

Article

A Manifold Alignment and Hierarchical Surrogate-Assisted Transfer Optimization Algorithm for Multi-UUV Shape Design

Junyu Xiang, Xinjing Wang *, Shengfa Wang, Guanghui Liu and Huachao Dong

School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China; xiangjunyu2022@mail.nwpu.edu.cn (J.X.); wangsf@mail.nwpu.edu.cn (S.W.); liuguanghui2023@mail.nwpu.edu.cn (G.L.); hdong@nwpu.edu.cn (H.D.)

* Correspondence: wangxinjing0213@nwpu.edu.cn

Abstract

In engineering practice, different requirements often give rise to distinct product designs. For the specific case of multi-UUVs, small-scale vehicles are typically designed with a rotational body shape to ensure superior hydrodynamic performance, whereas large-scale vehicles are often configured with a near-rectangular body shape to satisfy the demands of substantial payload capacity. These two tasks share a portion of common variables, while each also maintains its own task-specific variables. When each task is optimized independently, redundant computational efforts are incurred and inherent similarities among tasks remain unexploited, which frequently leads to suboptimal solutions. Typical multitask optimization algorithms assume completely heterogeneous tasks and therefore become inefficient when applied to this kind of partially heterogeneous problem. To address this, a manifold alignment and hierarchical surrogate-assisted transfer optimization algorithm (MAHSTO) is proposed in this work. In MAHSTO, an implicit knowledge transfer strategy is developed via manifold alignment. The design variables of both tasks are mapped onto a common low-dimensional latent space via manifold alignment, which enables implicit knowledge transfer across tasks. In addition, a hierarchical multisurrogate model with adaptive sampling is established. It comprises one shared global surrogate model that captures common trends across tasks and two task-specific surrogate models that focus on accurately fitting their respective tasks. Furthermore, an adaptive sampling criterion is adopted for different surrogate models to balance exploration and exploitation. Experiments on benchmark cases demonstrate that the proposed MAHSTO outperforms four state-of-the-art optimization algorithms, achieving the best performance in 58.3% of cases. Finally, MAHSTO is applied to the shape optimization of multi-UUVs. The results further verify its competitiveness in handling computationally expensive engineering problems.

Academic Editor: Zhiguang Feng

Received: 7 June 2026

Revised: 10 July 2026

Accepted: 16 July 2026

Published: 19 July 2026

Keywords: multitask optimization; surrogate-assisted optimization; unmanned underwater vehicle; manifold alignment; knowledge transfer

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Different performance requirements and application scenarios often lead to multiple variants of the same product family. Taking multi-UUVs as an example, small-scale multi-

UUVs focus primarily on hydrodynamic performance, so they are usually designed with a rotational body shape. Large-scale multi-UUVs, by contrast, emphasize payload capacity and tend to adopt a near-rectangular body shape. These two design tasks share most of the design variables, but each also retains its own task-specific variables. The shape optimization of these two configurations of multi-UUVs forms a partially heterogeneous multitask optimization problem. Traditional optimization methods usually optimize each design task independently. They fail to exploit knowledge shared between tasks. Each task has to start from scratch and go through iterative optimization on its own, which wastes computational resources [1,2].

Multitask optimization allows parallel optimization of related tasks through knowledge transfer [3–5]. When optimization tasks are correlated, search experience from one task can guide the optimization of others. This leads to faster convergence and better computational efficiency [6]. Lin et al. [7] proposed a multiobjective multitask evolutionary algorithm with decomposition-based transfer selection. Its transfer selection is guided by decomposition and only keeps solutions that bring clear improvement to their sub-problems. An adaptive multitask evolutionary algorithm based on knowledge reuse is designed by Cui et al. [8]. In this algorithm, the transferred solutions are linearly transformed according to the convergence interval of decision variables. These methods perform well on various benchmark cases and engineering applications. However, they typically assume that all tasks share the same decision space. For heterogeneous problems, directly transferring solutions often leads to negative transfer. Wu et al. [9] developed an orthogonal transfer method for tasks with different dimensions. They first mapped a source task's best solution into the target task's space via a cross-task mapping strategy. Then the dimensions of two individuals were blended using orthogonal experimental design. Han et al. [10] constructed a nonlinear mapping from source to target tasks using an adaptive kernel function. They aligned multiple source domains with weighted importance based on fitness landscape similarity, thereby achieving positive knowledge transfer across heterogeneous tasks.

Manifold alignment provides a typical way to align heterogeneous data. It takes datasets from different domains and projects them into a common low-dimensional latent space. Samples with similar semantics or functions end up close to each other in that space. Traditional alignment methods include unsupervised ones like locally linear embedding and Laplacian eigenmaps. There are also semi-supervised approaches that use supervised information from paired points [11,12]. In multitask optimization, some researchers have tried using manifold alignment for knowledge transfer across heterogeneous tasks. An autoencoder-based mapping method is developed by Xue et al. [13]. They map the search spaces of different tasks into a unified latent space, allowing genetic information to be exchanged there. He et al. [14] proposed a hybrid autoencoder method for heterogeneous multitask optimization. Their approach learns task-specific and shared features to build a general knowledge transfer channel. These methods offer new ideas for handling heterogeneous tasks. But most existing work focuses on fully heterogeneous tasks and does not handle partially heterogeneous ones well.

In this work, we propose a multitask optimization framework to handle these challenges. The framework consists of two strategies: implicit knowledge transfer via manifold alignment and adaptive sampling assisted by hierarchical surrogate models. The main contributions can be summarized as follows.

- (1) For partially heterogeneous design spaces, we develop an implicit knowledge transfer strategy via manifold alignment in this work. Both tasks are mapped onto a common low-dimensional latent space. Within this space, individuals from the two tasks can directly undergo evolutionary operations, thereby achieving effective knowledge transfer across tasks.

- (2) To reduce the computational cost, hierarchical surrogate models with adaptive sampling are constructed. We train the shared surrogate model using the common variables of both tasks to capture common trends across tasks. Two task-specific surrogate models are built to achieve precise predictions within the tasks, respectively. With the help of adaptive sampling criteria, efficient optimization can be achieved under a limited evaluation budget.
- (3) MAHSTO outperforms in most of the benchmark cases and has also been successfully applied to the shape optimization of multi-UUVs with rotational body shape and near-rectangular body shape. Experimental results show that the proposed algorithm can effectively solve the partially heterogeneous multitask optimization problems.

The remainder of this article is organized as follows. Section 2 covers the problem description and related works. The proposed MAHSTO is elaborated in Section 3. Section 4 validates the effectiveness of the algorithm on benchmark cases. The proposed algorithm is applied to solve the shape optimization of multi-UUVs in Section 5. Section 6 concludes the article and discusses future research directions.

2. Problem Description and Related Works

2.1. Mathematical Definition of Partially Heterogeneous Problems

This work focuses on the partially heterogeneous multitask optimization problems. In this kind of problem, the two tasks share a portion of common design variables while each possesses its own specific design variables. The dimensions and physical meanings of these specific variables differ between the two tasks.

Let the decision spaces of task T_A and task T_B be denoted as $\mathcal{X}_A \subseteq \mathbb{R}^{d_A}$ and $\mathcal{X}_B \subseteq \mathbb{R}^{d_B}$, respectively. They are expressed as Equation (1).

$$\mathcal{X}_A = \mathcal{X}_c \times \mathcal{X}_s^A, \quad \mathcal{X}_B = \mathcal{X}_c \times \mathcal{X}_s^B \quad (1)$$

where $\mathcal{X}_c \subseteq \mathbb{R}^{d_c}$ is the common variable space, $\mathcal{X}_s^A \subseteq \mathbb{R}^{d_s^A}$ and $\mathcal{X}_s^B \subseteq \mathbb{R}^{d_s^B}$ are the private variable spaces of task T_A and task T_B , respectively. The common variables x_c share the same value range and physical meaning in both tasks, while the dimensions of the private variables are different, that is, $d_s^A \neq d_s^B$.

The partially heterogeneous multitask optimization problem in this work is different from both traditional homogeneous multitask optimization problems (where $\mathcal{X}_A = \mathcal{X}_B$) and fully heterogeneous multitask optimization problems (where $\mathcal{X}_A \cap \mathcal{X}_B = \emptyset$). In this work, we aim to find the optimal solutions x_A^* and x_B^* for the two tasks under a limited budget of real evaluations. These solutions can be defined as:

$$x_A^* = \arg \min_{x_A \in \mathcal{X}_A} f_A(x_A), \quad x_B^* = \arg \min_{x_B \in \mathcal{X}_B} f_B(x_B) \quad (2)$$

where $f_A(\cdot)$ and $f_B(\cdot)$ are the corresponding objective functions of two tasks. In this work, both of them are set as black-box functions and the evaluation costs are very high.

In the field of multitask optimization, the reliable quantification of task similarity remains a challenging task, and the partially heterogeneous problems addressed in this work further increase this complexity. Therefore, a data-driven selective transfer strategy is adopted in this work, and the detailed similarity measurement method is described in Section 3.1.1.

