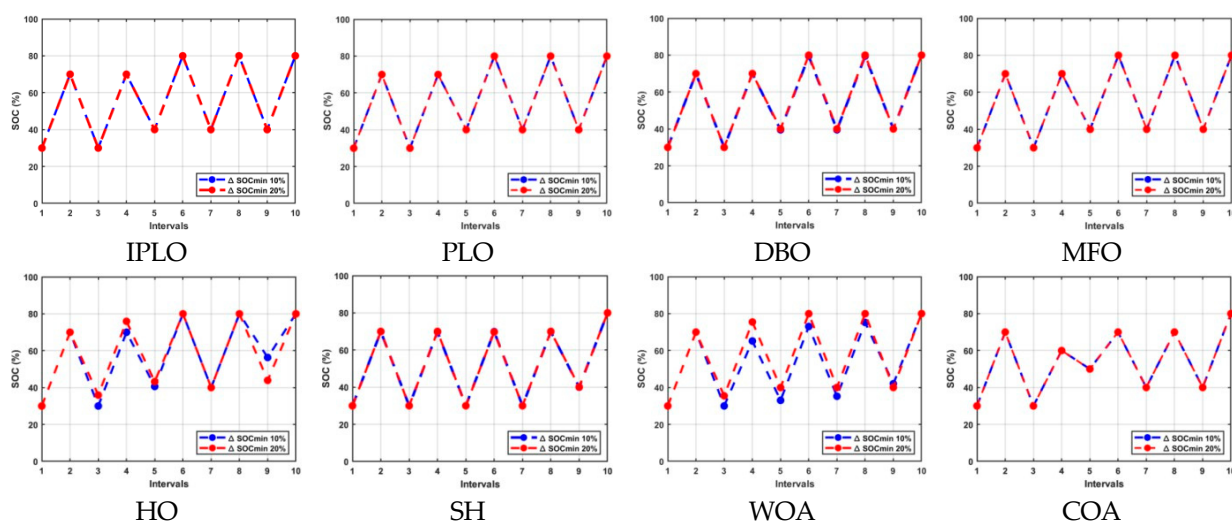


Figure 6. Discharging and charging operations with 10% and 20% limiting cycles for nine intervals with uncertainty at $k = 1$.

Table 6. Scheduling with limiting cycles for nine intervals with battery degradation uncertainty at $k = 4$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
IPLO	20%	SOC	30%	69%	30%	69%	40.01%	79.99%	39.99%	79.99%	40%	80%	
		Cost		71,665.16	$-110,564.51$	81,914.76	-73,717.46	92,118.37	$-119,793.06$	71,679.96	$-110,577.36$	100,603.69	3329.54
	10%	SOC	30%	69%	30%	69%	40.01%	79.99%	39.99%	79.99%	40%	80%	
		Cost		71,665.16	$-110,564.51$	81,914.76	-73,717.46	92,118.37	$-119,793.06$	71,679.96	$-110,577.36$	100,603.69	3329.54
PLO	20%	SOC	30%	69.99%	30%	69.98%	40.01%	79.99%	39.99%	79.99%	40%	80%	
		Cost		71,676.416	$-110,586.47$	81,893.3	-73,716.62	92,153.08	$-119,805.00$	71,678.20	$-110,583.70$	102,397.44	5106.32
	10%	SOC	30%	69%	30%	69%	40.01%	80%	40%	79.99%	40%	80%	
		Cost		71,678.20	$-110,589.23$	81,915.9	-73,723.99	92,139.26	-119,808	71,678.20	$-110,586.47$	102,397.44	5101.32
DBO	20%	SOC	30%	70%	30%	70%	38.7%	78.7%	38.7%	78.7%	40%	80%	
		Cost		71,680	-110,592	81,919.99	-76,946.03	92,160	$-119,807.98$	71,680	$-106,998.76$	102,400	5495.21
	10%	SOC	30%	70%	30%	70%	37.38%	77.38%	37.38%	77.38%	40%	80%	
		Cost		71,680	-110,592	81,920	-80,185.98	92,160	-119,808	71,680	$-103,354.96$	102,400	5899.06
MFO	20%	SOC	30%	70%	30%	70%	35.81%	75.81%	35.81%	75.81%	40%	80%	
		Cost		71,679.99	-110,592	81,920	$-84,040.35283$	92,160	-119,808	71,680	-99,020.13	102,400	6380
	10%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,680	-110,592	81,920	-98,334.72	92,160	-119,808	71,680	-82,944	102,400	8161
HO	20%	SOC	30%	70%	32.75%	72.75%	44.83%	80%	40.22%	80%	42.78%	80%	
		Cost		71,679.56	$-102,962.28$	81,919.47	-68,659.27	81,031.03	$-119,136.98$	71,278.53	$-102,884.43$	95,263.36	7528.98
	10%	SOC	30%	70%	30.06%	70.06%	45.55%	80%	40.11%	80%	41.79%	80%	
		Cost		71,679.89	$-110,402.81$	81,919.64	-60,264.38	79,363.09	$-119,472.11$	71,479.057	$-105,632.50$	97,807.88	6477.73
SH	20%	SOC	30%	69.72%	30%	69.99%	30%	69.94%	30%	69.8%	40%	80%	
		Cost		71,626	-110,509	81,912	-98,325	92,033	-119,643	71,434	-82,562	102,397	8364
	10%	SOC	30%	69.9%	30%	69.71%	30%	69.91%	30%	69.75%	40%	80%	
		Cost		71,662	-110,564	81,342	-97,641	91,960	-119,547	71,239	-82,264	102,400	8586
WOA	20%	SOC	30%	70%	30%	69.66%	30%	69.36%	30%	69.44%	40%	80%	
		Cost		71,680	$-110,591.99$	81,242.74	-97,521.75	90,693.4	$-117,901.42$	70,692.07	-81,413.39	102,394.09	9273.74
	10%	SOC	30%	70%	30.15%	70.14%	30.17%	70.15%	30.15%	70.15%	40.14%	80%	
		Cost		71,679.94	$-110,158.20$	81,894.72	-98,264.64	92,117.62	-119,808	71,674.27	-82,974.43	102,040.5	8201.77
COA	20%	SOC	30%	69.99%	30%	69.30%	38.17%	61.73%	30%	62.12%	40.27%	80%	
		Cost		71,662	$-110,564.23$	80,501.92	-76,527.91	54,277.45	-95,055.5	57,566.64	-60,406.69	101,694.06	23,147.74
	10%	SOC	30%	55.65%	30%	69.94%	52.28%	80%	56.52%	80%	40.03%	80%	
		Cost		45,970.39	-70,925.74	81,811.47	-43,422.31	63,857.79	-70,327.64	42,076.36	-110,501.6	102,316.29	40,855.02
Non-Optimized	20%	SOC	30%	70%	30%	60%	40%	70%	30%	50%	30%	80%	
		Cost		71,680	-110,592	61,440	-49,167.36	69,120	-119,808	35,840	-55,296	128,000	31,216.64
	10%	SOC	30%	70%	30%	60%	50%	70%	30%	40%	30%	80%	
		Cost		71,680	-110,592	61,440	-24,583.68	46,080	-119,808	17,920	-27,648	128,000	42,488.32

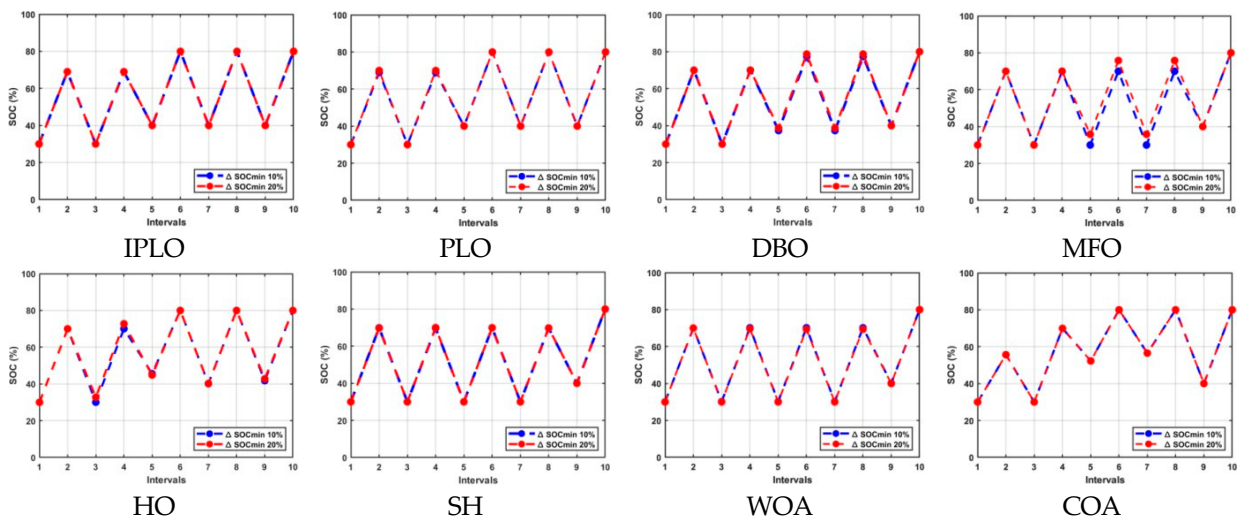


Figure 7. Discharging and charging operations with 10% and 20% limiting cycles for nine intervals with uncertainty at $k = 4$.

