

A Generic Test Suite for Evolutionary Multi-Fidelity Optimization

Handing Wang, *Member, IEEE*, Yaochu Jin, *Fellow, IEEE*, John Doherty

Abstract—Many real-world optimization problems involve computationally intensive numerical simulations to accurately evaluate the quality of solutions. Usually the fidelity of the simulations can be controlled using certain parameters and there is a trade-off between simulation fidelity and computational cost, i.e., the higher the fidelity, the more complex the simulation will be. To reduce the computational time in simulation-driven optimization, it is a common practice to use multiple fidelity levels in search for the optimal solution. So far, not much work has been done in evolutionary optimization that considers multiple fidelity levels in fitness evaluations. In this work, we aim to develop test suites that are able to capture some important characteristics in real-world multi-fidelity optimization, thereby offering a useful benchmark for developing evolutionary algorithms for multi-fidelity optimization. To demonstrate the usefulness of the proposed test suite, three strategies for adapting the fidelity level of the test problems during optimization are suggested and embedded in a particle swarm optimization algorithm. Our simulation results indicate that the use of changing fidelity is able to enhance the performance and reduce the computational cost of the particle swarm optimization, which is desired in solving expensive optimization problems.

Index Terms—Multi-fidelity optimization, expensive simulation-driven optimization, test problems, evolutionary computation, particle swarm optimization.

I. INTRODUCTION

Simulation-driven optimization, in which function evaluations involve numerical simulations, plays an important role in industrial design, such as aerodynamic design optimization [1]. Such numerical simulations are usually computationally expensive, yet the computational complexity of the simulations can be tuned. For example, the complexity of computational fluid dynamics (CFD) simulations can be tuned by changing the number of iterations [2], [3], [4], [5]. Simulation-driven optimization where the fidelity of the simulations can be controlled are known as multi-fidelity optimization (also variable fidelity optimization) [6], which has been commonly used in design of microwave structures [7], aerodynamic optimization [8], [9], ship design [10], engine design [11], and rocket design [12].

This work was supported in part by an EPSRC grant (No. EP/M017869/1) and in part by the National Natural Science Foundation of China (No. 61590922).

H. Wang is with the Department of Computer Science, University of Surrey, Guildford GU2 7XH, U.K. (e-mail: wanghanding.patch@gmail.com).

Y. Jin is with the Department of Computer Science, University of Surrey, Guildford GU2 7XH, U.K. (e-mail: yaochu.jin@surrey.ac.uk). He is also affiliated with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China. (*Corresponding author*)

J. Doherty is with the Department of Mechanical Engineering Sciences, University of Surrey, Guildford GU2 7XH, U.K. (e-mail: john.doherty@surrey.ac.uk).

Simulations with various fidelity levels can be seen as different approximations to the real fitness landscape of the problem, resulting in different approximation errors, and different computational costs as well. Generally speaking, high-fidelity simulations provide more accurate evaluations but also need more computational time. Thus, in many cases, it is not practical to perform optimization always relying on high-fidelity simulations, in particular when the number of needed fitness evaluations is large. For example, accurate 3-D CFD simulations often take hours for a single run [13]. On the contrary, low-fidelity simulations take less computational time but the results are less accurate. Consequently, there will be a trade-off between optimization performance and computational time in simulation-driven optimization [14], [15]. As the results from high-fidelity and low-fidelity simulations are more or less correlated [4], [16], [17], the motivation of multi-fidelity optimization is to take advantage of both high and low fidelity simulations to achieve satisfactory optimization performance with reduced computational cost [6], [18].

As simulations of multiple fidelity levels can be employed in optimization, controlling the fidelity levels affects both performance and efficiency of multi-fidelity optimization algorithms. Existing fidelity adjustment strategies can be individual-based [19] or generation-based [7]. The general idea of those fidelity control strategies is to make full use of various fidelity levels. More specifically, low-fidelity simulations can be more frequently used to evaluate solutions due to their cheap computational costs, therefore they can be used for pre-screening or a general exploration of the fitness landscape. By contrast, high-fidelity simulations are used when exploitation is needed to accelerate the optimization process. In some applications, two fidelity levels are used, i.e., low- and high-fidelity. For example, the approximate management framework (AMF) [20], which might be the first strategy to handle bi-fidelity fitness evaluations, uses the second-order Taylor series as the low-fidelity evaluation. A gradient-based search is applied on the low-fidelity fitness landscape, and the found optimum is re-evaluated using the high-fidelity fitness evaluation method. Although AMF is embedded in a trust region method to improve its performance [21], the gradient-based AMF remains sensitive to noises and the initial solution [22]. A genetic algorithm (GA) is used to perform search on the low-fidelity fitness landscape in [23], and the found optimum is re-evaluated using the high-fidelity evaluation method in every ten generations. Fidelity control strategies may become more complex when there are multiple fidelity levels. For instance in [7], the fidelity level is linearly increased as the evolution proceeds. In [4] the fidelity level is adjusted according to the