2.2. Two Configurations of Multi-UUVs—A Novel Engineering Case

In nature, there are numerous phenomena where combined shapes are adopted for drag reduction and efficiency enhancement [15–18]. For example, a flock of wild geese often adopt a line-shaped formation during their long flights. In this way, the following geese can obtain lift support by leveraging the upward airflow generated by the preceding ones [19]. A similar strategy of drag reduction is also adopted by aquatic animals. A line-shaped formation allows ducks or migrating fishes to reduce the hydrodynamic resistance during long-distance swimming [20]. Research on drag reduction through the line-shaped formation of combined vehicles has been widely applied in aerospace, transportation, and ocean engineering [21–24]. For underwater vehicles, connecting multiple single-body UUVs end-to-end reduces the resistance and energy consumption during navigation. This configuration thus offers a novel solution for complex ocean missions.

In engineering practice, design requirements for equipment vary significantly with different scales. Taking aviation as an example, small-scale aircraft prioritize superior flight performance. Their fuselage design often adopts a circular or near-circular cross-section to minimize flight drag. Large-scale aircraft design, by contrast, places greater emphasis on a larger cargo capacity. Large transport aircraft typically employ a near-rectangular cross-section for the fuselage to accommodate more cargoes, such as the European transport aircraft A400M [25]. Similarly, with the increasingly complex ocean tasks and the dynamically changing requirements of UUVs, conventional UUVs with rotational body shapes have been unable to fully meet the diverse demands of complex underwater tasks. Hence, large-scale underwater vehicles are now shifting toward designs with high fullness and nonaxisymmetric shapes [26].

This work focuses on these two different configurations of multi-UUVs. Small-scale multi-UUVs with a rotational body shape are regarded as T_A . Large-scale multi-UUVs with a near-rectangular body shape are treated as T_B . As illustrated in Figure 1, although these configurations share the same overall design principles, their cross-sectional shapes are quite different. The rotational body shape of T_A can be described with just one variable, while the near-rectangular body shape of T_B requires multiple variables to characterize its shape. These two design tasks are consistent with the definition of partially heterogeneous multitask optimization provided above. In this work, the small-scale multi-UUVs with the rotational body shape and the large-scale multi-UUVs with the near-rectangular body shape are treated as two partially heterogeneous multitask optimization problems, and a multitask optimization is carried out on them in Section 5.

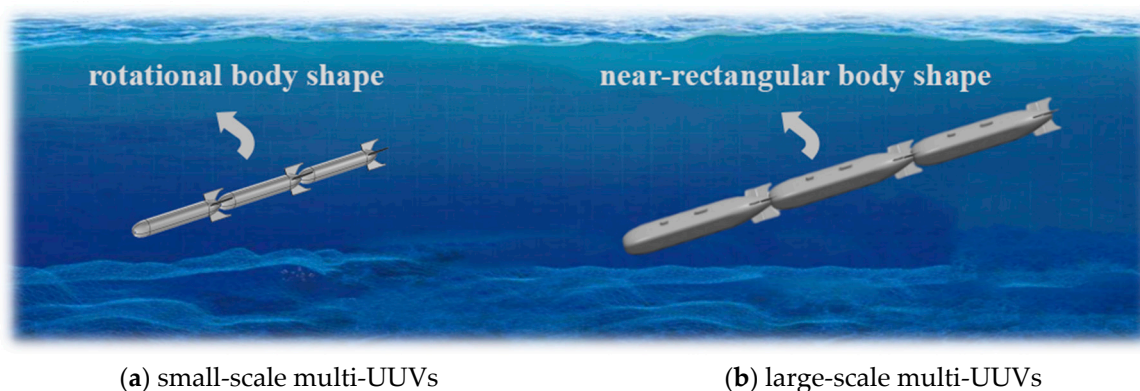


Figure 1. Line-shaped formation of underwater vehicles: (a) small-scale multi-UUVs, (b) large-scale multi-UUVs.

2.3. Surrogate Model

When confronted with complex engineering problems, the evaluation of unknown samples is often time-consuming and resource-intensive. With the rapid development of computer technology, surrogate modeling techniques have played a critical role in handling optimization problems that involve computationally expensive evaluations. The essence of a surrogate model lies in approximating the relationship between design variables and response values, thereby converting an originally implicit relationship into an explicit mathematical function. Commonly used surrogate models include polynomial response surface (PRS) [27], radial basis function (RBF) [28], support vector regression (SVR) [29], Kriging [30] and so on.

The PRS model is characterized by a simple functional form, but as the dimensionality of the problem increases, a larger number of sample points are required to determine its undetermined coefficients, making it unsuitable for high-dimensional problems. The RBF model is essentially an interpolative method, but when sample points are located too close to each other, the interpolation matrix may become ill-conditioned, thereby deteriorating the model accuracy. In comparison with other surrogate models, the Kriging model not only predicts the response values of unknown sample points but also provides an estimate of the prediction uncertainty [31]. Owing to its excellent performance in addressing nonlinear problems, the Kriging model has been widely adopted in various engineering applications.

The specific formulas of the Kriging model can be summarized as Equation (3).

$$\begin{aligned}\hat{f}(x) &= \hat{\mu} + \gamma^T(x)R^{-1}(y - 1\hat{\mu}) \\ \hat{s}^2(x) &= \hat{\sigma}^2[1 - \gamma^T(x)R^{-1}\gamma(x) + \frac{(1 - 1^T R^{-1}\gamma(x))^2}{1^T R^{-1}1}]\end{aligned}\quad (3)$$

where $\hat{f}(x)$ and $\hat{s}(x)$ denote the predicted value and the mean squared error (MSE), respectively. R is an $n \times n$ correlation matrix that reflects the correlation among the n known sample points. $R(x)$ is an n -dimensional vector that describes the correlation between the known sample points and the input point x . The 1 represents the regression matrix and $r(x)$ is the correlation variable. The parameters $\hat{\mu}$ and $\hat{\sigma}^2$ are estimated by maximum likelihood estimation.

It is worth noting that the mean squared error reflects the sample density in the design space. When the unknown point is close to the known sample points, the MSE is small, while otherwise it becomes large.

In this work, the Kriging model is utilized to construct the hierarchical multisurrogate model for Task A and Task B. Its mean squared error is also employed to select new sample points for exploration of the shared surrogate model. The detailed construction process is elaborated in Section 3.2.

3. Proposed Algorithm

To achieve efficient optimization for partially heterogeneous multitask problems, a manifold alignment and hierarchical surrogate-assisted transfer optimization algorithm is proposed in this work. This section provides a detailed introduction of the proposed algorithm.

The framework of MAHSTO is illustrated in Figure 2, consisting of two strategies: the implicit knowledge transfer via manifold alignment (highlighted in blue) and the hierarchical surrogate model construction (surrounded by dashed box).

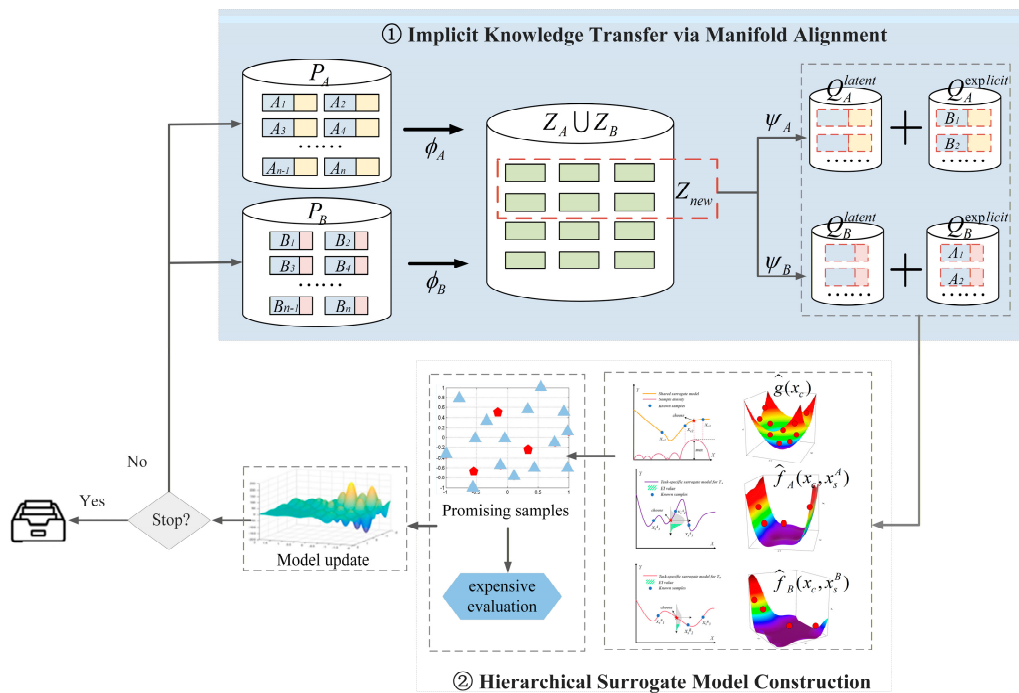


Figure 2. The framework of MAHSTO.