Table 7. Scheduling with limiting cycles for nine intervals under battery degradation uncertainty at $k = 7$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
IPLO	20%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.92%	39.93%	79.93%	39.94%	80%	
		Cost		71,669.88	-110,519.2 7	81,908.63	-76,197.09	92,106.01	-119,772.2 7	69,523.72	-110,575.7 6	100,764.69	-1091.473
	10%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.92%	39.93%	79.93%	39.94%	80%	
		Cost		71,669.88	-110,519.2 7	81,908.63	-76,197.09	92,106.01	-119,772.2 7	69,523.72	-110,575.7 6	100,764.69	-1091.473
PLO	20%	SOC	30%	69.99%	30%	69.99%	40%	80%	40%	80%	40%	80%	
		Cost		71,678.20	-110,589.2 3	81,911.80	-76,021.86	92,160	-119,808	69,529.6	-110,583.7 0	102,392.32	669.13
	10%	SOC	30%	69.99%	30%	69.99%	40%	80%	40%	80%	40%	80%	
		Cost		71,678.20	-110,589.2 3	81,917.95	-76,029.46	92,160	-119,805.0 8	69,527.86	-110,586.4 7	102,394.88	668.90
DBO	20%	SOC	30%	70%	30%	70%	39.33%	79.33%	39.33%	79.33%	40%	80%	
		Cost		71,680	-110,592	81,920	-77,709.6	92,160	-119,808	69,529.6	-108,761.8 9	102,400	818.1
	10%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,680	-110,592	81,920	-101,376	92,160	-119,808	69,529.6	-82,944	102,400	2969.6
MFO	20%	SOC	30%	70%	30.83%	70.83%	40%	80%	40%	80%	40%	80%	
		Cost		69,529.6	-108,296.2 6	81,920	-78,136.42	92,160	-119,808	71,680	-110,592	102,400	856.91
	10%	SOC	30%	70%	31.38%	71.38%	40%	80%	40%	80%	40%	80%	
		Cost		69,529.6	-106,751.3 6	81,920	-79,552.58	92,160	-119,808	71,680	-110,592	102,400	985.65
HO	20%	SOC	30%	70%	30.93%	67.81%	37.74%	77.74%	37.74%	77.74%	48.76%	80%	
		Cost		71,679.83	-108,004.5 7	75,523.47	-76,204.92	92,157.71	-119,803.0 8	69,528.82	-80,114.94	79,954.77	4717.10
	10%	SOC	30%	70%	40%	80%	43.16%	80%	40%	80%	57.19%	80%	
		Cost		71,679.86	-82,924.11	81,905.42	-93,355.43	84,868.57	-119,807.7 4	69,529.45	-63,050	58,379.62	7225.65
SH	20%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,583	-110,443	80,851	-100,053	91,139	-118,481	69,046	-82,175	102,400	3868
	10%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,352	-110,086	81,783	-101,206	92,139	-119,781	69,384	-82,692	102,382	3274
WOA	20%	SOC	30%	70%	34.84%	74.8%	36.38%	74.79%	34.79%	74.45%	40.2%	80%	
		Cost		71,680	-97,211.21	81,845.84	-97,374.42	88,499.02	-119,808	68,937.60	-94,703.28	101,888.92	3754.47
	10%	SOC	30%	70%	30%	66.17%	30.31%	70.31%	30.52%	70.52%	40.52%	80%	
		Cost		71,680	-110,571.4 0	74,062.51	-90,865.82	92,159.95	-119,179.9 1	69,529.52	-82,944.58	101,050.32	4920.59
COA	20%	SOC	30%	56.38%	76.38%	47.09%	78.92%	58.69%	38.68%	74.93%	52.70%	80%	
		Cost		47,272.23	55,296.67	-59,968.74	80,662.14	-46,621.64	-59,927.75	63,015.57	-61,475.58	69,889.15	88,142.05

Non-Optimized	10%	SOC	30%	69.46%	40.01%	80%	69.99%	80%	48.76%	80%	40.30%	80%	
		Cost		70,723.69	-81,429.20	81,890.85	-25,359.41	23,054.1	-93,566.32	54,300.46	-109,754.91	101,624.91	21,484.07
	20%	SOC	30%	70%	30%	80%	50%	70%	30%	60%	40%	80%	
		Cost		71,680	-110,592	102,400	-76,032	46,080	-119,808	52,147.2	-55,296	102,400	12,979.2
	10%	SOC	30%	70%	30%	80%	50%	60%	30%	50%	40%	80%	
		Cost		71,680	-110,592	102,400	-76,032	23,040	-89,856	34,764.8	-27,648	102,400	30,156.8

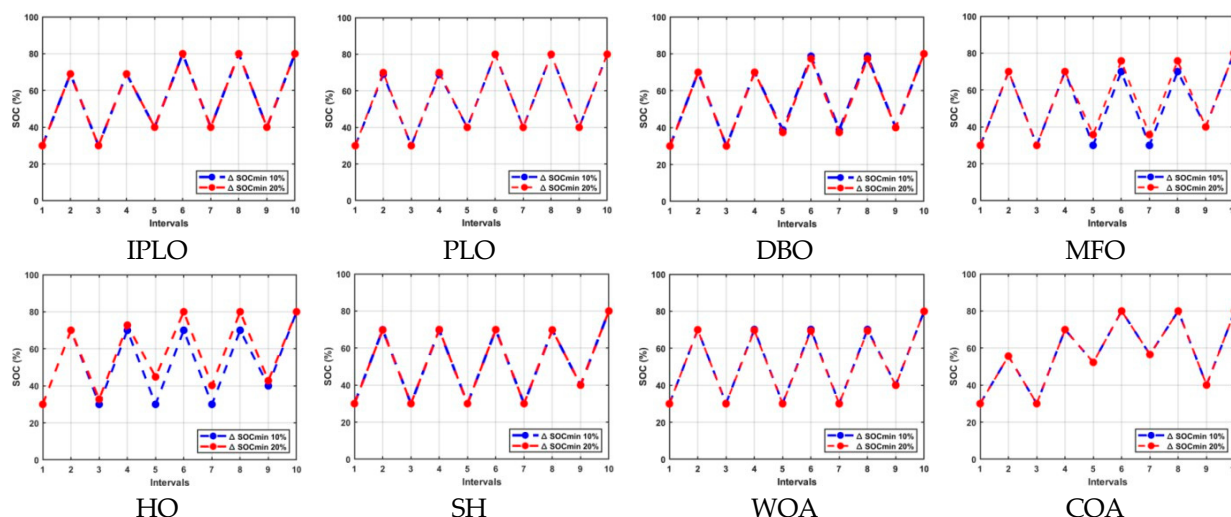


Figure 8. Discharging and charging operations with 10% and 20% limiting cycles for nine intervals with uncertainty at $k = 7$.

3.3. Test Case Study 3

Electricity Price Fluctuation Uncertainty

This case study explores the adaptability and robustness of the IPLO in handling uncertainties in electricity pricing, a major factor in minimizing EV charging costs. At the infrastructure level, real-time monitoring and smart grid communication enable charging systems to adjust schedules dynamically based on live electricity rate updates [42]. Simultaneously, at the experimental level, a metaheuristic algorithm plays a pivotal role [43], IPLO receives predefined pricing arrays and continuously compares them with real-time data as a scenario-based uncertainty modeling approach. When fluctuations are detected, it recalibrates the charging and discharging strategy using a correction factor linked to the updated prices, which also influences its exploration and exploitation dynamics. This adaptive mechanism allows IPLO to remain cost-efficient and responsive, with decisions executed directly through EV infrastructure controls. The approach not only reduces costs but also supports grid stability and sustainability. Tables 8–10 show the scheduling results with electricity pricing uncertainty at $k = 2$, $k = 3$ and $k = 5$, and $k = 6$, respectively.

IPLO's performance was evaluated across different uncertain intervals ($k = 2, 3$ and $5, 6$) and consistently demonstrated the lowest operational costs compared to other algorithms. In the interval $k = 2$, IPLO achieved Rs 10,268.68 for both 10% and 20% limiting cycles, outperforming all competitors. PLO followed, with Rs 12,034.27 and Rs 12,034.8 for 20% and 10% limiting cycles, respectively. DBO and MFO followed, but with noticeably higher costs, especially for the 20% limiting cycle, while COA showed instability under stricter conditions. WOA showed a moderate adaptability, whereas HO, SH, and the non-optimized approach registered the highest costs. In intervals $k = 3$ and $k = 5$, with uncertainty present in two-time intervals, IPLO again led, with Rs 19,748.52 for both limiting cycles, showing a significant difference from the original PLO, which followed with Rs 23,049.62 and Rs 22,788.01 for the 20% and 10% limiting cycles. DBO and MFO were

the next best performers but still trailed IPLO. COA and SH showed a reduced reliability, and the non-optimized version had the poorest results. These outcomes confirm IPLO's superiority in managing dynamic electricity pricing scenarios, offering an improved cost-effectiveness and reliability over other contemporary optimization methods; Figures 9–11 show the results of charging and discharging operations for this case study.

Table 8. Scheduling with limiting cycles for 10 time intervals with electricity pricing uncertainty at $k = 2$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
Initial R_k				7	12	8	11	9	13	7	12	10	
New R_k				7	11	8	11	9	13	7	12	10	
IPLO	20%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.92%	39.93%	79.93%	39.94%	80%	
		Cost		71,669.88	-101,309.33	81,908.63	-76,197.09	92,106.01	-119,772.27	71,673.93	-110,575.76	100,764.69	10,268.68
	10%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.92%	39.93%	79.93%	39.94%	80%	
		Cost		71,669.88	-101,309.33	81,908.63	-76,197.09	92,106.01	-119,772.27	71,673.93	-110,575.76	100,764.69	10,268.68
PLO	20%	SOC	30%	69.99%	38.81%	78.8%	40%	80%	40%	80%	40%	80%	
		Cost		71,678.208	-79,045.40	81,917.952	-98,354.99	92,157.696	-119,808	71,680	110,580.94	102,389.76	12,034.27
	10%	SOC	30%	70%	30%	69.99%	40%	79.99%	40%	80%	40%	80%	
		Cost		71,680	-101,376	81,917.95	-76,024.39	92,153.08	-119,802	71,678.20	-110,592	102,400	12,034.8
DBO	20%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,680	-101,376	81,920	-101,376	92,160	-119,808	71,680	-82,944	102,400	14,336
	10%	SOC	30%	70%	38.57%	78.57%	40%	80%	40%	80%	40%	80%	
		Cost		71,680	-79,651.94	81,920	-97,756.06	92,160	-119,808	71,680	-110,592	102,400	12,032
MFO	20%	SOC	30%	70%	30%	70%	30%	70%	30%	70%	40%	80%	
		Cost		71,680	-101,376	81,920	-101,376	92,160	-119,808	71,680	-82,944	102,400	14,336
	10%	SOC	30%	70%	35.88%	75.88%	37.08%	77.08%	37.08%	77.08%	40%	80%	
		Cost		71,680	-86,451.53	81,920	-98,353.02	92,160	-119,808	71,680	-102,523.03	102,400	12,704.41
HO	20%	SOC	30%	70%	40.36%	80%	42.55%	80%	40%	80%	53.08%	80%	
		Cost		71,679.98	-75,108.22	81,173.53	-94,901.86	86,274.42	-119,807.02	71,679.42	-74,412.33	68,900.31	15,478.21
	10%	SOC	30%	70%	42.14%	80%	52.46%	80%	40%	80%	41.07%	80%	
		Cost		71,680	-70,606.8	77,536	-69,794	63,448.59	-119,807.74	71,679.84	-107,613.46	99,642.09	16,165.08
SH	20%	SOC	30%	69.9%	30%	70%	30%	69.9%	30%	69.5%	40%	80%	
		Cost		71,540	-101,180	81,910	-101,360	91,840	-119,390	70,830	-81,630	102,400	14,950
	10%	SOC	30%	69.5%	30%	69.5%	30%	69.7%	30%	70%	40%	80%	
		Cost		70,750	-100,060	80,920	-100,140	91,490	-118,940	71,640	-82,880	102,400	15,180
WOA	20%	SOC	30%	70%	32.18%	72.18%	32.19%	72.18%	32.18%	72.18%	40.37%	80%	
		Cost		71,680	-95,832.96	81,912.29	-101,351.64	92,137.91	-119,807.98	71,679.68	-87,941.4	101,438.21	13,914.10
	10%	SOC	30%	70%	39.59%	79.59%	43.97%	80%	40%	80%	40.90%	80%	
		Cost		71,679.83	-77,068.66	81,920	-90,261.93	82,998.93	-119,808	71,680	-108,090.33	100,083.64	13,133.48
COA	20%	SOC	30%	69.99%	47.26%	67.90%	30%	58.37%	30%	63.07%	40.13%	80%	
		Cost		71,673.55	-57,616.07	42,277.82	-96,069.62	65,365.77	-84,975.50	59,273.81	-63,435.03	102,059.26	12,038
	10%	SOC	30%	63.99%	30%	57.08%	30%	54.81%	30%	69.99%	41.82%	80%	
		Cost		60,920.19	-86,158.55	55,459.68	-68,631.36	57,161.35	-74,309.76	71,678.64	-77,896.28	97,728.11	35,952.03
Non-Optimized	20%	SOC	30%	50%	30%	70%	50%	80%	30%	60%	30%	80%	
		Cost		35,840	-50,688	81,920	-50,688	69,120	-149,760	53,760	-82,944	128,000	34,560
	10%	SOC	30%	60%	30%	70%	40%	50%	30%	70%	30%	80%	
		Cost		53,760	-76,032	81,920	-76,032	23,040	-59,904	71,680	-110,592	128,000	35,840