local correlation between the precise fitness evaluation and different approximate evaluation methods.

In case the fitness evaluation method of the lowest fidelity level is still computationally expensive, cheap surrogate models can be constructed from the data [24], [25] and employed to approximate the fitness for different fidelity levels [26], [27]. Many well-known models have been employed to build surrogate models [28], [29], [30], such as Gaussian processes (GP or Kriging) [31], response surface methodologies (RSM) [32], polynomial regression (PR) [33], [34], radial basis function (RBF) [10], support vector machine (SVM) [35], and artificial neural networks (ANN) [36], [37]. Among them, the Kriging model is one of the most popular models for multi-fidelity optimization [31], [38]. In [26], [39], [40], [41], fitness evaluations with different fidelity levels are approximated using multiple Kriging models, their connection (or called as bridge function) can be inferred to improve the efficiency of multi-fidelity optimization algorithms [42], [43], [44] due to the correlation between fitness evaluations with different fidelity levels [45]. Heterogeneous surrogate models are also adopted to approximate fitness evaluations with different fidelity levels [5], [32], [34], [37], [46]. Moreover, in different fidelity levels, surrogate models can be built on different subsets of decision variables [47], [48], [49], in which separable decision variables are grouped into different subsets.

Evolutionary algorithms (EAs), which are population-based optimization algorithms, have found many successful real-world applications [50]. However, a majority of EAs have not taken into account of various fidelity levels in fitness evaluations. In addition, little work has been reported so far on designing test problems dedicated to multi-fidelity optimization. Note that two test functions consisting of two functions representing two fidelity levels are defined in [51]. Unfortunately, the two functions used for representing two fidelity levels do not consider the correlation between fitness evaluations of different fidelity levels. The test problem recently reported in [2] considers partially converged simulations and has six fidelity levels. In that work, the correlation between different fidelity levels is taken into account, although these test problems are not scalable, nor easily extendible to other situations where various levels of fidelity exist.

As evidenced in several research areas of evolutionary computation, we are convinced that designing generic benchmark problems that take into account of the particular characteristics and challenges in multi-fidelity optimization are of great significance and highly desirable. The availability of such test problems are expected to promote the application of EAs to solving multi-fidelity optimization problems, which are commonly seen in real-world optimization. To this end, this work aims to design a multi-fidelity benchmark (MFB) that is able to capture typical characteristics commonly seen in real-world multi-fidelity optimization. The main contributions of this paper include:

- A few common characteristics of multi-fidelity optimization are extracted from different scenarios of real-world multi-fidelity simulations, which are represented by three types approximation errors.

- A generic test suite for multi-fidelity optimization, termed MFB, is designed based on the three types of approximation errors.
- Three fidelity control strategies are proposed to solve multi-fidelity optimization problems.

The rest of the paper is organized as follows. Section II first gives a brief introduction to multi-fidelity simulations. Section III presents the details of the proposed test problems. In Section IV, we propose three fidelity control strategies and embed them into a particle swarm optimization (PSO) algorithm for multi-fidelity optimization. Section V compares the performance of the proposed algorithms with a canonical PSO on the proposed MFB problems. Section VI concludes the paper and discusses the future work.