3.1. Implicit Knowledge Transfer via Manifold Alignment

The different dimensionalities and physical meanings of design variables make it difficult to conduct explicit knowledge transfer in the original spaces. Therefore, we propose an implicit knowledge transfer method based on manifold alignment as shown in Figure 3. It helps to map the decision spaces of the two tasks onto a common low-dimensional latent space $\mathcal{Z}$. The dimension of $\mathcal{Z}$ is lower than that of $\mathcal{T}_A$ and $\mathcal{T}_B$ ($D_Z < \min(D_A, D_B)$). In this latent space $\mathcal{Z}$, the samples from the two tasks are aligned based on their performance similarities. Thus, the co-evolution between the two tasks can be carried out in this latent space.

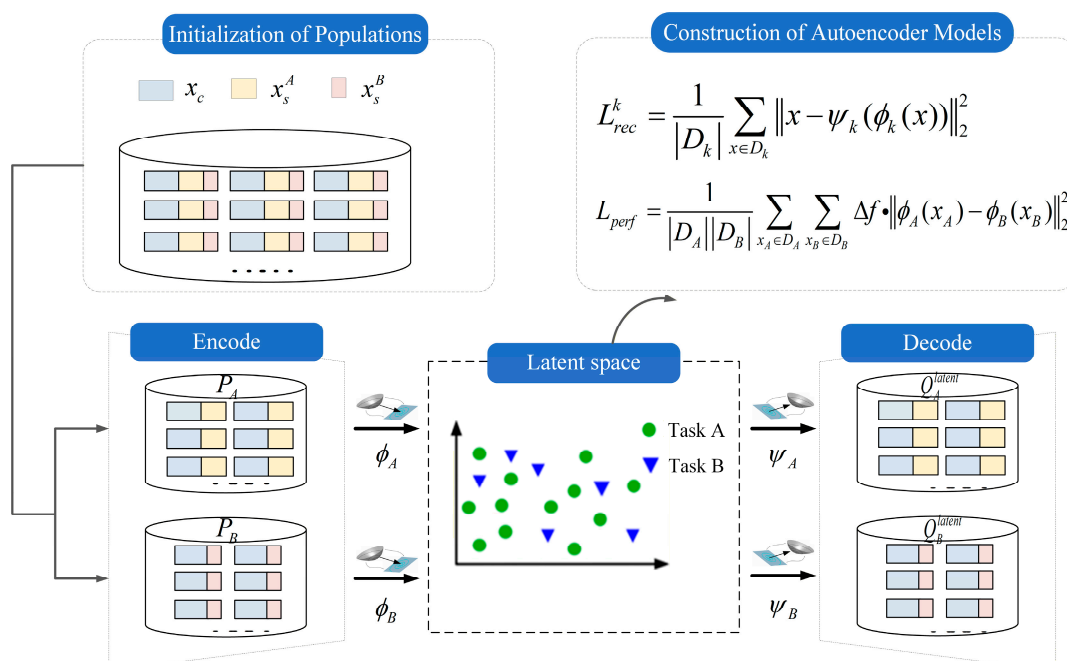


Figure 3. Illustration of implicit Knowledge Transfer via Manifold Alignment.

3.1.1. Mapping Model Construction

For each task $k \in \{A, B\}$, an autoencoder is constructed, comprising an encoder $\phi_k: \mathcal{X}_k \rightarrow \mathcal{Z}$ and a decoder $\psi_k: \mathcal{Z} \rightarrow \mathcal{X}_k$. The encoder ϕ_k maps the original design variables into a low-dimensional latent vector. The decoder ψ_k attempts to reconstruct the original variables from the latent vector. The autoencoder satisfies $\psi_k(\phi_k(x_k)) \approx x_k$. Both T_A and T_B share the same latent space, but their network architectures of the encoders and decoders are designed independently according to their different input dimensions.

The two tasks' autoencoders are designed as fully connected neural networks. The hidden layers employ the ReLU activation function to capture the nonlinear mapping relationship between the complex design variables in the original decision space and the objective function. The output layer of the latent space does not apply an activation function and directly outputs real-valued latent vectors to ensure that the latent space has a complete real-valued continuous domain. The two tasks' autoencoders together consist of four subnetworks and a total of 12 fully connected layers. The decoders of both tasks are symmetrical to their respective encoders.

Specifically, taking task T_A as an example, the encoder ϕ_A starts from the 21-dimensional input layer. It first passes through a fully connected layer with 15 neurons, followed by ReLU activation, then passes through another fully connected layer with 10 neurons and ReLU activation, and finally compresses to the latent space.

As for the training of the autoencoder, the Adam optimizer is adopted. The initial learning rate is 0.001. The decay coefficient for the first-order moment estimation β_1 is 0.9, and the decay coefficient for the second-order moment estimation β_2 is 0.999. The numerical stability constant ϵ is 10^{-8} . The batch size is 64 and the number of training epochs is 200. The autoencoders are trained using the currently evaluated sample sets D_A and D_B . For mini-batch gradient descent, the computational complexity of a single batch's forward and backward propagation is $O(\sum_{l=1}^L n_{l-1}n_l)$, where L represents the number of fully connected layers, and n_l is the number of neurons in layer l . The total computational time of autoencoder training accounts for less than 1% of the entire optimization process, far lower than the cost of high-fidelity CFD simulations. The optimization objectives are the reconstruction loss and the performance alignment loss.

The reconstruction loss ensures that the autoencoder for each task can accurately reconstruct the original design variables. The reconstruction loss can be expressed as:

$$L_{rec}^k = \frac{1}{|D_k|} \sum_{x \in D_k} \|x - \psi_k(\phi_k(x))\|_2^2 \quad (4)$$

where $|D_k|$ denotes the number of samples in task T_k , $\|\cdot\|_2$ is the Euclidean norm.

The reconstruction loss L_{rec}^k serves as a regularizer against the degenerate case where all samples are mapped to the similar low-dimensional manifold. If the encoder ϕ_k produces a degenerate latent space representation where several latent vectors converge, it is difficult for the decoder ψ_k to reconstruct diverse original inputs from these latent vectors. The reconstruction loss will significantly increase. Therefore, the degenerate mapping will be effectively excluded by the reconstruction loss L_{rec}^k during the optimization process.

The performance alignment loss forces samples from the two tasks that have similar objective values to be mapped to close representations in the latent space $\mathcal{Z}$. For any pair

of samples $(x_A, x_B) \in D_A \times D_B$, the difference of objective values is defined as $\Delta f = |f_A(x_A) - f_B(x_B)|$. The performance alignment loss is then given by Equation (5).

$$L_{perf} = \frac{1}{|D_A| |D_B|} \sum_{x_A \in D_A} \sum_{x_B \in D_B} \Delta f \cdot \|\phi_A(x_A) - \phi_B(x_B)\|_2^2 \quad (5)$$

The performance alignment loss L_{perf} uses the objective value difference Δf as the weighting coefficient for the distance of the latent vectors. It encourages samples with close objective values to be positioned near each other in the latent space, while allowing larger distances for those with significantly different objectives, thereby making the geometry of the latent space reflect performance similarity. This design essentially implements a data-driven selective transfer strategy. The effectiveness of transfer does not depend on prior assumptions of task similarity but is dynamically determined by the actual objective differences between samples.

3.1.2. Co-Evolution in the Latent Space

In each generation of the optimization, the following steps are performed:

- (1) Encoding: Each individual in the current populations P_A and P_B is mapped to the latent space via the respective encoders, thereby obtaining the latent vector sets $Z_A = \{\phi_A(x) : x \in P_A\}$ and $Z_B = \{\phi_B(x) : x \in P_B\}$.
- (2) Evolution in latent space: The latent vector sets Z_A and Z_B are merged into a joint population Z_S . Parent latent vectors are randomly selected from Z_S and subjected to crossover and mutation operations. Simulated binary crossover and polynomial mutation are employed. The generated new latent vectors are denoted as Z_{new} .
- (3) Decoding: Each latent vector in Z_{new} is decoded by both decoders ψ_A and ψ_B to produce candidate populations Q_A for task T_A and Q_B for task T_B . These candidate individuals embed features from both tasks in the latent space, realizing cross-task knowledge transfer.
- (4) Explicit transfer supplement: To enhance the stability of knowledge transfer, an explicit transfer is performed each iteration. Specifically, the top individuals from each task of T_A and T_B are selected, and their common-variable parts are directly injected into the population of the other task, replacing the worst individuals. When injecting, the specific-variable parts of the new individual are taken from the individual in the receiving population that is most similar to the injected common variables.
- (5) Population update: The candidate sets Q_A and Q_B are merged with the current populations P_A and P_B , respectively. The best individuals are retained as the next-generation populations.