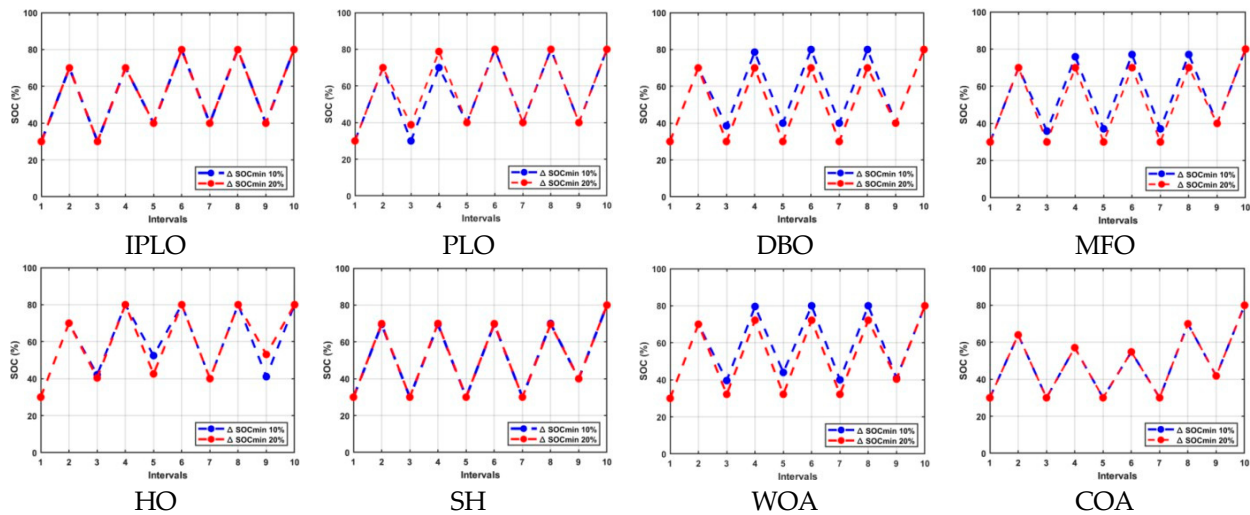


Figure 9. Charging and discharging cycles with 10% and 20% cycle limits over nine intervals under $k = 2$ uncertainty.

Table 9. Scheduling with limiting cycles under electricity price uncertainty at $k = 3$ and $k = 5$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
Initial R_k				7	12	8	11	9	13	7	12	10	
New R_k				7	12	9	11	10	13	7	12	10	
IPLO	20%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.23%	39.24%	79.23%	39.24%	80%	
		Cost		71,669.88	-110,519.2 7	92,147.21	-76,197.09	100,557.21	-119,772.2 7	71,673.93	-110,575.7 6	100,764.69	19,748.52
	10%	SOC	30%	69.99%	30.02%	70.01%	39.95%	79.23%	39.24%	79.23%	39.24%	80%	
		Cost		71,669.88	-110,519.2 7	92,147.21	-76,197.09	100,557.21	-119,772.2 7	71,673.93	-110,575.7 6	100,764.69	19,748.52
PLO	20%	SOC	30%	69.99%	30%	69.99%	49.73%	79.99%	40%	80%	40%	80%	
		Cost		71,678.20	-110,589.2 3	92,150.78	-51,352.01	77,475.84	-119,802	71,680	-110,589.2 3	102,397.44	23,049.62
	10%	SOC	30%	70%	30%	69.99%	59.93%	80%	40%	80%	40%	80%	
		Cost		71,680	-110,592	92,143.87	-25,503.66	51,379.2	-119,808	71,680	-110,586.4 7	102,394.88	22,788.01
DBO	20%	SOC	30%	69.99%	30%	70%	49.99%	78.59%	38.59%	78.49%	40%	80%	
		Cost		71,678.99	-110,590.4 4	92,159.36	-50,692.74	73,212.55	-119,807.8 4	71,497.89	-106,426.8 1	102,396.3	23,427.27
	10%	SOC	30%	70%	30%	70%	80%	70%	30%	70%	40%	80%	
		Cost		71,680	-110,592	92,160	25,344	-25,600	-119,808	71,680	-82,944	102,400	24,320
MFO	20%	SOC	30%	70%	30.30%	70.29%	39.58%	79.58%	39.58%	79.53%	40.64%	80%	
		Cost		71,680	-109,760.7 6	92,154.1	-77,848.09	102,399.58	-119,806.8 4	71,587.35	-107,503.8 9	100,743.75	23,645
	10%	SOC	30%	70%	30%	70%	80%	70%	30%	70%	40%	80%	
		Cost		71,680	-110,592	92,160	25,344	-25,600	-119,808	71,680	-82,944	102,400	24,320
HO	20%	SOC	30%	69.99%	30.33%	70.11%	46.58%	80%	40.56%	80%	54.73%	80%	
		Cost		71,673.03	-109,645.1 3	91,635.74	-59,624.17	85,542.17	-118,129.7 9	70,675.94	-69,865.28	64,690.07	26,952.58
	10%	SOC	30%	70%	30.2%	70.2%	47.76%	80%	40.03%	80%	57.63%	80%	
		Cost		71,679.24	-110,022.5 6	92,154.07	-56,868.91	82,523.76	-119,704.8 6	71,618.29	-61,826.21	57,246.49	26,799.30
SH	20%	SOC	30%	67.44%	30%	65.36%	39.34%	68.7%	30%	69.47%	40%	80%	
		Cost		67,106.80	-103,536.2 1	81,478.04	-65,938.49	75,150.69	-115,920.4 5	70,746.65	-81,503.98	102,400	29,983.05
	10%	SOC	30%	70%	30%	69.5%	79.6%	69.3%	30%	69.8%	40%	80%	
		Cost		71,640	-110,530	90,910	25,580	-26,310	-117,630	71,390	-82,500	102,400	24,950
WOA	20%	SOC	30%	70%	30%	69.99%	30%	69.99%	30%	69.99%	40%	80%	
		Cost		71,680	-110,592	92,157.40	-101,373.1 5	102,395.63	-119,802.8 8	71,675.54	-82,934.73	102,397.78	25,603.6
	10%	SOC	30%	70%	30.30%	70.29%	39.58%	79.58%	39.58%	79.53%	40.64%	80%	
		Cost		71,680	-109,760.7 6	92,154.1	-77,848.09	102,399.58	-119,806.8 4	71,587.35	-107,503.8 9	100,743.75	23,645.19

COA	20%	SOC	30%	69.97%	35.65%	71.62%	53.96%	75.99%	43.79%	74.71%	40.09%	80%	36,667.32
		Cost		71,634.51	−94,899.44	82,886.08	−44,755.29	56,398.46	−96,448.68	55,398.66	−95,700.04	102,153.06	
	10%	SOC	30%	55.04%	30.98%	46.68%	32.18%	65.56%	30.20%	70.19%	58.01%	80%	51,603.27
		Cost		44,880.76	−66,511.14	36,162.97	−36,759.15	85,463.03	−105,904.33	71,651.78	−33,653.43	56,272.78	
Non-Optimized	20%	SOC	30%	70%	30%	70%	40%	80%	40%	60%	30%	80%	40,704
		Cost		71,680	−110,592	92,160	−76,032	102,400	−119,808	35,840	−82,944	128,000	
	10%	SOC	30%	60%	30%	70%	40%	50%	30%	60%	40%	80%	
		Cost		53,760	−82,944	92,160	−76,032	25,600	−59,904	53,760	−55,296	102,400	

Table 10. Scheduling with limiting cycles for 10 time intervals with electricity pricing uncertainty at $k = 6$.