II. MULTI-FIDELITY SIMULATIONS

In simulation-driven optimization, no analytical mathematical function is available for evaluating the fitness of candidate solutions. Instead, numerical simulations must be conducted for fitness evaluations. More often than not, various simulation methods are available that differ either in methodology or in simulation resolution. For example, in aerodynamic optimization, Navier-Stokes equations or coupled boundary layer methods can be adopted, and for both methods, either 2-D or 3-D simulations can be carried out [13]. In addition, the resolution of the simulations can be adjusted by changing the resolution of the mesh size [32], or by controlling the tolerance error [4] or the maximum number of iterations [2] in CFD simulations. Therefore, the number of fidelity levels in multi-fidelity simulations could range from two to infinity, depending on the specific application. Note, however, that typically there is a trade-off between the fidelity (or approximation accuracy) and the computational cost of the simulation [25], [52], i.e., the higher the accuracy, the more time it requires to perform the simulation.

Simulations with different fidelity levels are different approximations of the theoretical exact fitness function (say the physical experiment). As a result, low-fidelity simulations will inevitably introduce errors, which may mislead the optimization process [53]. Thus, it is not feasible to perform optimization relying on low-fidelity simulations only, and the ultimate purpose of multi-fidelity optimization is to find the true optimum while reducing the computational time.

A. Approximation Errors in Multi-Fidelity Optimization

In some cases, simulations of different resolutions can be used. Taking CFD simulations as an example, where a flow domain is divided into a number of meshes and the governing equations are solved inside each mesh [32]. As illustrated in Fig. 1 (a), three different resolutions of meshes are set up for CFD simulations: a coarse, medium, and fine. Consequently, the CFD simulation with the fine mesh will produce the highest fidelity, although it also takes the largest amount of computational time. Theoretically, the CFD simulation becomes an infinite approximation when the number of meshes increases to infinity, which is infeasible due to the huge amount of computational time needed. Similarly, in trauma system design [54],

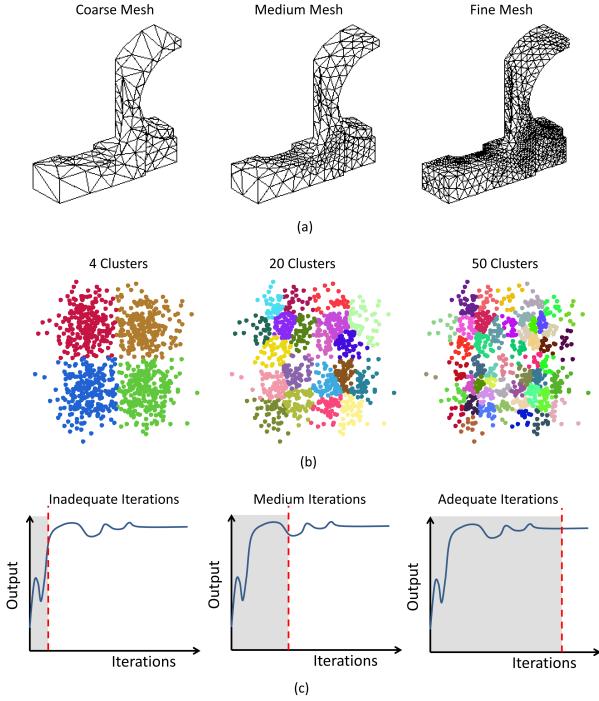


Fig. 1. Examples of approximation with different resolutions. (a) CFD simulations with different meshes. (b) Data with different clusters. (c) Calculation with different iterations.

which is a data-driven constrained multi-objective optimization problem, an exact fitness evaluation needs to calculate the objectives and constraints using all data records, which is very time-consuming. To reduce the computational time, the data can be clustered into groups to obtain approximated fitness values. As shown in Fig. 1 (b), the center in each cluster can be seen as the representative of the dataset. As a result, the number of data points used in calculating the fitness values can be reduced. A smaller number of clusters in the dataset requires less computational time but at the cost of accuracy. In the examples mentioned above, the number of meshes, or the number of clusters is a parameter that can control the fidelity level of fitness evaluations. Note that local errors introduced by approximation can hardly maintain a strict consistency with the global error [52]. In this work, we define such errors as resolution errors. Resolution errors can also result from partial or incomplete simulations. For example, in CFD simulations, usually a large number of iterations are needed to accurately solve differential equations [2]. In Fig. 1 (c), the blue lines are the convergence curves of CFD simulations and the red dot lines denote the termination criterion. It is clear that different termination criteria end up with different fidelity levels. The CFD calculation with a large number of iterations provides the most accurate result but takes the longest computational time. Resolution errors resulting from different factors may share some similarities: their landscape in the whole search space is multi-modal and may become smoother when the fidelity level increases [2].