3.2. Hierarchical Multisurrogate Model with Adaptive Sampling

3.2.1. Hierarchical Multisurrogate Model Construction

For expensive multitask optimization, due to the high cost, the number of real evaluations is strictly limited. Therefore, it is necessary to construct surrogate models to predict the objective values of unevaluated individuals, thereby guiding the sampling direction and assisting the optimization process. However, since the two tasks share a portion of common variables and have their private variables as well, it is difficult to fully utilize the multitask knowledge by simply constructing a single surrogate model. To address this issue, this work proposes a hierarchical multisurrogate model construction strategy, which includes a shared surrogate model and two task-specific surrogate models.

Specifically, the shared surrogate model $\hat{g}(x_c)$ uses only the common variables x_c as input. It is trained on the common variables and corresponding objective values of all evaluated samples from both tasks. By making use of the common knowledge shared across the two tasks, this model attempts to capture how the common variables x_c influence performance.

Different from the shared surrogate model $\hat{g}(x_c)$, the two task-specific surrogate models $\hat{f}_A(x_c, x_s^A)$ and $\hat{f}_B(x_c, x_s^B)$ use the whole decision variables of each task as input. They are trained independently using the evaluated sample sets D_A and D_B . They only focus on the relationships between the variables of their own tasks and the objective values, thereby achieving the accurate fitting for their respective tasks.

The evaluated samples sets D_A and D_B serve as the initial training data of the three surrogate models. Subsequently, after applying the adaptive sampling strategy to supplement new samples, these three models will be continuously updated.

The computational complexity of Kriging models primarily arises from the construction and inversion of the covariance matrix. With n training samples of dimension d , covariance matrix construction costs $O(n^2d)$, and matrix inversion costs $O(n^3)$. Prediction for a single new sample costs $O(n^2)$. On a standard CPU, Kriging model training time grows from subseconds to several seconds as samples increase, far below the cost of a single CFD simulation.

3.2.2. Adaptive Sampling of the Multisurrogate Models

For the surrogate-assisted optimization algorithm, the sampling criterion needs to balance the exploitation and exploration to achieve efficient global optimization. The exploitation requires adding new samples in the area where the current model predicts the best to accelerate convergence, while the exploration tends to add new samples in the regions with high predictive uncertainty to improve the global model accuracy. The trade-off between them becomes more complex for the partially heterogeneous multitask optimization problems.

In this work, the shared surrogate model $\hat{g}(x_c)$ is constructed on the common variables of both T_A and T_B . The predictive uncertainty $\sigma_{share}(x_c)$ reflects the information that both tasks lack in the common variable space. In the shared model, higher predictive uncertainty $\sigma_{share}(x_c)$ implies that the samples of the two tasks are sparser in this area. Sampling in such regions can improve the accuracy of both task-specific surrogate models simultaneously.

The task-specific surrogate models $\hat{f}_A(x_c, x_s^A)$ and $\hat{f}_B(x_c, x_s^B)$ focus on accurate fitting within each task. The expected improvement $EI_k(x)$ measures the potential improvement over the current best solution of the task. We select $EI(x)$ as the sampling criterion to push the search towards regions that are likely to enhance the performance of each task.

Our evaluation set consists of three samples. One sample comes from maximizing the predictive uncertainty of the shared surrogate model $\hat{g}(x_c)$. Other samples are selected by maximizing the expected improvement of the task-specific surrogate models $\hat{f}_A(x_c, x_s^A)$ and $\hat{f}_B(x_c, x_s^B)$ for each task. The first sample drives cross-task exploration and the other two assist in the exploitation within each task. This approach ensures that at least one candidate individual is selected for each task.

For each sampling round, the algorithm first collects all candidate individuals produced by Strategy 1. These candidate individuals include: the new individual candidate sets $\mathcal{Q}_A^{\text{latent}}$ and $\mathcal{Q}_B^{\text{latent}}$ obtained through the evolution in the latent space, and the transferred individual sets $\mathcal{Q}_A^{\text{explicit}}$ and $\mathcal{Q}_B^{\text{explicit}}$ obtained through the explicit transfer. These two kinds of candidate individuals are merged to form the complete candidate sets $C_A = \mathcal{Q}_A^{\text{latent}} \cup \mathcal{Q}_A^{\text{explicit}}$ for task T_A and $C_B = \mathcal{Q}_B^{\text{latent}} \cup \mathcal{Q}_B^{\text{explicit}}$ for task T_B .

Subsequently, the following sampling operations are performed:

- (1) Exploration sampling: For all candidate individuals from both task T_A and task T_B , the shared surrogate model $\hat{g}(x_c)$ is invoked to compute the predictive uncertainty $\sigma_{\text{share}}(x_c)$ of their common-variable part. The individual with the largest predictive uncertainty is selected as x^{share} , serving as the exploration sampling point.
- (2) Exploitation sampling for task T_A : For the candidate set C_A of task T_A , the task-specific surrogate model $\hat{f}_A(x_c, x_s^A)$ is invoked to compute the expected improvement $EI_A(x)$ for each individual. The individual with the largest expected improvement is selected as x^A , serving as the exploitation sampling point for task T_A .
- (3) Exploitation sampling for task T_B : For the candidate set C_B of task T_B , the task-specific surrogate model $\hat{f}_B(x_c, x_s^B)$ is invoked to compute the expected improvement $EI_B(x)$ for each individual. The individual with the largest expected improvement is selected as x^B , serving as the exploitation sampling point for task T_B .

The three new samples are then merged to form the evaluation set $S = \{x^{\text{share}}, x^A, x^B\}$. This set contains three individuals, corresponding to cross-task exploration, task T_A exploitation, and task T_B exploitation, respectively. For one candidate individual, EI computation requires one prediction from the task-specific surrogate model $O(n^2)$, and uncertainty computation requires one prediction from the shared surrogate model $O(n^2)$. The computational cost is far lower than the training cost of the Kriging models themselves.

Once the evaluation set S is determined, real evaluation is carried out on each individual in S to obtain its objective value. After evaluation, the new samples are added to the sample sets D_A and D_B of the corresponding tasks, and the shared surrogate model $\hat{g}(x_c)$ along with the corresponding task-specific surrogate model $\hat{f}_A(x_c, x_s^A)$ and $\hat{f}_B(x_c, x_s^B)$ are retrained using the updated sample sets. The above process is iterated until the preset budget of real evaluations is exhausted. The illustration of the adaptive sampling is shown in Figure 4. The corresponding pseudocode is summarized in Algorithm 1.

Algorithm 1: Hierarchical Surrogate Model with Adaptive Sampling

Input: Task-specific surrogate models $\hat{f}_A(x_c, x_s^A)$, $\hat{f}_B(x_c, x_s^B)$, shared surrogate model $\hat{g}(x_c)$, candidate sets C_A , C_B

/*Initialization*/

1. Evaluation set $S \leftarrow \emptyset$;

/*Main loop*/

While $NFEs \leq NFE_{\max}$ **do**

2. **Cross-task exploration sampling:**

3. Merge candidate sets $C \leftarrow C_A \cup C_B$;

4. For each $x \in C$, compute the shared model predictive standard deviation $\sigma_{share}(x_c)$;
 5. Select $x^{share} \leftarrow \arg \max_{x \in C} \sigma_{share}(x_c)$;
 6. $S \leftarrow S \cup \{x^{share}\}$;
 7. **Task A exploitation sampling:**
 8. For each $x \in C_A$, compute the expected improvement $EI_A(x)$;
 9. Select $x^A \leftarrow \arg \max_{x \in C_A} EI_A(x)$;
 10. $S \leftarrow S \cup \{x^A\}$;
 11. **Task B exploitation sampling:**
 12. For each $x \in C_B$, compute the expected improvement $EI_B(x)$;
 13. Select $x^B \leftarrow \arg \max_{x \in C_B} EI_B(x)$;
 14. $S \leftarrow S \cup \{x^B\}$;
 15. **Perform real evaluation** for each individual in S to obtain objective values;
 16. **Update** sample sets D_A, D_B ;
 17. **Retrain** $\hat{g}(x_c), \hat{f}_A(x_c, x_s^A), \hat{f}_B(x_c, x_s^B)$;
 18. **Return:** Newly evaluated sample set S
- End**