Algorithm	ΔSOC_{min}	k	0	1	2	3	4	5	6	7	8	9	J_k
Initial R_k				7	12	8	11	9	13	7	12	10	
New R_k				7	12	8	11	9	8	7	12	10	
IPLO	20%	SOC	30%	69.99%	30%	69%	30%	30%	40%	80%	40%	80%	21,779.59
		Cost		71,676.13	−110,586.03	81,919.36	−101,375.07	−0.122	18,707.35	71,412.29	−110,575.03	100,600.72	
	10%	SOC	30%	69.99%	30%	69%	30%	30%	40%	80%	40%	80%	
		Cost		71,676.13	−110,586.03	81,919.36	−101,375.07	−0.122	18,707.35	71,412.29	−110,575.03	100,600.72	
PLO	20%	SOC	30%	69.99%	31.26%	71.26%	50%	30%	51.75%	80%	40%	80%	29,056.53
		Cost		71,676.41	−107,097.29	81,915.9	−53,868.67	−46,091.52	40,087.75	50,625.79	−110,578.17	102,387.2	
	10%	SOC	30%	70%	30%	70%	40%	30%	40%	80%	40%	80%	
		Cost		71,680	−110,589.23	81,917.95	−76,032	−23,040	18,432	71,680	−110,592	102,400	
DBO	20%	SOC	30%	70%	40%	80%	50%	30%	50%	80%	40%	80%	30,976
		Cost		71,680	−82,944	81,920	−76,032	−46,080	36,864	53,760	−110,592	102,400	
	10%	SOC	30%	70%	30%	70%	40%	30%	40%	80%	40%	80%	
		Cost		71,680	−82,944	81,920	−101,376	−23,040	18,432	71,680	−110,592	102,400	
MFO	20%	SOC	30%	70%	30%	70%	30%	60%	40%	80%	40%	80%	37,376
		Cost		71,680	−110,592	81,920	−101,376	69,120	−36,864	71,680	−110,592	102,400	
	10%	SOC	30%	70%	30%	70%	40%	30%	40%	80%	40%	80%	
		Cost		71,680	−82,944	81,920	−101,376	−23,040	18,432	71,680	−110,592	102,400	
HO	20%	SOC	30%	70%	33.05%	75.05%	36.05%	58.05%	38.72%	77.72%	44.72%	80%	42,034.7263
		Cost		71,679.99	−110,591.55	61,548.09	25,344.06	−92,157.58	19,661.25	70,388.09	−51,808.24	47,970.61	
	10%	SOC	30%	70%	30.07%	70.07%	30.23%	51.79%	40.84%	80%	45.06%	80%	
		Cost		71,679.99	−110,391.57	81,919.95	−100,959.80	−49,679.20	20,192.03	70,167.61	−96,579.23	89,425.21	
SH	20%	SOC	30%	69.9%	30%	67%	30%	50%	30%	69.7%	40%	80%	43,790
		Cost		71,520	−110,350	75,720	−93,710	46,080	−36,860	71,220	−82,160	102,320	
	10%	SOC	30%	69.37%	30%	68.54%	40.05%	30%	40.05%	79.22%	79.22%	80%	
		Cost		70,562.35	−108,867.62	78,933.04	−72,201.60	23,161.85	18,433.93	70,285.62	−108,443.43	102,400	
WOA	20%	SOC	30%	70%	30%	66.32%	30%	69.2%	41.9%	80%	40%	80%	43,502.53
		Cost		71,680	−110,592	74,384.07	−92,050.28	90,315.8	−50,311.35	68,268.19	−110,590.51	102,398.62	
	10%	SOC	30%	70%	30%	58.08%	68.19%	30%	44.31%	80%	40.01%	80%	
		Cost		71,680	−110,591.99	57,509.71	25,637.60	−88,005.34	26,393.21	63,939.93	−110,563	102,373.14	
COA	20%	SOC	30%	69.99%	45.19%	74.40%	42.93%	68.01%	31.18%	67.37%	40.03%	80%	64,635.66
		Cost		71,667.91	−68,571.87	59,827.01	−79,764.64	57,783.51	−67,878.10	64,856.27	−75,602.32	102,317.9	
	10%	SOC	30%	69.98%	35.60%	71.08%	32.26%	59.85%	43.82%	66.20%	40.01%	80%	
		Cost		71,658.94	−95,057.07	72,662.87	−98,379.32	63,556.12	−29,554.75	40,121.46	−72,436.67	102,375.33	
Non-Optimized	20%	SOC	30%	60%	30%	70%	40%	70%	30%	70%	40%	80%	63,232
		Cost		53,760	−82,944	81,920	−76,032	69,120	−73,728	71,680	−82,944	102,400	
	10%	SOC	30%	60%	30%	70%	60%	80%	40%	60%	30%	80%	
		Cost		53,760	−82,944	81,920	−25,344	46,080	−73,728	35,840	−82,944	128,000	

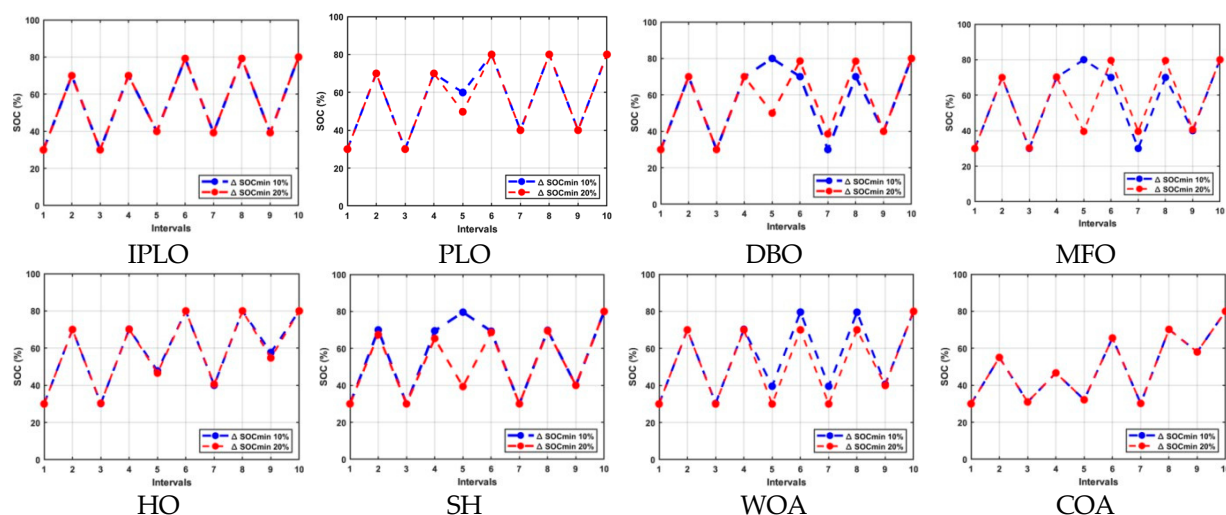


Figure 10. Discharging and charging cycles with 10% and 20% limiting cycles at $k = 3$ and $k = 5$ uncertainty.

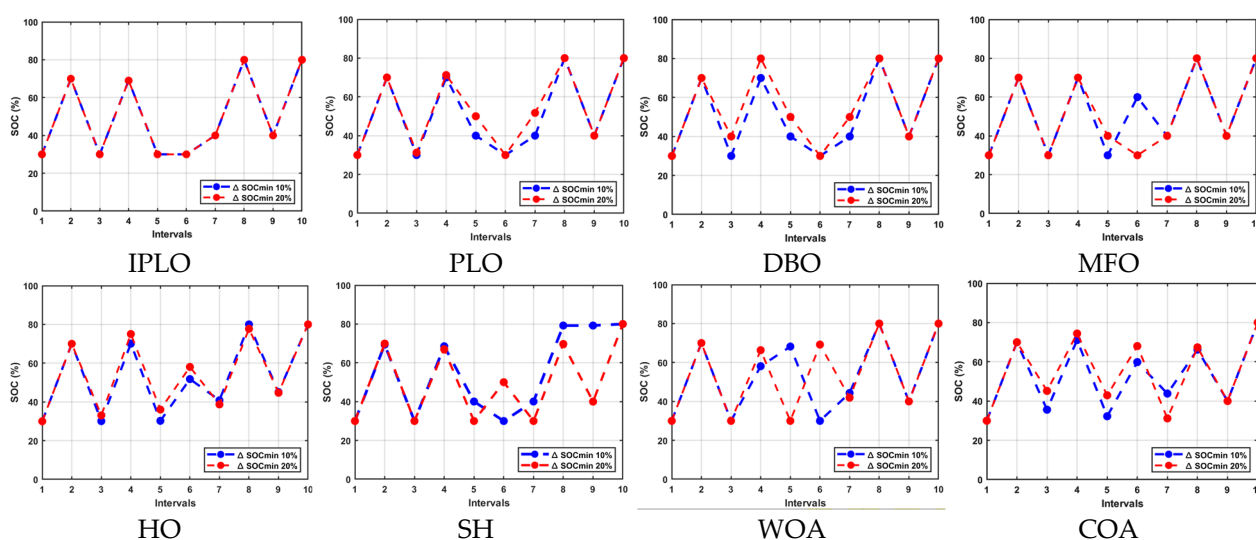


Figure 11. Discharging and charging operations with 10% and 20% limiting cycles for nine intervals at $k=6$ uncertainty.

Table 10 shows the scheduling results under extended uncertainty at the interval $k = 6$. The IPLO maintained a superior cost efficiency, achieving Rs 21,779.59 for both 10% and 20% limiting cycles (l.c.), demonstrating a high adaptability and consistent performance even at late-stage price fluctuations. PLO followed, with Rs 29,056.53 and Rs 25,857.8 for 20% and 10%, respectively. SH ranked third under the 10% l.c., with Rs 27,940.29, but its cost rose sharply to Rs 43,790 under the 20% l.c., reflecting a limited robustness. DBO followed, with Rs 28,160 (10%) and Rs 30,976 (20%), while MFO matched DBO at Rs 28,160 (10%) but increased to Rs 37,376 (20%), indicating a sensitivity to higher uncertainty. HO, WOA, and COA recorded significantly higher costs, suggesting a weaker performance in uncertain conditions. The non-optimized case scored the highest cost (Rs 80,640 at 10% l.c.), highlighting the necessity of optimization for stable, cost-effective operation. This case highlights IPLO's ability to adapt to price changes and maintain performance over different stages. Notably, IPLO kept a minimal gap between the two l.c. values, offering flexibility and ensuring battery safety. Despite the extended uncertainty, IPLO consistently outperformed others, confirming its robustness and reliability in dynamic environments.

3.4. Comparative Study

3.4.1. Performance, Robustness, and Stability

A comprehensive comparative evaluation was conducted to assess the performance, robustness, and stability of the IPLO against a diverse set of state-of-the-art algorithms across multiple operating scenarios. The analysis integrates descriptive statistics, Friedman ranking (as shown in Table 11), and pairwise Wilcoxon rank-sum tests (illustrated in Table 12) to ensure statistically reliable conclusions. The Friedman test revealed statistically significant performance differences among the competing algorithms ($p < 0.05$). Across all benchmark cases, IPLO consistently achieved the lowest average rank (1.00), confirming its superior overall performance. PLO and MFO formed the second performance tier, while DBO followed closely. In contrast, HO, SH, WOA, and COA consistently occupied lower ranks, reflecting a weaker solution quality and reduced robustness across varying scenarios. Pairwise Wilcoxon rank-sum tests further validated these findings. IPLO achieved statistically significant wins against nearly all competing algorithms in every scenario, indicating a reliable performance under varied conditions.