In addition to deterministic simulations, stochastic simulations [55] have been applied to portfolio optimization [56], chemical reactions [57], and biochemical reactions [58]. Due

to the stochastic nature, a solution evaluated by the same simulation may have different fitness values in different runs. We define errors resulting from stochastic simulations as stochastic errors.

Finally, some simulations may suffer from instability or even crashes for cases having very complex flow features, particularly for cheap CFD simulations with limited models of physics [5], [59]. We define errors caused by instability in simulations as instability errors.

In summary, we consider three types of errors in multi-fidelity optimization in this work: resolution, stochastic, and instability errors.

- **Resolution errors:** The landscape of resolution errors is multi-modal due to the inconsistency of global and local errors. It is deterministic but gets smoother when the fidelity level increases.
- **Stochastic errors:** The errors are stochastic, i.e., the fitness value of the same solution varies in different simulation runs. The stochastic error decreases as the fidelity level increases.
- **Instability errors:** They represent failed simulations. A failure happens at a certain probability, and the probability decreases as the fidelity level increases.

B. A Case Study

To help further understand multi-fidelity optimization, here we use the RAE2822 airfoil design problem as a case study. The airfoil geometry is represented by 14 control points of a non-rational B-spline curve. The geometry of the baseline design is plotted in Fig. 2. In airfoil design, the objective function is to minimize the normalized ratio between the drag and lift coefficients in two design conditions:

$$f_{Airfoil} = \min \sum_{i=1}^2 w_i \left(\frac{C_d}{C_l} / \frac{C_d^b}{C_l^b} \right)_i \quad (1)$$

where C_d , C_l , C_d^b and C_l^b are the drag and lift coefficients of a new design and the baseline design, respectively. CFD simulations must be conducted to calculate those coefficients.

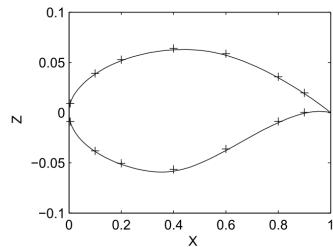


Fig. 2. The geometry of the baseline design with 14 control points of RAE2822 airfoil.

Note that different numbers of iterations in the CFD simulations result in different fidelity levels. As a fully-converged CFD simulation is computationally very expensive, we cannot afford a large number of fully-converged CFD simulations. For this reason, we calculate the objective values of 70 different geometries obtained by Latin hypercube sampling in

the 14-dimensional decision space with different numbers of iterations ranging from 50 to 900 in the CFD simulations. Assuming that the results from the CFD simulation with 900 iterations are accurate, we can then calculate the errors of the rest simulations by comparing them with the results obtained using 900 iterations. Figs. 3 and 4 show the individual errors and the average root mean square error (RMSE) of 70 samples changing over the number of iterations in CFD simulations. From these results, we can see that the error of each incomplete (partial) simulation does not decrease monotonically as the number of iterations increases, whereas the average RMSE decreases nearly monotonically with the increasing number of iterations. Moreover, two out of the 70 CFD simulations have crashed, which indicates that CFD simulations for airfoil design suffer from instability.

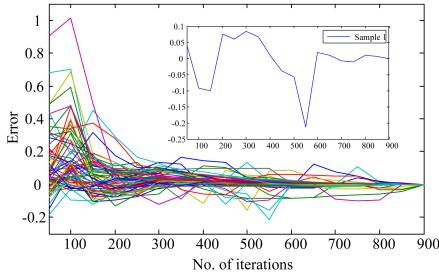


Fig. 3. Errors of 70 samples changing over the number of iterations in CFD simulations for RAE2822 airfoil.

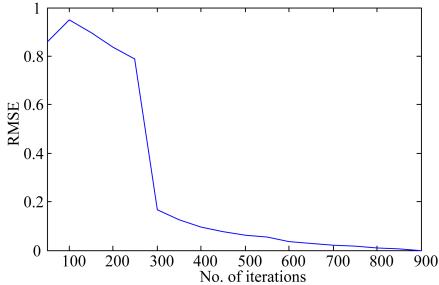


Fig. 4. Average RMSE of 70 samples varying over the number of iterations in the CFD simulation for RAE2822 airfoil.