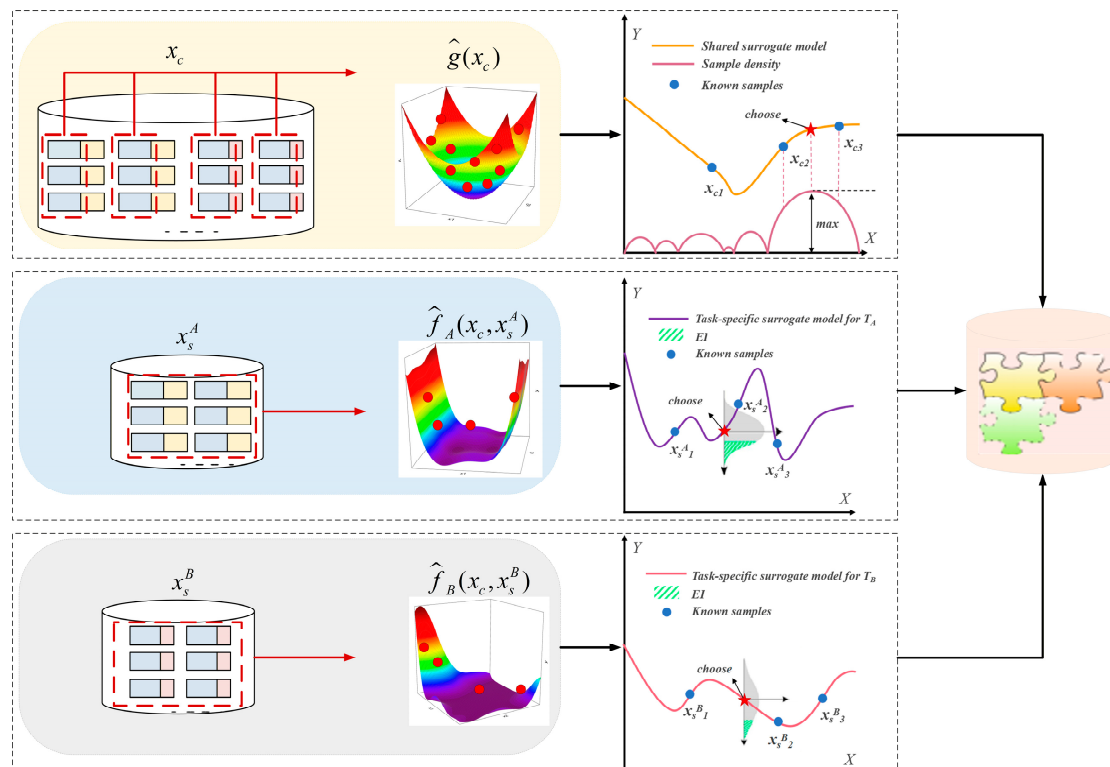


Figure 4. Hierarchical Multisurrogate Model with Adaptive Sampling.

4. Experimental Studies

In this section, we conduct a set of experiments to assess the effectiveness of the proposed MAHSTO. Existing benchmark suites do not fit the partially heterogeneous multi-task optimization problems. Therefore, we adapt several classical test functions into customized variants that better align with the partially heterogeneous optimization problem

following Jin et al. [32]. The proposed MAHSTO is compared against four state-of-the-art algorithms on the constructed benchmark cases.

4.1. Benchmark Cases

The benchmark functions used in this work have been widely employed in previous optimization literature [33,34] for algorithm validation. To intuitively demonstrate the performance gains achieved by the proposed approach on partially heterogeneous multi-task problems, the benchmark cases adopted in [35] are appropriately modified. The modified benchmark cases retain the characteristics of the original functions while incorporating features that reflect the complexity of actual engineering. Equation (6) gives the error term.

$$e_r(x, \phi) = \sum_{i=1}^d a(x_i, \phi) \cos(w(\phi)x_i + b(\phi) + \pi) \quad (6)$$

where a is the maximum error, w represents the number of local optima, and the deviation between the global optimum and the true global optimum at a given fidelity level is influenced by $b(\phi)$.

Specifically, six benchmark cases are adopted to verify the performance and advantages of MAHSTO. The detailed information of these cases is provided in Table 1.

Table 1. Detailed information of benchmark cases.

Function	Range	Function
Sphere	$[-5.12, 5.12]^n$	$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$
Trid	$[-n^2, n^2]^n$	$f(\mathbf{x}) = \sum_{i=1}^n (x_i - 1)^2 - \sum_{i=2}^n x_i x_{i-1}$
ROSENBROCK	$[-2, 2]^n$	$f(\mathbf{x}) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$
Zakharov	$[-5, 10]^n$	$f(\mathbf{x}) = \sum_{i=1}^n x_i^2 + (\sum_{i=1}^n 0.5ix_i)^2 + (\sum_{i=1}^n 0.5ix_i)^4$
GW	$[-10, 10]^n$	$f(\mathbf{x}) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
RS	$[-5.12, 5.12]^n$	$f(\mathbf{x}) = 20 + \sum_{i=1}^2 (x_i^2 - 10 \cos(2\pi x_i))$

To evaluate the performance in solving task pairs with varying dimensions and similarities, the test cases are further adapted in this work. The specific settings are presented in Table 2. These task pairs are collectively designed to serve as benchmark cases for the proposed transfer optimization framework.

Table 2. Revised task pairs of benchmark cases.

Function	Task Pairs	Dimension	Similarity Level
Case_1	Sphere_T1	$D = 5$	$\phi = 2000$
	Sphere_T2	$D = 8$	$\phi = 5000$
Case_2	Trid_T1	$D = 6$	$\phi = 2000$
	Trid_T2	$D = 10$	$\phi = 8000$
Case_3	ROSENBROCK_T1	$D = 10$	$\phi = 2000$

	ROSENBROCK_T2	$D = 20$	$\phi = 5000$
Case_4	Zakharov_T1	$D = 10$	$\phi = 2000$
	Zakharov_T2	$D = 20$	$\phi = 5000$
Case_5	GW_T1	$D = 15$	$\phi = 2000$
	GW_T2	$D = 20$	$\phi = 5000$
Case_6	RS_T1	$D = 15$	$\phi = 2000$
	RS_T2	$D = 20$	$\phi = 8000$

4.2. Ablation Experiments

To clearly evaluate the individual contributions of each component in the proposed algorithm, we designed four ablation variants corresponding to the key components of the core strategies:

MAHSTO_I: removing the autoencoders and the manifold alignment mechanism. The two tasks are optimized independently in their original decision spaces without cross-task knowledge transfer.

MAHSTO_II: retaining only the implicit knowledge transfer in the latent space and removing the explicit transfer mechanism in strategy one.

MAHSTO_III: canceling the construction of the shared surrogate model $\hat{g}(x_c)$. Adaptive sampling selects points solely based on the expected improvement of each task-specific surrogate model.

MAHSTO_IV: ignoring the construction of the two task-specific surrogate models. Add promising samples only based on the predictive uncertainty of the shared surrogate model $\hat{g}(x_c)$.

All ablation variants stay the same as the complete MAHSTO except for the above deleted component. Eight cases with different dimensions and similarities are picked out for the ablation experiment. To eliminate randomness, 30 independent runs are conducted for each case. The experimental results are analyzed using the Wilcoxon rank-sum test, with the significance level set to 0.05. The statistical results are shown in Table 3, and the bold font indicates the best result.

Table 3. Best values on the ablation experiment.

Cases	Tasks	MAHSTO_I	MAHSTO_II	MAHSTO_III	MAHSTO_IV	MAHSTO
Case_2	Task 1	-3.9784e+1 (1.73e+1) –	-4.6842e+1 (7.81e+0) =	-4.6183e+1 (8.29e+0) =	-4.5961e+1 (9.17e+0) =	-4.6719e+1 (8.32e+0)
	Task 2	-1.9176e+2 (8.94e+1) =	-2.0538e+2 (6.73e+1) =	-2.0491e+2 (7.26e+1) =	-2.0514e+2 (7.12e+1) =	-2.0716e+2 (6.24e+1)
Case_4	Task 1	9.3615e-1 (4.97e-1) –	6.9173e-1 (2.61e-1) =	6.9014e-1 (2.97e-1) =	8.7623e-1 (4.69e-1) –	6.8952e-1 (2.47e-1)
	Task 2	7.9932e-1 (6.52e-1) –	4.0921e-1 (3.12e-1) =	6.2835e-1 (4.89e-1) –	6.9752e-1 (5.14e-1) –	4.1367e-1 (2.85e-1)
Case_5	Task 1	1.0357e+0 (7.84e-1) –	8.9716e-1 (6.34e-1) –	9.5492e-1 (7.17e-1) –	8.4116e-1 (5.82e-1) –	7.3160e-1 (3.91e-1)
	Task 2	9.8142e-1 (5.27e-1) –	7.9473e-1 (4.31e-1) –	7.2263e-1 (4.05e-1) –	8.1024e-1 (4.81e-1) –	5.7849e-1 (7.34e-2)
Case_6	Task 1	1.2657e+0 (6.13e-1) –	9.5244e-1 (4.76e-1) –	7.9532e-1 (3.24e-1) –	9.1124e-1 (3.89e-1) –	6.8337e-1 (1.82e-1)
	Task 2	7.5921e-1 (5.84e-1) –	4.9634e-1 (3.78e-1) –	1.3179e-1 (8.47e-2) –	3.8534e-1 (2.91e-1) –	4.0726e-2 (3.67e-2)
+/-/=		0/7/1	0/4/4	0/5/3	0/6/2	Base

The statistical results of the ablation experiments show that MAHSTO outperforms the four variants, verifying the necessity of each component. MAHSTO_I achieves less satisfactory optimization results in the majority of cases. Without the autoencoder-based mapping that aligns the two tasks in a shared latent space, each task is forced to explore its own decision space from scratch. The intrinsic correlations in the common variables, which could otherwise accelerate the search through knowledge transfer, remain entirely untapped. MAHSTO_II shows a noticeable performance drop relative to the complete MAHSTO on test cases where the two tasks share relatively high similarity. When tasks are inherently more similar, the direct exchanges tend to carry more relevant and trustworthy information, making the absence of explicit transfer particularly damaging. It can be found that MAHSTO_III performs poorly on Case_5. With the shared surrogate model $\hat{g}(x_c)$ removed, the algorithm relies solely on the task-specific surrogates to guide sampling through expected improvement, which makes it prone to getting stuck in local optima. MAHSTO_IV suffers a severe performance drop compared to the complete MAHSTO. After the task-specific surrogates are removed, adaptive sampling is driven entirely by the uncertainty predictive from the shared surrogate model. While this encourages exploration of the common variable space, it provides no guidance for navigating the private variable regions that are unique to each task.