Table 11. Friedman test ranking results.

Case	Intervals	IPLO	PLO	DBO	MFO	HO	SH	WOA	COA
I	/	1	2	4	3	6	5	7	8
	K = 1	1	2	4	3	6	5	7	8
II	K = 4	1	2	3	4	6	5	7	8
	K = 7	1	2	4	3	7	5	6	8
	K = 2	1	2	3	4	6	5	7	8
III	K = 3 & 5	1	2	3	3	6	5	7	8
	K = 6	1	2	4	3	7	5	6	8
Average ranks		1.00	2.00	3.57	3.29	6.29	5.00	6.71	8.00

Table 12. Wilcoxon test results.

IPLO vs.		PLO		DBO		MFO		HO		SH		WOA		COA	
Case	Interval	P	win	P	win	P	win	P	win	P	win	P	win	P	win
I	/	5.81×10^{-1}	+	5.42×10^{-1}	+	5.96×10^{-1}	+	5.96×10^{-1}	+	5.95×10^{-1}	+	5.95×10^{-1}	+	5.22×10^{-1}	+
	K = 1	8.83×10^{-1}	+	8.82×10^{-1}	+	8.86×10^{-1}	+	8.84×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	8.77×10^{-1}	+
II	K = 4	8.12×10^{-1}	+	8.57×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	8.11×10^{-1}	+
	K = 7	8.72×10^{-1}	+	8.73×10^{-1}	+	8.86×10^{-1}	+	8.84×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	8.63×10^{-1}	+
	K = 2	6.13×10^{-1}	+	4.37×10^{-1}	+	8.86×10^{-1}	+	8.43×10^{-1}	+	8.86×10^{-1}	+	8.86×10^{-1}	+	1.19×10^{-1}	+
III	K = 3 & 5	7.94×10^{-1}	+	5.10×10^{-1}	+	8.86×10^{-1}	+	8.43×10^{-1}	+	8.86×10^{-1}	+	8.84×10^{-1}	+	5.10×10^{-1}	+
	K = 6	7.74×10^{-2}	+	7.74×10^{-2}	+	8.86×10^{-1}	+	8.18×10^{-1}	+	8.73×10^{-1}	+	1.19×10^{-1}	+	7.74×10^{-2}	+
Rank-sum test (+/=/-)		7/0/0		7/0/0		7/0/0		7/0/0		7/0/0		7/0/0		7/0/0	

The descriptive statistical analysis shown in Table 13 reinforces these conclusions. IPLO exhibited the smallest ranges and lowest standard deviations in nearly all cases, indicating a highly stable and repeatable performance. While PLO showed zero variance

due to its deterministic nature, it consistently delivered an inferior solution quality compared to IPLO. Algorithms such as WOA, COA, and HO displayed large ranges and standard deviations, reflecting unstable search behavior and susceptibility to premature convergence or erratic exploration. In contrast, IPLO effectively balanced exploration and exploitation, achieving a low variability while maintaining a superior cost performance. To assess robustness under uncertainty, the coefficient of variation (CV) was used as a robust metric, defined as the ratio of the standard deviation to the mean objective value across 20 runs. The original PLO achieved the lowest average CV, reflecting a high stability but limited adaptability due to fixed parameters. The IPLO attained the second-lowest average CV (0.0046), demonstrating an excellent robustness while preserving adaptive search capability. Compared with competing metaheuristics (DBO, MFO, HO, SH, WOA, and COA), IPLO exhibits a substantially lower variability across all scenarios, confirming that the integration of Random Walk Exploitation and Periodic Random Parameter Tuning improves stability without compromising adaptability. These findings complement the Friedman and Wilcoxon tests, indicating that IPLO delivers not only a superior cost performance but also consistent and reliable behavior under diverse operating conditions, which is essential for practical EV charging optimization.

Table 13. Statistical results.

Case	Algorithm	Min	Max	Range	Mean	Median	STD	CV	RT	
I	IPLO	1050.23	1060.00	9.77	1053.33	1054.00	3.26	0.0031	12.31	
	PLO	2817.94	2817.94	0.00	2817.94	2817.94	0.00	0.0000	12.58	
	DBO	2816.00	5120.00	2304.00	4251.57	5120.00	1134.00	0.2667	5.84	
	MFO	3493.13	5728.00	2234.87	3599.55	3493.13	487.69	0.1355	2.48	
	HO	20,006.00	39,021.00	19,015.00	27,529.48	27,581.00	5146.01	0.1869	59.42	
	SH	5368.00	7720.00	2352.00	6147.14	5966.00	735.80	0.1197	6.47	
	WOA	11,098.00	55,164.00	44,066.00	31,432.05	33,758.00	13,464.82	0.4284	5.38	
	COA	20,493.00	55,996.00	35,503.00	37,201.81	35,843.00	10,039.34	0.2699	3.71	
II	K = 1	IPLO	−1107.59	−1057.00	50.59	−1092.38	−1094.00	11.26	0.0103	12.59
		PLO	687.00	687.00	0.00	687.00	687.00	0.00	0.0000	12.69
		DBO	666.00	11,162.00	10,496.00	2337.95	2811.50	2362.27	10.104	3.27
		MFO	666.00	2970.00	2304.00	1322.85	666.00	975.29	0.7373	4.56
		HO	18,122.00	40,209.00	22,087.00	26,644.90	26,043.50	5752.41	0.2159	66.25
		SH	3241.00	5736.00	2495.00	3928.25	3633.00	769.94	0.1960	5.32
		WOA	8947.00	54,319.00	45,372.00	29,352.05	30,174.00	14,052.78	0.4788	5.4
		COA	23,716.61	54,582.00	30,865.39	36,045.96	34,029.00	9371.15	0.2600	4.09
	K = 4	IPLO	3329.54	3350.00	20.46	3334.49	3331.50	6.40	0.0019	12.94
		PLO	5122.00	5122.00	0.00	5122.00	5122.00	0.00	0.0000	12.68
		DBO	5495.21	11,162.00	5666.79	5778.55	5495.21	1267.13	0.2193	3.11
		MFO	6836.98	8646.00	1809.02	7192.24	6836.98	639.10	0.0889	4.52
		HO	18,122.00	40,209.00	22,087.00	26,765.70	26,043.50	5707.88	0.2133	65.89
		SH	8389.00	11,171.00	2782.00	9383.95	9323.50	757.18	0.0807	5.14
		WOA	8947.00	54,319.00	45,372.00	29,062.30	30,174.00	13,908.75	0.4786	5.46
		COA	23,147.74	55,704.00	32,556.26	37,986.67	34,590.00	9522.92	0.2507	3.59
	K = 7	IPLO	−1107.00	−1023.00	84.00	−1095.40	−1100.00	18.20	0.0166	12.64
		PLO	688.00	688.00	0.00	688.00	688.00	0.00	0.0000	12.88
		DBO	818.10	5292.00	4473.90	2219.69	2970.00	1273.96	0.5739	3.29
		MFO	856.91	3028.00	2171.09	1273.08	856.91	853.71	0.6706	4.04
		HO	15,358.00	39,418.00	24,060.00	29,338.60	30,488.00	6521.07	0.2223	59.24
		SH	3274.00	5103.00	1829.00	3839.50	3659.00	587.61	0.1530	4.77
		WOA	9077.00	53,013.00	43,936.00	28,232.65	26,403.00	13,908.15	0.4926	5.49
		COA	21,484.07	54,176.00	32,691.93	36,007.26	34,033.00	9861.02	0.2739	3.7
III	K = 2	IPLO	10,268.68	10,269.00	0.32	10,268.70	10,268.68	0.07	0.0000	12.60
		PLO	12,043.00	12,043.00	0.00	12,043.00	12,043.00	0.00	0.0000	12.87
		DBO	12,032.00	14,336.00	2304.00	12,953.70	12,032.00	1157.96	0.0894	3.97

k = 3 & k=5	MFO	12,905.75	14,337.00	1431.25	13,263.11	12,905.75	635.05	0.0479	3.85
	HO	23,889.00	41,081.00	17,192.00	33,112.50	33,769.00	5060.49	0.1528	62.57
	SH	14,950.00	16,030.00	1080.00	15,259.35	15,152.00	353.18	0.0231	6.57
	WOA	18,009.00	51,564.00	33,555.00	35,667.85	36,523.00	11,357.97	0.3184	5.9
	COA	29,706.00	59,147.00	29,441.00	43,463.40	44,320.50	8775.85	0.2019	3.65
	IPLO	19,748.52	19,777.00	28.48	19,753.54	19,748.52	8.99	0.0005	12.85
	PLO	22,788.01	22,788.01	0.00	22,788.01	22,788.01	0.00	0.0000	12.70
	DBO	23,427.27	30,464.00	7036.73	24,283.52	24,260.50	1565.20	0.0645	4.01
	MFO	23,636.09	23,636.09	0.00	23,636.09	23,636.09	0.00	0.0000	4.73
	HO	36,360.00	48,828.00	12,468.00	41,547.10	41,311.50	3870.95	0.0932	59.17
k = 6	SH	24,950.00	27,335.00	2385.00	25,592.85	25,248.00	827.53	0.0323	5.72
	WOA	29,255.00	55,073.00	25,818.00	40,112.20	38,640.00	8713.79	0.2172	5.87
	COA	36,667.32	54,922.00	18,254.68	45,065.53	43,551.50	6429.19	0.1427	3.87
	IPLO	21,779.59	21,779.59	0.00	21,779.59	21,779.59	0.00	0.0000	12.70
	PLO	25,857.80	25,857.80	0.00	25,857.80	25,857.80	0.00	0.0000	13.41
	DBO	28,160.00	28,160.00	0.00	28,160.00	28,160.00	0.00	0.0000	3.55
	MFO	28,160.00	28,160.00	0.00	28,160.00	28,160.00	0.00	0.0000	3.85
	HO	39,862.00	56,103.00	16,241.00	48,465.75	49,232.50	4597.96	0.0949	59.88
	SH	27,760.00	34,359.00	6599.00	30,436.35	29,144.50	2759.71	0.0907	5.76
	WOA	43,502.53	73,748.00	30,245.47	51,051.61	48,486.00	8708.15	0.1706	5.82
	COA	54,946.90	55,996.00	1049.10	54,999.36	54,946.90	234.59	0.0043	3.67

3.4.2. Execution Time

Runtime Analysis

In terms of execution time, DBO, MFO, and COA ranged between 3 and 7 s, SH and WOA took between 5 and 6 s, and both IPLO and the original PLO completed their runs within 12 to 13 s; HO was the slowest, requiring 59 to 66 s. Despite IPLO's slightly longer runtime, its efficiency and consistent performance justify the trade-off. The average runtime and its standard deviation in Table 14 were computed across all tested cases under identical experimental settings (same population size, function evaluations, and hardware). The standard deviation reflects the stability of the runtime and sensitivity to problem variations.