We can see that the CFD simulations in the above case study suffer from two types of errors: resolution and instability errors. The experimental results confirm the characteristics we have discussed.

III. FORMULATION OF MULTI-FIDELITY BENCHMARK PROBLEMS

As discussed in Section II, multi-fidelity optimization aims to adaptively control the level of fidelity of the simulations to reduce the computational cost. However, in practice, it is not straightforward for a computer scientist to run the CFD simulations with varying fidelity levels. In addition, a large number of simulations are required for designing and testing a multi-fidelity optimization algorithm, which makes it intractable if all fitness evaluations must rely on CFD simulations. To address the above issues in designing multi-fidelity optimization algorithms, in this paper, we design a

set of multi-fidelity benchmark (MFB) problems that are able to capture a few main characteristics found in multi-fidelity simulations, thereby facilitating the design of evolutionary multi-fidelity optimization algorithms.

A. Design Principles

To emulate the mechanisms of fidelity control in simulations, we introduce a parameter ϕ in our benchmark problems, in addition to the d -dimensional decision vector \mathbf{x} . If $f(\mathbf{x})$ denotes the exact objective function, a simulation of a fidelity level ϕ approximates $f(\mathbf{x})$ with a certain error $e(\mathbf{x}, \phi)$. Thus, the output from the simulation $\tilde{f}(\mathbf{x}, \phi)$ can be formulated as follows:

$$\tilde{f}(\mathbf{x}, \phi) = f(\mathbf{x}) + e(\mathbf{x}, \phi), \mathbf{x} \in \Omega, \phi \in \Psi \quad (2)$$

where Ω and Ψ are the entire set of \mathbf{x} and ϕ . Parameter ϕ provides the optimization algorithm a handle to adapt the fidelity level, which can be continuous or discrete, depending on the number of available fidelity levels. In this work, ϕ is defined to range from 0 to 10000, where 0 represents the lowest fidelity level and 10000 represents the highest fidelity level. Various error functions $e(\mathbf{x}, \phi)$ result in different problems. To capture different characteristics of resolution, stochastic and instability errors discussed in Section II, specific error functions are designed in the following sub-sections.

In addition to the controllable fidelity level, the proposed MFB also takes into account the computational cost corresponding with different fidelity levels. To achieve this, we introduce a function $c(\phi)$ representing the computational cost for evaluating a single solution associated with the fidelity level ϕ . Therefore, each MFB problem outputs two values for each evaluated solution, namely, the fitness $\tilde{f}(\mathbf{x}, \phi)$, and the computational cost $c(\phi)$.

B. Implemented Instances

As we discussed in the previous subsection, generic benchmark problems for multi-fidelity optimization can be generated by setting $f(\mathbf{x})$, $e(\mathbf{x}, \phi)$, $c(\phi)$, and Ψ . To focus on the three types of approximation errors in multi-fidelity optimization, we adopt one single-objective test function as an example of the exact objective function $f(\mathbf{x})$ to design all benchmark problems in this test suite. Since the landscape of most real-world optimization problems is multi-modal, the exact objective function should be scalable and multi-modal. Without loss of generality, we employ the following modified Rastrigin function:

$$\min f_e(\mathbf{x}) = \sum_{i=1}^d [x_i^2 + 1 - \cos(10\pi x_i)] \quad (3)$$

$$\mathbf{x} \in [-1, 1]^d$$

where the d -dimensional decision variable are defined in $[-1, 1]^d$, and its global optimum \mathbf{x}^* is $(0, \dots, 0)$.

Generally speaking, the computational cost for evaluating a single solution increases as the level of fidelity increases. In the proposed MFB, we take into account two different types of relationship between the computational cost and the fidelity as follows:

$$c_l(\phi) = \phi \quad (4)$$

$$c_{nl}(\phi) = (0.001\phi)^4 \quad (5)$$

where $c_l(\phi)$ indicates a linear relationship and $c_{nl}(\phi)$ a nonlinear relationship. As shown in Fig. 5, both curves of $c_l(\phi)$ and $c_{nl}(\phi)$ begin and end with the same computational cost for the lowest and highest fidelity levels. To separate those two curves, we use a biquadrate function for $c_{nl}(\phi)$ rather than a quadratic or cubic function.