4.3. Comparative Tests

In this section, a comprehensive comparative test is conducted among the proposed MAHSTO and four classic optimization algorithms on a suite of benchmark cases specifically designed for partially heterogeneous multitask optimization problems. The four compared algorithms are summarized as follows.

OTMTO [9] is characterized by an explicit cross-task mapping strategy, where the global optimal solution of one task is mapped into the search space of another task via an orthogonal transfer method. Different from the implicit transfer adopted in the MAHSTO, an explicit transfer is employed to handle heterogeneous multitask problems. HAM [36] builds a hybrid ensemble of three surrogate models and adaptively switches its sampling criterion between exploration and exploitation. Similar to the proposed method, multiple surrogate models and an adaptive infill strategy are used to reduce the number of expensive function evaluations. MTDE-ADKT [37] is designed with an explicit dual-transfer framework, in which the intensity of knowledge transfer is adjusted according to an estimated similarity between tasks. MOMFEA-SADE [38] introduces a subspace alignment strategy based on principal component analysis. The source and target task data are first projected into their respective low-dimensional subspaces, after which a linear transformation is learned to minimize the distribution discrepancy between the two tasks. Consequently, through comparative tests against the above algorithms, the effectiveness of the proposed strategies can be comprehensively validated.

The termination criterion is $NFEs > NFEmax$, where NFE is the number of evaluations. In this work, $NFEmax$ is set to 500. To reduce the influence of randomness on the outcomes, each test case is independently run 30 times. For statistical comparison, all experimental results are analyzed using the Wilcoxon rank-sum test, with the significance level set to 0.05. Where statistically significant differences are identified, “+” and “-” indicate whether the proposed algorithm outperforms or underperforms the comparison algorithm; “=” denotes that the performance difference is not statistically significant. The average performance metrics obtained by each algorithm are presented in Table 4.

Table 4. Best values on the mathematical cases.

Cases	Task Pairs	OTMTO	HAM	MTDE-ADKT	MOMFEA-SADE	MAHSTO
Case_1	Task 1	2.1674e-2 (5.96e-3) +	5.8729e-2 (2.31e-2) −	4.1805e-2 (7.63e-3) −	3.6291e-2 (8.52e-3) −	2.7185e-2 (8.14e-3)
	Task 2	1.9210e-2 (6.72e-3) =	4.6183e-2 (1.86e-2) −	3.0174e-2 (5.81e-3) −	1.8972e-2 (6.10e-3) =	1.8723e-2 (5.93e-3)
Case_2	Task 1	−4.6193e+1 (1.79e+1) =	−3.7410e+1 (1.98e+1) −	−4.6589e+1 (8.42e+0) =	−4.8522e+1 (9.21e+0) =	−4.6719e+1 (8.32e+0)
	Task 2	−2.0592e+2 (8.13e+1) =	−1.8351e+2 (9.28e+1) =	−2.0318e+2 (8.04e+1) =	−2.0894e+2 (6.73e+1) =	−2.0716e+2 (6.24e+1)
Case_3	Task 1	1.5894e+0 (3.81e-1) −	1.2948e+0 (6.39e-1) −	9.2576e-1 (2.58e-1) −	1.0892e+0 (3.22e-1) −	8.4157e-1 (2.86e-2)
	Task 2	1.0472e+0 (4.37e-1) −	7.6821e-1 (5.92e-1) −	4.8092e-1 (3.13e-1) −	4.6938e-1 (5.87e-2) −	3.7831e-1 (2.51e-2)
Case_4	Task 1	8.7952e-1 (4.91e-1) −	1.0243e+0 (5.26e-1) −	6.4531e-1 (3.84e-1) =	6.8746e-1 (2.79e-1) =	6.8952e-1 (2.47e-1)
	Task 2	5.1521e-1 (2.96e-1) =	7.5862e-1 (3.71e-1) −	4.0794e-1 (2.93e-1) =	4.0392e-1 (2.14e-1) =	4.1367e-1 (2.85e-1)
Case_5	Task 1	9.5163e-1 (6.83e-1) −	8.2632e-1 (5.92e-1) −	1.5820e+0 (9.28e-1) −	1.4621e+0 (7.51e-1) −	7.3160e-1 (3.91e-1)
	Task 2	7.8310e-1 (4.29e-1) −	5.7921e-1 (7.62e-1) =	1.0326e+0 (5.26e-1) −	8.3957e-1 (4.83e-1) −	5.7849e-1 (7.34e-2)
Case_6	Task 1	8.9417e-1 (3.06e-1) −	1.5728e+0 (6.13e-1) −	8.4293e-1 (3.74e-1) −	1.8943e+0 (7.85e-1) −	6.8337e-1 (1.82e-1)
	Task 2	6.0248e-1 (3.16e-1) −	1.3291e+0 (5.72e-1) −	5.8719e-1 (2.06e-1) −	1.0421e+0 (5.68e-1) −	4.0726e-2 (3.67e-2)
+/-/=		1/7/4	0/10/2	0/8/4	0/7/5	Base

MAHSTO achieves competitive results on the majority of benchmark cases. It ranks first on Case_3, Case_5 and Case_6. Its results on Case_1 and Case_4 are very close to the best. OTMTO shows strong performance on problems with moderate dimensionality differences, for instance, on Case_1. When the disparity in variable dimensions becomes large as seen on Case_3 and Case_4, it starts to struggle. HAM is well suited to tasks that require strong local exploitation, such as Case_5. Due to the reliance on explicit similarity estimation, we find it difficult for MTDE-ADKT to obtain the optimal solution of problems like Case_3 and Case_5. As for MOMFEA-SADE, it achieves satisfactory results on Case_2 and Case_4, where the inter-task correlation is approximately linear. This algorithm seems unable to handle more complex nonlinear relationships, as its performance drops on Case_5 and Case_6.

The analysis of the above mathematical cases suggests that the proposed implicit knowledge transfer via manifold alignment strategy helps to facilitate knowledge transfer across partially heterogeneous tasks and reduce the risk of negative transfer. We also find that the construction of a hierarchical surrogate model with adaptive sampling effectively lowers computational cost without sacrificing accuracy. These results demonstrate the advantages of MAHSTO in addressing such partially heterogeneous problems.

5. Engineering Optimization Application on Multi-UUVs

To assess its effectiveness in engineering applications, the MAHSTO is applied to the shape optimization of the multi-UUVs. The hydrodynamic shape design is a critical aspect of multi-UUVs, as it not only determines the hydrodynamic performance but also directly affects its payload capacity. In this work, the multi-UUVs with the rotational body shape

and the multi-UUVs with the near-rectangular body shape are treated as two partially heterogeneous multitask optimization problems, and a multitask optimization is performed on them. For the shape design of multi-UUVs, the optimization objective is to minimize the total resistance during underwater navigation. The navigation resistance is evaluated through high-fidelity computational fluid dynamics simulations, which are computationally expensive and time-consuming. This makes the shape design of multi-UUVs an expensive black-box problem. For both tasks, the optimization constraints are defined as no volume interference between the connected vehicles. Figure 5 shows the detailed information of the shape design parameters of the two kinds of multi-UUVs.