Table 14. Statistical analysis.

Algorithm	Avg. Runtime (s)	STD
IPLO	12.70	±0.18
PLO	12.83	±0.16
DBO	3.58	±0.31
MFO	4.00	±0.29
HO	61.60	±3.12
SH	5.54	±0.62
WOA	5.62	±0.55
COA	3.78	±0.34

IPLO exhibits a runtime comparable to the original PLO, confirming that RWE and PRPT introduce no significant computational overhead. The low standard deviation for IPLO indicates a stable execution time across different uncertainty scenarios. HO shows the highest runtime and variability, reflecting its heavier internal operators. Lightweight algorithms (DBO, COA) are faster but, as shown previously, provide inferior robustness and solution quality. IPLO achieves a balanced trade-off between accuracy, robustness, and efficiency, which is critical for online EV charging scheduling.

Runtime Complexity

The computational complexity of the original PLO per iteration is

$$\mathcal{O}(P \cdot D + P \cdot F(X)) \quad (31)$$

where P is the population size, D is the problem dimension, and $F(X)$ denotes the cost of evaluating the objective function. Over T iterations, the total complexity is

$$\mathcal{O}(T(P \cdot D + P \cdot F(X))) \quad (32)$$

The IPLO preserves the same asymptotic complexity. RWE introduces a linear local perturbation step $\mathcal{O}(D)$ applied with fixed probability, while the PRPT incurs negligible overhead due to infrequent scalar updates. The price–degradation-aware operational mechanism is embedded within the objective function and adds only linear operations with respect to D , thereby increasing the constant factor of $F(X)$ without altering its order. Consequently, IPLO achieves an improved convergence accuracy, robustness, and adaptability without increasing the asymptotic computational complexity compared to the original PLO.

3.4.3. Convergence

Figures 12–14 illustrate the convergence behavior of all algorithms under different cases and uncertainty levels. In all scenarios, IPLO demonstrates a fast and smooth convergence and achieves lower final fitness values compared to PLO and all other competing algorithms across all scenarios. The enhanced IPLO exhibits a steeper descent in the early iterations and a stable convergence pattern in the later stages, with almost no clear oscillations. While PLO shows oscillations especially during the early stage, IPLO continues refining the solution and consistently reaches better optima. Other algorithms such as SH and HO displayed a delayed convergence with harsh oscillations and aggressive descents, or stagnation at higher fitness levels, such as with COA. DBO, MFO and WOA display more smoothness with the least oscillations and stabilize to a solution around the 1000th iteration, which is a sign of local optimality, while IPLO continues its smooth descents to the optimal solution, highlighting the effectiveness of the presented mechanism improvement. Overall, the convergence curves clearly confirm the high performances of IPLO in terms of convergence speed, stability, and solution quality.

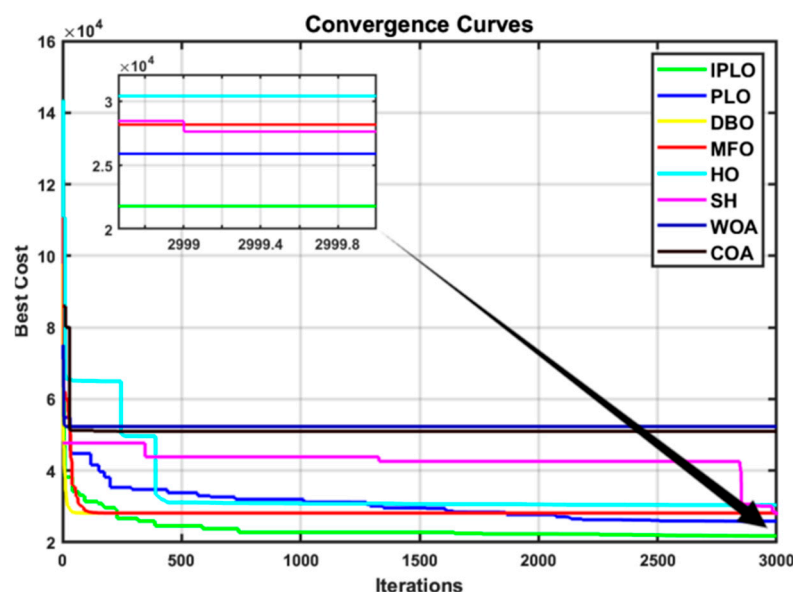


Figure 12. Case I convergence curves.

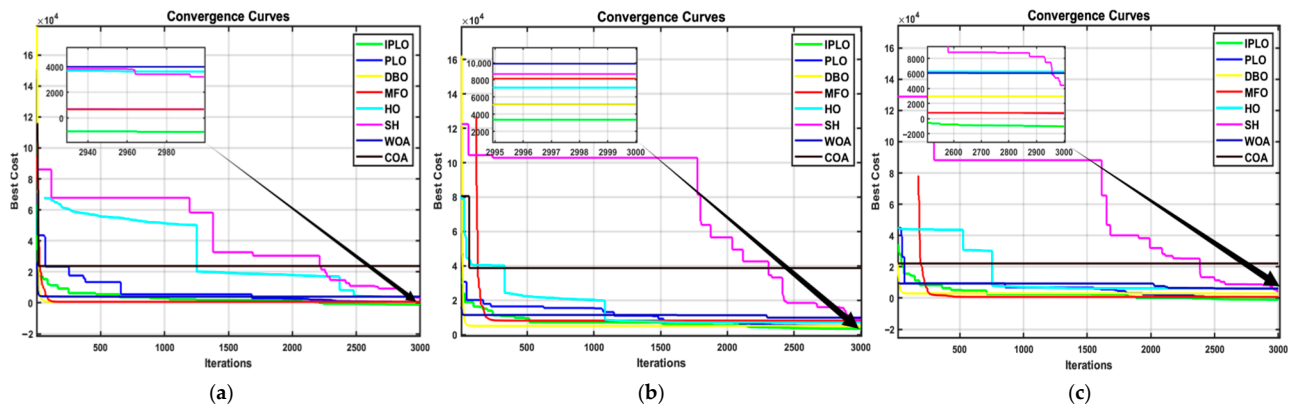


Figure 13. (a) Case II, uncertainty at K1. (b) Case II, uncertainty at K4. (c) Case II, uncertainty at K7.

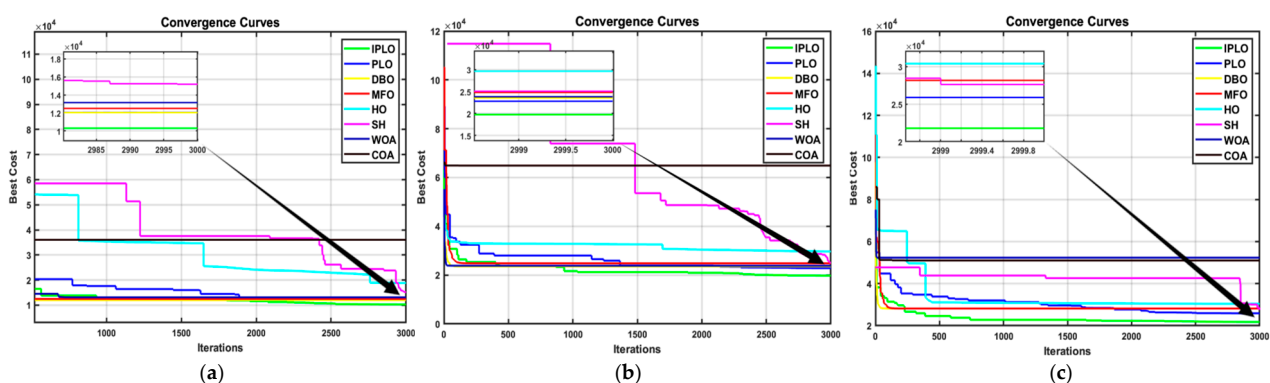


Figure 14. (a) Case III, uncertainty at K2. (b) Case III, uncertainty at K3 and K5. (c) Case III, uncertainty at K6.

3.4.4. Box Plot Analysis

Figure 15 presents the Boxplot comparison of the best fitness values obtained by different optimization algorithms across seven test cases: (a)–(g). In each boxplot, the blue box represents the interquartile range (IQR), the red horizontal line indicates the median fitness value, and whiskers denote the minimum and maximum values across independent runs. Across all box plot scenarios, the IPLO algorithm consistently presents the lowest median fitness values and the most compact box distributions, indicating a superior solution quality and high robustness. The narrow interquartile ranges and short whiskers confirm that IPLO exhibits a low variability and strong stability over repeated runs. In contrast, competing algorithms, particularly COA and WOA, show wider and higher-positioned boxes, reflecting an increased dispersion and inferior performance. Although some methods such as GWO and MVO demonstrate reasonable stability, their median values remain consistently above those of IPLO, indicating a convergence toward sub-optimal solutions. Overall, the boxplot analysis clearly demonstrates that IPLO achieves the most accurate and consistent optimization performance among all compared algorithms. The stability and narrow variability of IPLO's performance further support the conclusion that the presented algorithm offers a superior efficiency and robustness in EV charging cost optimization. These results confirm that IPLO significantly outperforms all other tested algorithms.