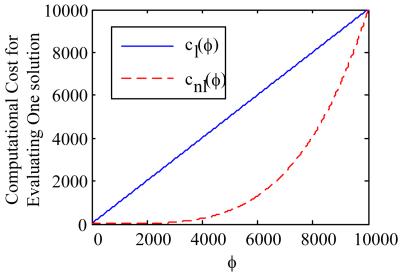


Fig. 5. Linear ($c_l(\phi)$) and nonlinear ($c_{nl}(\phi)$) relationships between the computational cost and simulation fidelity.

The MFB problems try to emulate three types of errors in multi-fidelity simulations, i.e., resolution, stochastic, and instability errors, referring to Section II for details. In this work, we implemented 13 instances to cover the three error types, which are denoted as Type I, II, and III, respectively. Theoretically, an arbitrary number of instances can be created from the generic MFB if the modified Rastrigin function is replaced with other objective functions.

1) *Type I: MFB1-MFB7*: As shown in Section II-B, the main characteristic of resolution errors is the inconsistency between the global and local errors. To capture this characteristics, we emulate the resolution error $e_r(\mathbf{x}, \phi)$ using Equation (6):

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

where a is the maximum error, w determines the number of local optima, and b influences the offset of the global optimum at a certain fidelity level ϕ to the true global optimum \mathbf{x}^* . Note that $a(x_i, \phi)$, $w(\phi)$, and $b(\phi)$ decrease as ϕ increases. Among them, $a(x_i, \phi)$ can be either independent of or dependent on \mathbf{x} .

Equations (7)-(10) are four different instantiations of resolution errors, e_r^1 , e_r^2 , e_r^3 , and e_r^4 :

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

where

$$a^1(\phi) = \theta^1(\phi), w^1(\phi) = 10\pi\theta^1(\phi), b^1(\phi) = 0.5\pi\theta^1(\phi),$$

$$\theta^1(\phi) = 1 - 0.0001\phi \quad (7)$$

$$e_r^2(\mathbf{x}, \phi) = \sum_{i=1}^d a^2(\phi) \cos(w^2(\phi)x_i + b^2(\phi) + \pi),$$

where

$$a^2(\phi) = \theta^2(\phi), w^2(\phi) = 10\pi\theta^2(\phi), b^2(\phi) = 0.5\pi\theta^2(\phi),$$

$$\theta^2(\phi) = e^{-0.00025\phi} \quad (8)$$

$$e_r^3(\mathbf{x}, \phi) = \sum_{i=1}^d a^3(\phi) \cos(w^3(\phi)x_i + b^3(\phi) + \pi),$$

where

$$a^3(\phi) = \theta^3(\phi), w^3(\phi) = 10\pi\theta^3(\phi), b^3(\phi) = 0.5\pi\theta^3(\phi),$$

$$\theta^3(\phi) = \begin{cases} 1 - 0.0002\phi & 0 \leq \phi < 1000 \\ 0.8 & 1000 \leq \phi < 2000 \\ 1.2 - 0.0002\phi & 2000 \leq \phi < 3000 \\ 0.6 & 3000 \leq \phi < 4000 \\ 1.4 - 0.0002\phi & 4000 \leq \phi < 5000 \\ 0.4 & 5000 \leq \phi < 6000 \\ 1.6 - 0.0002\phi & 6000 \leq \phi < 7000 \\ 0.2 & 7000 \leq \phi < 8000 \\ 1.8 - 0.0002\phi & 8000 \leq \phi < 9000 \\ 0 & 9000 \leq \phi < 10000 \end{cases} \quad (9)$$

$$e_r^4(\mathbf{x}, \phi) = \sum_{i=1}^d a^4(x_i, \phi) \cos(w^4(\phi)x_i + b^4(\phi) + \pi),$$

where

$$a^4(x_i, \phi) = \theta^4(\phi)\psi(x_i), w^4(\phi) = 10\pi\theta^4(\phi), b^4(\phi) = 0.5\pi\theta^4(\phi),$$

$$\theta^4(\phi) = 1 - 0.0001\phi, \psi(x_i) = 1 - |x_i - x_i^*| \quad (10)$$

In the above instantiations, $\theta^1(\phi)$ and $\theta^4(\phi)$ are linear monotonically decreasing functions, $\theta^2(\phi)$ is a nonlinear monotonically decreasing function, and $\theta^3(\phi)$ is a piecewise linear monotonically decreasing function. In $e_r^1(\mathbf{x}, \phi)$, $e_r^2(\mathbf{x}, \phi)$, and $e_r^3(\mathbf{x}, \phi)$, $a(\phi)$ is independent of \mathbf{x} , whereas in error $e_r^4(\mathbf{x}, \phi)$, $a(x_i, \phi)$ is dependent on \mathbf{x} via $\psi(x_i)$, which increases the deceptiveness of the global optimum \mathbf{x}^* .