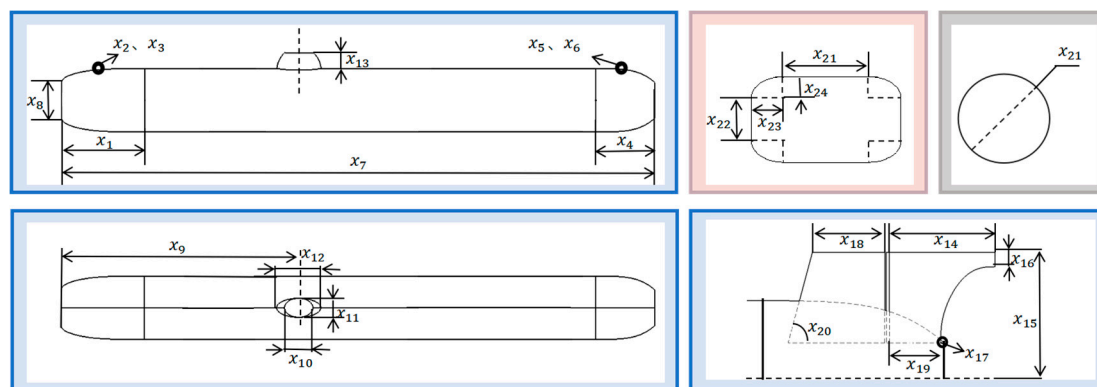


Figure 5. Parameter diagram of UUV.

As shown in Figure 5, the parameters enclosed in the blue box are defined as the shared parameters x_c between task T_A and task T_B . Those in the gray box are the task-specific variables of task T_A , which corresponds to the multi-UUVs with the rotational body shape. Similarly, the pink box contains the task-specific variables of task T_B , corresponding to the multi-UUVs with the near-rectangular body shape. Table 5 summarizes the geometric meanings of each shape design parameter.

Table 5. Detailed description of shape parameters.

Parameter	x_1	x_2, x_3	x_4	x_5, x_6	x_7
Meaning	length of head	adjustable parameters of the head curve	length of tail	adjustable parameters of the tail curve	length of UUV
Parameter	x_8	x_9	x_{10}	x_{11}	x_{12}
Meaning	diameter of two ends	distance from head to the protrusion	major axis of the upper surface of protrusion	minor axis of protrusion	major axis of the lower surface of protrusion
Parameter	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}
Meaning	height of protrusion	length of rudder	height of rudder	height of rudder end	distance from rudder to the end of body
Parameter	x_{18}	x_{19}	x_{20}		
Meaning	length of fin	width of rudder lower end	sweep forward angle of fin		
Parameter	x_{21}	x_{22}	x_{23}	x_{24}	x_{21}
Meaning	longer side of the near-rectangle	shorter side of the near-rectangle	major axis of the near-rectangle's corner	minor axis of the near-rectangle's corner	diameter of UUV

A more accurate optimization formula for the multi-UUVs shape design can be expressed as:

$$\begin{aligned} \min \quad & \{f_A(x_c, x_s^A), f_B(x_c, x_s^B)\} \\ \text{s.t.} \quad & \begin{cases} \varphi > 0.85 \\ x_A \in [lb_i, ub_i], i = 1, 2, \dots, 21 \\ x_B \in [lb_j, ub_j], j = 1, 2, \dots, 24 \end{cases} \end{aligned} \quad (7)$$

where f_x is the drag force of the multi-UUV, φ is the fullness of vehicles, lb and ub denote the lower and upper bounds of the decision space.

To accurately evaluate the hydrodynamic performance of the multi-UUVs, an automatic simulation and analysis framework is established in this work. The shape parametric modeling, mesh generation and CFD analysis are all integrated in this framework. On the basis of model accuracy verification and to eliminate the influence of mesh resolution on simulation results, mesh independence verification is conducted for the two tasks. It is finally determined that the mesh number is set to approximately 2,350,000 for the multi-UUVs. The mesh convergence tests confirm the reasonableness of the selected mesh quantities. Both mesh settings satisfy the engineering requirements for subsequent analysis.

Regarding the boundary condition setup for the fluid simulation, the front and bottom surfaces are defined as velocity inlets with an inflow velocity of 2 knots. The top and rear surfaces serve as pressure outlets, where we set the static pressure to 0 Pa. To take advantage of geometric symmetry and reduce computational cost, we impose a symmetry plane boundary condition on the XY-plane. The surface of the multi-UUVs is treated as a no-slip wall, and the wall boundaries are assigned as specified-shear walls.

The CFD solver and setup employed in this work have been systematically validated on a benchmark case. The SUBOFF standard model (an internationally recognized benchmark for underwater vehicle hydrodynamics) is used for validation. Its hydrodynamic experimental data have been widely used as benchmarks for CFD verification by authoritative institutions such as the International Towing Tank Conference (ITTC). By comparing the CFD calculation results with the public experimental data, the reliability and accuracy of this numerical method can be objectively evaluated. The drag force is computed using identical solver settings and boundary conditions, and the results are compared with the experimental data published by ITTC at velocities of 3.045 m/s, 5.14 m/s and 6.09 m/s. The relative errors are 5.84%, 2.19% and 2.04%, respectively. The validation tests show that the deviation between the computed drag force and the experimental value is within 6%, confirming the reliability of the CFD setup. This validation result provides a reliable foundation for the engineering test of the work.

An automatic design and analysis framework for the multi-UUVs is established in this work. The design variables are firstly input via MATLAB. The shape point files describing the multi-UUVs are then passed into SOLIDWORKS for shape modeling. Subsequently, we import the generated model into ANSYS for mesh generation and CFD evaluation. This whole process will be repeatedly called within the optimization loop.

To validate the effectiveness of the proposed MAHSTO in real-world engineering applications, it is compared against four state-of-the-art counterparts, namely OTMTO, HAM, MTDE-ADKT, and MOMFEA-SADE, on two partially heterogeneous multi-UUVs design tasks. Tables 6 and 7 show the drag force results of the multi-UUVs obtained through all methods.

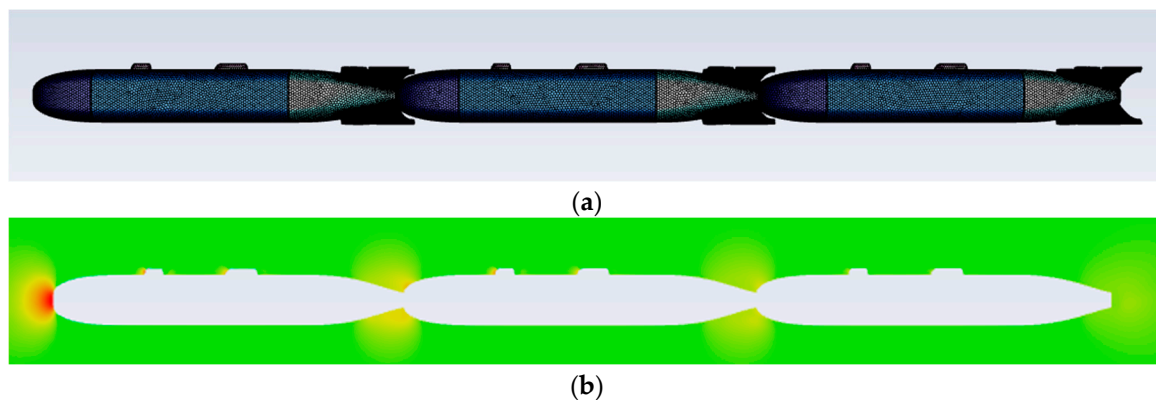
Table 6. Comparison results of the multi-UUVs with rotational body shape design.

	OTMTO	HAM	MTDE-ADKT	MOMFEA-SADE	MAHSTO
Drag force/N	296.27	292.94	301.83	326.75	259.24

Table 7. Comparison results of the multi-UUVs with near-rectangular body shape design.

	OTMTO	HAM	MTDE-ADKT	MOMFEA-SADE	MAHSTO
Drag force/N	312.16	334.27	339.95	347.82	291.23

The mesh division and fluid contour of the optimal design obtained by MAHSTO is shown in Figure 6.

**Figure 6.** Optimal design obtained by MAHSTO: (a) mesh division of the multi-UUVs obtained by MAHSTO, (b) fluid contour of the multi-UUVs obtained by MAHSTO.

6. Conclusions

To address the challenge of partially heterogeneous multitask optimization problems, a manifold alignment and hierarchical surrogate-assisted multitask optimization algorithm (MAHSTO) is proposed. In this method, an autoencoder is constructed to map the decision spaces of both tasks into a shared low-dimensional latent space, where cross-task knowledge is exchanged implicitly without requiring explicit variable-to-variable correspondence. Furthermore, a hierarchical multisurrogate model with adaptive sampling is constructed. It consists of a shared surrogate model and two task-specific surrogate models. With the help of hybrid surrogate models and adaptive sampling criteria, the optimization efficiency improves substantially. We test MAHSTO on six mathematical case pairs and it outperforms four classic algorithms. We also apply MAHSTO to solve the multitask optimization problem of multi-UUVs. The results verify its competitiveness in handling partially heterogeneous engineering problems.

Despite these promising results, there are still several limitations in the current research. This framework is most suitable for problems in which the two tasks share common variables that significantly influence the objective functions, enabling the performance alignment loss to better facilitate positive knowledge transfer. The manifold alignment is further dependent on the quality of the latent space. If the training samples fail to adequately cover the critical regions of the decision space, knowledge transfer may become ineffective or even induce negative transfer. The optimization of this work only takes the drag force as the optimization objective and the fullness as the constraint. Extending the proposed framework to the multiobjective optimization of underwater vehicles is an important direction for future research.