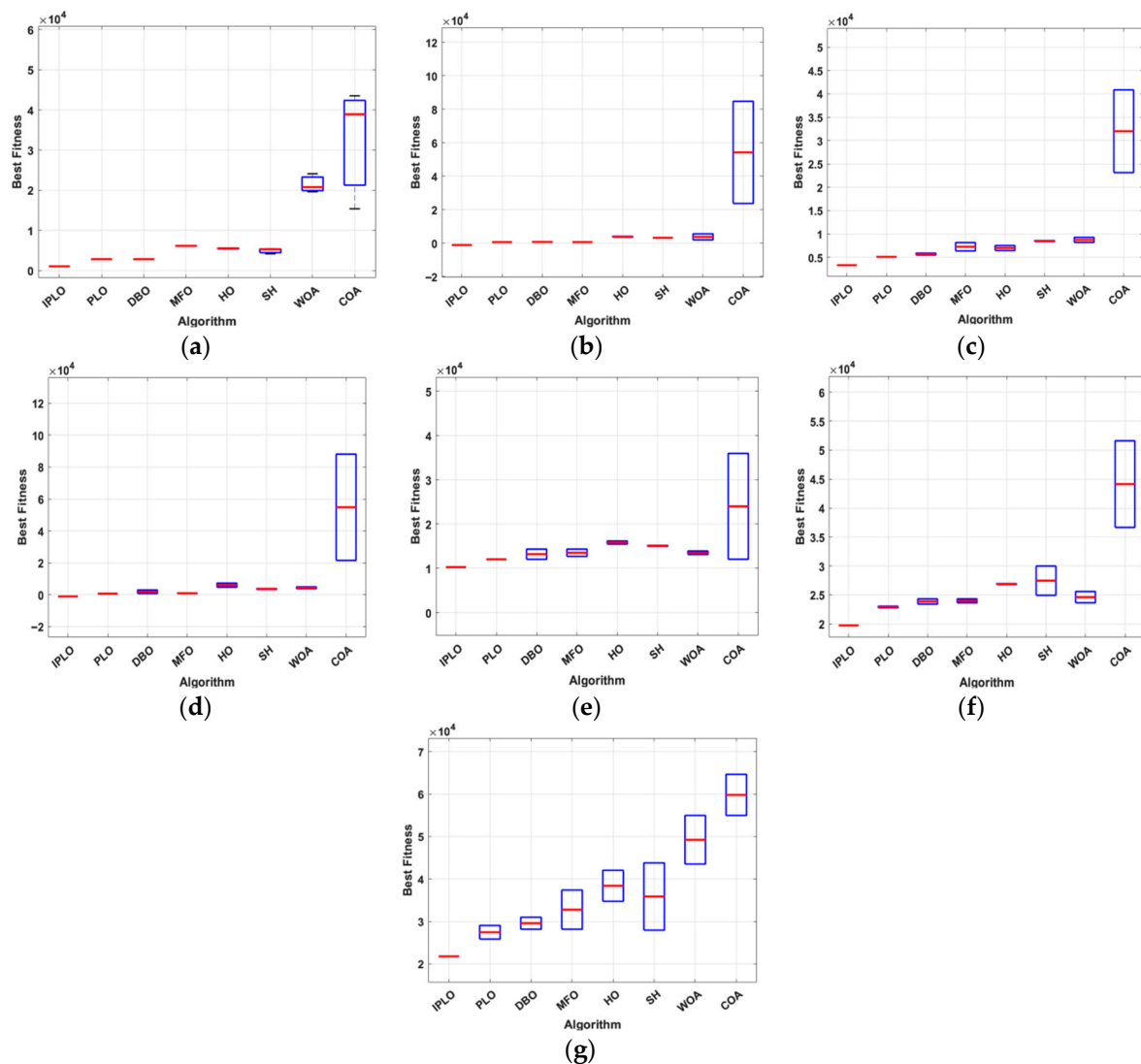


Figure 15. Boxplots: (a) Case I. (b) Case II, uncertainty K1. (c) Case II, uncertainty K4. (d) Case II, uncertainty K7. (e) Case III, uncertainty K2. (f) Case III, uncertainty K3 and K5. (g) Case III, uncertainty K6.

4. Discussion

The main objective of this research is to minimize EV charging and discharging costs under various constraints and uncertainties using the IPLO algorithm. In the first case study, IPLO is tested under two scenarios: without a limit on charging/discharging cycles and with a limit on it. As shown in Table 2, IPLO consistently achieves the lowest costs in both scenarios, maintaining its performance regardless of the cycle limit, with the same number of search agents and iterations, and outperforming the original PLO, highlighting its robustness and the efficacy of the improvement model implemented. A constraint “Constraint-V” is introduced (Table 3), which limits frequent mode switching based on SOC, using thresholds of 20% and 10%. Since cost outcomes remain constant across both limits, the tighter 10% $\Delta\text{SOC}_{\min}$ is recommended for an improved battery longevity.

The second case study explores IPLO’s resilience under battery condition uncertainties. As shown in Tables 4–6, IPLO maintains profitability even during critical intervals with unpredictable power levels, demonstrating an adaptability and robust performance in real-world fluctuating environments. The third case study examines the impact of electricity price fluctuations on scheduling. During V2G operations, rising prices increase user

profits, while, during G2V, falling prices reduce costs. These fluctuations positively influence total cost, emphasizing the need for dynamic scheduling. IPLO successfully adapts to such variations, optimizing cost savings by adjusting operations through price-responsive adjustments in real time. Through the analysis of SOC profiles, although the SOC curves of IPLO and the original PLO appear almost identical, particularly in the first and second study cases, this does not imply a similar performance. Even minor variations in SOC lead to substantial improvements in total cost; this behavior occurs because the RWE mechanism contributes to this behavior by introducing small, adaptive steps around promising regions of the search space, refining scheduling decisions without significantly changing the SOC profile, demonstrating that the enhancements improve economic efficiency while preserving stable battery operation. Additionally, SOC values are highly bounded by operational and health constraints; large deviations would neither be expected nor beneficial, and IPLO achieves better results by fine-tuning the timing and magnitude of charge and discharge events within these limits.

The statistical results clearly demonstrate the superiority and robustness of the IPLO. The Friedman test reveals significant performance differences among all algorithms ($p \ll 0.05$), with IPLO achieving the best overall average rank (1.00), indicating a consistent top performance across all scenarios. Wilcoxon rank-sum tests further confirm this dominance, as IPLO records statistically significant wins against nearly all competitors in every case.

Descriptive statistics reinforce these findings by highlighting IPLO's exceptional stability. Across all cases, IPLO exhibits the 2nd smallest ranges and standard deviations, confirming a reliable convergence and minimal sensitivity to scenario variations. While PLO shows zero variance due to its deterministic nature, it consistently yields higher costs, demonstrating that stability alone is insufficient without solution quality. In contrast, algorithms such as HO, WOA, and COA display large variability and wide performance ranges, reflecting unstable search behavior.

Overall, IPLO effectively balances exploration and exploitation, achieving a superior solution quality without sacrificing robustness. This combination of statistical superiority, consistency, and adaptability confirms its suitability for complex, real-world optimization problems where dependable performance across diverse operating conditions is essential.

5. Conclusions

This study has presented IPLO as an optimization framework for EV charging cost minimization under diverse operating conditions. Comprehensive statistical analyses, including Friedman rankings, Wilcoxon rank-sum tests, and descriptive metrics, consistently demonstrated that IPLO achieves a superior solution quality, stability, and robustness compared to state-of-the-art optimizers. The integration of random walks and adaptive Parameter Tuning significantly enhanced the exploration–exploitation balance, yielding substantial cost improvements and a smoother convergence behavior. Moreover, the cost–battery adaptive model strengthened decision-making by incorporating battery health and price dynamics, resulting in practical, stable, and durable scheduling outcomes. Despite the slight increase in variability compared to the original method, IPLO's reliable convergence, balanced search mechanism, and consistently superior performance justify this trade-off. Overall, IPLO emerges as a competitive and well-balanced optimizer, particularly suited for complex, dynamic EV energy management problems where accuracy, adaptability, and robustness are critical.

Author Contributions: Conceptualization, A.B. and F.J.; methodology, A.B. and F.J.; software, A.B. and S.K.; validation, A.B. and S.K.; formal analysis, S.K.; investigation, A.B. and S.K.; resources, A.B., S.K. and F.J.; data curation, A.B., S.K. and F.J.; writing—original draft, F.J.; writing—review and editing, S.K. and F.J.; visualization, A.B. and S.K.; supervision, S.K.; project administration, F.J.; funding acquisition, F.J. 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: The authors acknowledge the use of graphic elements from Canva’s library. During the preparation of this manuscript, the authors used ChatGPT(GPT-4/GPT-4.1 model). OpenAI, San Francisco, CA, USA. Available online: <https://chat.openai.com> (accessed on 15 September 2025). Gemini (Google, Gemini 1.5 model), Available online: <https://gemini.google.com> (accessed on 19 November 2025) to improve language quality and readability. The authors reviewed and edited all generated content and take full responsibility for the final content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest. The foundations had no role in the design of the study, the collection, analysis, or interpretation of data, the writing of the manuscript, or the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

Symbol	Description	Unit
i	Index of search agent/solution	—
k	Interval index	—
P	Population size	—
D	Problem dimension	—
$MaxFEs$	Maximum number of function evaluations	—
T	Total number of iterations	—
$F(k)$	Final cost function	Rs
C_k	Charging cost component	Rs
D_k	Discharging revenue/cost component	Rs
J_k	Total charging cost objective	Rs
S_k	SOC at interval k	—
S_{bk}	Charging/discharging state indicator (G2V/V2G)	—
$Q(k)$	Available battery capacity at interval k	Ah
Q	Battery capacity reference	Ah
R	Electricity price	Rs
V	Battery nominal voltage	Volt
φ_k	Battery degradation Responsive Operational Scaling Factor	—
Φ_k	Price Responsive Operational Scaling Factor	—
$\gamma_{ch}(k), \gamma_{dis}(k)$	SOH-Aware SOC Regulation Factors	—
k_{ch}, k_{dis}	Charging/discharging damping constants	—
σ	Degradation sensitivity coefficient	—
λ	Random number uniformly distributed in $([0, 1])$	—
ρ_k	Random number uniformly distributed in $([0, 1])$	—
η	Charging efficiency	—
SOC	State of charge	—
ΔSOC_{charge}	Maximum chargeable SOC	—
$\Delta SOC_{discharge}$	Maximum dischargeable SOC	—
ΔSOC_{min}	Minimum limit for SOC deviation	—
$\Delta S_{ch}, \Delta S_{dis}$	Charging and discharging SOC variations	—