In this paper, we construct seven test problems of Type I resolution errors. Table I lists the detailed descriptions of MFB1-MFB7. To help further understand the characteristics of Type I MFB problems, we plot in Fig. 6 the fitness landscape of one-dimensional MFB1 when the fidelity level varies ($\phi = 1000, 4000, 7000, 10000$). For MFB1, $\tilde{f}(\mathbf{x}, 10000)$ is equivalent to $f(\mathbf{x})$, which is the exact objective function. In Fig. 6, when $\phi = 1000$, $\tilde{f}(\mathbf{x}, 1000)$ cannot capture the general trend of $f(\mathbf{x})$, then ϕ increases to 4000, $\tilde{f}(\mathbf{x}, 4000)$ is a better approximation of $f(\mathbf{x})$, but the optimum of $\tilde{f}(\mathbf{x}, 4000)$ is around $x = -0.4$ rather than the true optimum $x^* = 0$, i.e., the true global optimum cannot be found at this fidelity level. When the fidelity level increases to a higher level $\phi = 7000$, the optimum of $\tilde{f}(\mathbf{x}, 7000)$ is near the true global optimum, but a small error remains in the decision space.

TABLE I
DEFINITIONS OF TYPE I MFB PROBLEMS (MFB1-7)

Problem	$f(\mathbf{x})$	Ψ	$e(\mathbf{x}, \phi)$	$c(\phi)$
MFB1	$f_e(\mathbf{x})$	$[0, 10000]$	$e_r^1(\mathbf{x}, \phi)$	$c_l(\phi)$
MFB2	$f_e(\mathbf{x})$	$[0, 10000]$	$e_r^2(\mathbf{x}, \phi)$	$c_l(\phi)$
MFB3	$f_e(\mathbf{x})$	$[0, 10000]$	$e_r^3(\mathbf{x}, \phi)$	$c_{nl}(\phi)$
MFB4	$f_e(\mathbf{x})$	$\{0, 1000, \dots, 10000\}$	$e_r^1(\mathbf{x}, \phi)$	$c_{nl}(\phi)$
MFB5	$f_e(\mathbf{x})$	$\{1000, 3000, 10000\}$	$e_r^2(\mathbf{x}, \phi)$	$c_{nl}(\phi)$
MFB6	$f_e(\mathbf{x})$	$\{1000, 10000\}$	$e_r^1(\mathbf{x}, \phi)$	$c_l(\phi)$
MFB7	$f_e(\mathbf{x})$	$[0, 10000]$	$e_r^4(\mathbf{x}, \phi)$	$c_l(\phi)$

The above example quantitatively illustrates the main characteristic of Type I MFB, where the objective function with a low fidelity level can provide a general trend of the exact

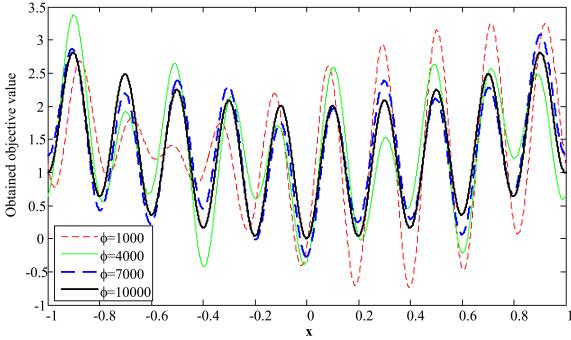


Fig. 6. The fitness landscape of 1D MFB1 with various fidelity levels ($\phi = 1000, 4000, 7000, 10000$).

objective function, but fails to deliver the detailed information of the exact fitness function $f(\mathbf{x})$. As a result, an optimizer may fail to find the true global optimum.

To better understand Type I MFB, in the following, we measure the Pearson correlation coefficient (ρ) and the RMSE of Type I MFB problems ($d = 30$) with the changing fidelity levels over 30 independent times. The settings of the experiments are as follows.