In light of the above limitations, future research can be conducted in the following directions. One is to integrate the feature selection when building the shared latent space.

This might reduce the information loss during the knowledge transfer process. Another direction is to incorporate the measurement of task similarity into the framework to boost adaptive transfer and optimization efficiency of the algorithm.

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

Funding: This research work is supported by the National Natural Science Foundation of China (Grant No. 52575312 and Grant No. 52505251) and the China Postdoctoral Science Foundation (Grant No. GZC20242194 and Certificate No. 2025M774335).

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

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

References

1. Wang, Z.; Cao, L.; Feng, L.; Jiang, M.; Tan, K.C. Evolutionary Multitask Optimization With Lower Confidence Bound-Based Solution Selection Strategy. *IEEE Trans. Evol. Comput.* **2025**, *29*, 132–144.
2. Dong, H.; Han, X. Data-driven whale optimization for high-dimensional expensive problems using hybrid local search strategy. *Swarm Evol. Comput.* **2026**, *106*, 102427.
3. Gupta, A.; Ong, Y.-S.; Feng, L. Multifactorial evolution: Toward evolutionary multitasking. *IEEE Trans. Evol. Comput.* **2016**, *20*, 343–357.
4. Gao, W.; Cheng, J.; Gong, M.; Li, H.; Xie, J. Multiobjective multitasking optimization with subspace distribution alignment and decision variable transfer. *IEEE Trans. Emerg. Top. Comput. Intell.* **2022**, *6*, 818–827.
5. Dong, H.; Wang, J.; Wang, W.; Shen, J. Data-driven multi-task global optimization with two-stage knowledge transfer for blended-wing-body underwater gliders design. *Appl. Soft Comput.* **2026**, *194*, 114867.
6. Wu, S.-H.; Zhan, Z.-H.; Tan, K.C.; Zhang, J. Transferable adaptive differential evolution for many-task optimization. *IEEE Trans. Cybern.* **2023**, *53*, 7295–7308.
7. Lin, Q.; Wu, Z.; Ma, L.; Gong, M.; Li, J.; Coello, C.A.C. Multiobjective multitasking optimization with decomposition-based transfer selection. *IEEE Trans. Cybern.* **2024**, *54*, 3146–3159.
8. Cui, Z.; Zhao, B.; Zhao, T.; Cai, X.; Chen, J. Adaptive multi-task evolutionary algorithm based on knowledge reuse. *Inf. Sci.* **2023**, *648*, 119568.
9. Wu, S.-H.; Zhan, Z.-H.; Tan, K.C.; Zhang, J. Orthogonal transfer for multitask optimization. *IEEE Trans. Evol. Comput.* **2023**, *27*, 185–200.
10. Han, H.; Bai, X.; Hou, Y.; Qiao, J. Multitask particle swarm optimization with heterogeneous domain adaptation. *IEEE Trans. Evol. Comput.* **2024**, *28*, 178–192.
11. Wang, C.; Mahadevan, S. Heterogeneous domain adaptation using manifold alignment. In *Proceedings of the International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16–22 July 2011*; AAAI Press: Menlo Park, CA, USA; pp. 1541–1546.
12. Zhang, W.; Wu, D. Manifold embedded knowledge transfer for brain-computer interfaces. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 1117–1127.
13. Feng, L.; Ong, Y.-S.; Jiang, S.; Gupta, A. Autoencoding evolutionary search with learning across heterogeneous problems. *IEEE Trans. Evol. Comput.* **2017**, *21*, 760–771.
14. Tang, Z.; Gong, M.; Wu, Y.; Liu, W.; Xie, Y. Regularized evolutionary multitask optimization: Learning to intertask transfer in aligned subspace. *IEEE Trans. Evol. Comput.* **2021**, *25*, 262–276.
15. Sumpter, D.J.T. The principles of collective animal behaviour. *Philos. Trans. R. Soc. B Biol. Sci.* **2006**, *361*, 5–22.
16. SPortugal, J.; Hubel, T.Y.; Fritz, J.; Heese, S.; Trobe, D.; Voelkl, B.; Hailes, S.; Wilson, A.M.; Usherwood, J.R. Upwash exploitation and downwash avoidance by flap phasing in ibis formation flight. *Nature* **2014**, *505*, 399–402.
17. Lissaman, P.B.S.; Shollenberger, C.A. Formation flight of birds. *Science* **1970**, *168*, 1003–1005.

18. Weihs, D. Hydromechanics of fish schooling. *Nature* **1973**, *241*, 290–291.
19. Völkl, B.; Fritz, J. Relation between travel strategy and social organization of migrating birds with special consideration of formation flight in the northern bald ibis. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **2017**, *372*, 20160235.
20. Yuan, Z.; Chen, M.; Jia, L.; Ji, C.; Incecik, A. Wave-riding and wave-passing by ducklings in formation swimming. *J. Fluid Mech.* **2021**, *928*, R2.
21. Yin, Z.; He, W.; Zou, Y. Efficient formation of flapping-wing aerial vehicles based on wild geese queue effect. *Acta Autom. Sin.* **2021**, *47*, 1355–1367.
22. Zhou, Z.; Duan, H.; Fan, Y. Unmanned aerial vehicle close formation control based on the behavior mechanism in wild geese. *Sci. China Technol. Sci.* **2017**, *47*, 230–238.
23. Jiang, T.; Shi, Y.; Xu, P. Drag simulation analysis and fuel-saving rate calculation for heavy-duty truck platooning. *Intern. Combust. Engine Powerpl.* **2022**, *39*, 81–85.
24. DARPA Tactical Technology Office Broad Agency Announcement: Sea Train; Rep. HR001120S0043; DARPA: Arlington, VA, USA, 2020.
25. Reckzeh, D. Aerodynamic design of the A400M high-lift system. In *Proceedings of the 26th International Congress of the Aeronautical Sciences, Anchorage, AK, USA, 14–19 September 2008*; International Council of the Aeronautical Sciences (ICAS): Bonn, Germany, 2008; pp. 1–10.
26. Zhang, Y. Analysis of Large AUVs and Surface Reconnaissance Technology. *Digit. Ocean Underw. Warf.* **2023**, *6*, 406–412.
27. Zhao, Y.; Ye, S.; Chen, X.; Xia, Y.; Zheng, X. Polynomial response surface based on basis function selection by multitask optimization and ensemble modeling. *Complex Intell. Syst.* **2022**, *8*, 1015–1034.
28. Hardy, R.L. Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **1971**, *76*, 1905–1915.
29. Tran, N.K.; Kühle, L.C.; Klau, G.W. A critical review of multi-output support vector regression. *Pattern Recognit. Lett.* **2024**, *178*, 69–75.
30. Pang, Y.; Zhang, S.; Jin, Y.; Wang, Y.; Lai, X.; Song, X. Surrogate information transfer and fusion in high-dimensional expensive optimization problems. *Swarm Evol. Comput.* **2024**, *88*, 101586.
31. Ginsbourger, D.; Le Riche, R.; Carraro, L. Kriging is well-suited to parallelize optimization. In *Computational Intelligence in Expensive Optimization Problems*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 131–162.
32. Wang, H.; Jin, Y.; Doherty, J. A generic test suite for evolutionary multifidelity optimization. *IEEE Trans. Evol. Comput.* **2017**, *22*, 836–850.
33. Alarie, S.; Audet, C.; Diago, M.; Le Digabel, S.; Lebeuf, X. Inter-DS: A cost saving algorithm for expensive constrained multifidelity blackbox optimization. *Comput. Optim. Appl.* **2025**, *90*, 607–629.
34. Andrés-Thió, N.; Muñoz, M.A.; Smith-Miles, K. Methodology and challenges of surrogate modelling methods for multi-fidelity expensive black-box problems. *ANZIAM J.* **2024**, *66*, 35–61.
35. Chen, W.; Dong, H.; Wang, P.; Wang, X. Surrogate-assisted global transfer optimization based on adaptive sampling strategy. *Adv. Eng. Inform.* **2023**, *56*, 101914.
36. Gu, J.; Li, G.Y.; Dong, Z. Hybrid and adaptive meta-model-based global optimization. *Eng. Optim.* **2012**, *44*, 87–104.
37. Zhang, T.; Gong, W.; Li, Y. Multitask Differential Evolution with Adaptive Dual Knowledge Transfer. *Appl. Soft Comput.* **2024**, *165*, 112040.
38. Liang, Z.; Dong, H.; Liu, C.; Liang, W.; Zhu, Z. Evolutionary Multitasking for Multiobjective Optimization With Subspace Alignment and Adaptive Differential Evolution. *IEEE Trans. Cybern.* **2022**, *52*, 2096–2109.

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.