$X(N, D)$	Particle of population N and dimension D position	—
LB	Lower bound	—
UB	Upper bound	—
R	Updated aurora oval weight after PRPT	—
m	Mass	—
q	Particle charge	—
v	Velocity	—
B	Magnetic field strength	—
α	Perturbation applied to damping factor	—
C	Constant	—
β	Levy flight stability control	—
d	Levy flight step size	—
Ao	Change in particle position expression	—
X_{avg}	Average position of all particles	—
r_1, r_2, r_3, r_4, r_5	Random factor	—
W_1	Local search intensity weight	—
W_2	Global search intensity weight	—
K	Collision probability factor	—
α	Step controlling factor	—
n	Random value from normal distribution	—
X_i	Particle i position	—
ΔX	Random step size	—
X_{RWE}	RWE particle updated position	—
w_{1new}	Updated local search weight	—
w_{2new}	Updated global search weight	—
α_{new}	Updated damping factor	—
$\delta\alpha$	Random damping factor perturbation	—
δw_1	Local search weight perturbation step	—
δw_2	Global search weight perturbation step	—
$\delta\alpha$	Damping factor perturbation	—

References

- Schade, C.; Aliasghari, P.; Egging-Bratseth, R.; Pfister, C. Least cost vehicle charging in a smart neighborhood considering uncertainty and battery degradation. *Batteries* **2025**, *11*, 104.
- Yadav, J.; Shukla, D.; Karvir, P. Electrical vehicle past, present and future—A review. In *Proceedings of the AIP Conference Proceedings*; AIP Publishing: Melville, NY, USA, 2024; Volume 3178, p. 070010.
- Singh, S.; Vaidya, B.; Mouftah, H.T. Smart EV charging strategies based on charging behavior. *Front. Energy Res.* **2022**, *10*, 773440.
- Eltohamy, M.S.; Saber, A.; Tawfiq, M.H.; Ahmed, M.M.R.; Alaas, Z.; Mohammed, B.; Ahmed, I.; Youssef, H.; Raouf, A. A comprehensive review of vehicle-to-grid (V2G) technology: Technical, economic, regulatory, and social perspectives. *Energy Convers. Manag. X* **2025**, *15*, 101138.
- Yadav, N.K.; Das, S. Optimal V2G and G2V scheduling for cost-effective power management in distribution systems. In *Proceedings of the IEEE 4th International Conference on Sustainable Energy and Future Electric Transportation (SEFET)*; IEEE: New Delhi, India, 2024.
- Ahwiadi, M.; Wang, W. Battery health monitoring and remaining useful life prediction techniques: A review. *Batteries* **2025**, *11*, 31.
- Alaee, P.; Bems, J.; Anvari-Moghaddam, A. A review of the latest trends in technical and economic aspects of EV charging management. *Energies* **2023**, *16*, 3669.
- Iversen, E.B.; Morales, J.M.; Madsen, H. Optimal charging of an electric vehicle using a Markov decision process. *Appl. Energy* **2014**, *123*, 1–12.
- Sahinidis, N.V. Optimization under uncertainty: State-of-the-art and opportunities. *Comput. Chem. Eng.* **2004**, *28*, 971–983.
- Sowmya, R.; Sankaranarayanan, V. Optimal vehicle-to-grid and grid-to-vehicle scheduling strategy with uncertainty management using improved marine predator algorithm. *Comput. Electr. Eng.* **2022**, *100*, 107949.
- Huang, T.; Huang, F.; Qin, Z.; Pan, J. An improved polar lights optimization algorithm for global optimization and engineering applications. *Sci. Rep.* **2025**, *15*, 11581.

12. Chen, X.; Yang, S.; Wang, T.; Chye, E.U.; Voronin, V.; Qin, H. Research on improved polar lights optimization algorithm. *Vestn. Pac. State Univ.* **2024**, *4*, 59–66.
13. Ahmed, M.; Kamel, S.H.; Abbasy, N.H.; Abouelseoud, Y. A Gaussian random walk salp swarm algorithm for optimal dynamic charging of electric vehicles. *Appl. Soft Comput.* **2023**, *147*, 110838.
14. Nassef, A.M.; Abdelkareem, M.A.; Maghrabie, H.M.; Baroutaji, A. The role of random walk-based techniques in enhancing metaheuristic optimization algorithms—A systematic review. *IEEE Access* **2024**, *12*, 139573–139608.
15. Kazikova, A.; Pluhacek, M.; Senkerik, R. Why tuning the control parameters of metaheuristic algorithms is so important for fair comparison? *Mendel* **2020**, *26*, 9–16.
16. Tatsis, V.A.; Ioannidis, D. Online cluster-based parameter control for metaheuristics. *arXiv* **2025**, arXiv:2504.05144.
17. Wei, K. Dynamic Parameter Tuning Method with Presets for Metaheuristics. Master's Thesis, University of Twente, Enschede, The Netherlands, 2022.
18. Nagalla, S. Dynamic pricing for electric vehicle charging: A real-time optimization framework for grid stability and economic efficiency. *Eur. J. Comput. Sci. Inf. Technol.* **2025**, *13*, 42–53.
19. Chen, L.; He, L.; Zhou, Y. An exponential cone programming approach for managing electric vehicle charging. *Oper. Res.* **2024**, *72*, 2215–2240.
20. Wang, Z.; Jochem, P.; Fichtner, W. A scenario-based stochastic optimization model for charging scheduling of electric vehicles under uncertainties. *J. Clean. Prod.* **2020**, *246*, 119886.
21. Bakar, M.H.A.; Rasid, M.M.; Nasir, S.; Hussin, S.M.; Sapari, N.; Ramli, A. Modeling of battery degradation for energy scheduling of a grid-connected DC microgrid. In *Proceedings of the IEEE 5th International Conference on Power Engineering Applications (ICPEA)*; IEEE: Shah Alam, Malaysia, 2025.
22. Le, C.N.; Vinayagam, A.; Tran, P.T.; Stojcevski, S.; Dinh, T.N.; Stojcevski, A.; Chandran, J. State of health aware adaptive scheduling of battery energy storage system charging and discharging in rural microgrids using long short-term memory and convolutional neural networks. *Energies* **2025**, *18*, 5641.
23. Gu, S.; Qian, K.; Yang, Y. Optimization of electric vehicle charging and discharging strategies considering battery health state: A safe reinforcement learning approach. *World Electr. Veh. J.* **2025**, *16*, 286.
24. Sayed, K.; Aref, M.; Almalki, M.M.; Mossa, M.A. Optimizing fast charging protocols for lithium-ion batteries using reinforcement learning. *Results Eng.* **2025**, *25*, 104302.
25. Zhao, Z.; Kong, S.; Bo, B.; Li, X.; Hao, L.; Xu, F.; Chen, L. Research on battery aging and user revenue of electric vehicles in V2G scenarios. *Electronics* **2025**, *14*, 4567.
26. Yuan, C.; Zhao, D.; Heidari, A.A.; Liu, L.; Chen, Y.; Chen, H. Polar lights optimizer: Algorithm and applications in image segmentation and feature selection. *Neurocomputing* **2024**, *607*, 128427.
27. Kegeleirs, M.; Ramos, D.G.; Birattari, M. Random walk exploration for swarm mapping. In *Towards Autonomous Robotic Systems*; Springer: Cham, Switzerland, 2019; pp. 235–246.
28. Gupta, S.; Deep, K. Random walk grey wolf optimizer for constrained engineering optimization problems. *Comput. Intell.* **2018**, *34*, 1025–1045.
29. Yaqoob, A.; Verma, N.K.; Aziz, R.M. Metaheuristic algorithms and their applications: A comprehensive review. In *Metaheuristics for Machine Learning*; Elsevier: Amsterdam, The Netherlands, 2024; pp. 1–35.
30. Xue, J.; Shen, B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. *J. Supercomput.* **2023**, *79*, 7305–7336.
31. Sahoo, S.K.; Saha, A.K.; Ezugwu, A.E.; Agushaka, J.O.; Abuhaija, B.; Alsoud, A.R.; Abualigah, L. Moth flame optimization: Theory, modifications, hybridizations and applications. *Arch. Comput. Methods Eng.* **2022**, *30*, 391–426.
32. Amiri, M.H.; Hashjin, N.M.; Montazeri, M.; Mirjalili, S.; Khodadadi, N. Hippopotamus optimization algorithm: A novel nature-inspired optimization algorithm. *Sci. Rep.* **2024**, *14*, 5032.
33. Zhao, S.; Zhang, T.; Ma, S.; Wang, M. Sea-horse optimizer. *Appl. Intell.* **2023**, *53*, 11833–11860.
34. Abualigah, L.; Abualigah, R.; Ikotun, A.M.; Zitar, R.A.; Alsoud, A.R.; Khodadadi, N.; Ezugwu, A.E.; Hanandeh, E.S.; Jia, H. Whale optimization algorithm: Analysis and survey. In *Metaheuristic Optimization Algorithms*; Morgan Kaufmann: Burlington, MA, USA, 2024; pp. 105–115.
35. Dehghani, M.; Montazeri, Z.; Trojovská, E.; Trojovský, P. Coati optimization algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems. *Knowl.-Based Syst.* **2023**, *259*, 110011.
36. Rahman, T.; Alharbi, T. Exploring lithium-ion battery degradation. *Batteries* **2024**, *10*, 220.

37. Zhang, J.; Wang, S.; Ono, K. Electrochemical impedance spectroscopy. In *Microscopy and Microanalysis for Lithium-Ion Batteries*; CRC Press: Boca Raton, FL, USA, 2023; pp. 301–350.
38. Gismero, A.; Nørregaard, K.; Johnsen, B.; Stenhøj, L.; Stroe, D.-I.; Schaltz, E. EV battery state-of-health estimation using incremental capacity analysis. *J. Energy Storage* **2023**, *64*, 107110.
39. Lee, J.; Won, J. Enhanced coulomb counting for SoC and SoH estimation. *IEEE Access* **2023**, *11*, 15449–15459.
40. Zhang, H.; Li, Y.; Zheng, S.; Lu, Z.; Gui, X.; Xu, W.; Bian, J. Battery lifetime prediction across diverse ageing conditions with inter-cell deep learning. *Nat. Mach. Intell.* **2025**, *7*, 270–277.
41. Zhang, H.; Gui, X.; Zheng, S.; Lu, Z.; Li, Y.; Bian, J. BatteryML: An open-source platform for machine learning on battery degradation. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024.
42. Anthony Jnr, B.; Petersen, S.A.; Ahlers, D.; Krogstie, J. API deployment for big data management towards sustainable energy prosumption in smart cities—A layered architecture perspective. *Int. J. Sustain. Energy* **2020**, *39*, 263–289.
43. Shirley, J.A.J.; Pooja, R.P.; Reddy, M.J.B. Metaheuristic algorithm-based energy management system for EV charging stations. *IEEE Access* **2024**, *12*, 116354–116367.

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.