- Randomly choose 10000 solutions in the 30-dimensional decision space $[-1, 1]^{30}$ and calculate their exact fitness values $f(\mathbf{x})$.
- Calculate the values $\tilde{f}(\mathbf{x}, \phi)$ of those solutions with different fidelity levels. For MFB1, MFB2, MFB3, and MFB7, ϕ is sampled at an interval of 100 within $[0, 10000]$; for MFB4-6, all the feasible fidelity levels are sampled.
- Calculate ρ and RMSE of $\tilde{f}(\mathbf{x}, \phi)$ to $f(\mathbf{x})$.

The resulting average correlation coefficient and RMSE of 30-dimensional MFB1-7 with changing fidelity levels are shown in Fig. 7. MFB1 is a test problem with linear errors e_r^1 and infinite fidelity levels. The correlation coefficient nearly linearly increases with the increasing ϕ , and the RMSE nearly linearly decreases as ϕ increases. Due to local errors, these two metrics are not strictly linearly correlated with ϕ . MFB2 is a nonlinear version of MFB1, where both metrics change slowly when ϕ is larger than 5000. Therefore, the balance between approximation accuracy and computational cost is a challenge for adjusting the fidelity of MFB2. MFB3 is a piecewise linear version of MFB1 because of e_r^3 . When the RMSE remains unchanged with ϕ in a small region, a higher fidelity level should be avoided to reduce the cost. MFB4 has 11 available fidelity levels, whereas MFB5 has three fidelity levels. MFB6 only provides binary fidelity levels (1000 and 10000). MFB7 is more complicated than MFB1 due to the deceptiveness in finding the global optimum. In the interval of $\phi \in [9000, 10000]$, there is a sharp increase of RMSE around the true global optimum, which makes MFB7 hard to solve.

2) *Type II: MFB8-MFB11:* We employ the Gaussian distribution to model stochastic errors for Type II MFB problems:

$$e_s(\mathbf{x}, \phi) = N(\mu(\mathbf{x}, \phi), \sigma(\phi)) \quad (11)$$

where μ and σ stands for the mean and standard deviation, respectively.

Both $\mu(\mathbf{x}, \phi)$ and $\sigma(\phi)$ are non-increasing functions with ϕ . $\mu(\mathbf{x}, \phi)$ can be independent of or dependent on \mathbf{x} . In the following, we exemplify four different stochastic errors, e_s^1 , e_s^2 , e_s^3 , and e_s^4 :

$$\begin{aligned} e_s^1(\mathbf{x}, \phi) &= N(\mu^1(\phi), \sigma^1(\phi)) \\ \text{where } \mu^1(\phi) &= 0, \sigma^1(\phi) = 0.1\vartheta^1(\phi), \vartheta^1(\phi) = 1 - 0.0001\phi \end{aligned} \quad (12)$$

$$\begin{aligned} e_s^2(\mathbf{x}, \phi) &= N(\mu^2(\phi), \sigma^2(\phi)) \\ \text{where } \mu^2(\phi) &= 0, \sigma^2(\phi) = 0.1\vartheta^2(\phi), \vartheta^2(\phi) = e^{-0.0005\phi} \end{aligned} \quad (13)$$

$$e_s^3(\mathbf{x}, \phi) = N(\mu^3(\mathbf{x}, \phi), \sigma^3(\phi))$$

where

$$\begin{aligned} \mu^3(\mathbf{x}, \phi) &= \frac{0.1\vartheta^3(\phi)}{d}\vartheta(\mathbf{x}), \sigma^3(\phi) = 0.1\vartheta^3(\phi), \\ \vartheta^3(\phi) &= 1 - 0.0001\phi, \gamma(\mathbf{x}) = \sum_{i=1}^d (1 - |x_i - x_i^*|) \end{aligned} \quad (14)$$

$$e_s^4(\mathbf{x}, \phi) = N(\mu^4(\mathbf{x}, \phi), \sigma^4(\phi))$$

where

$$\begin{aligned} \mu^4(\mathbf{x}, \phi) &= \frac{0.1\vartheta^4(\phi)}{d}\vartheta(\mathbf{x}), \sigma^4(\phi) = 0.1\vartheta^4(\phi), \\ \vartheta^4(\phi) &= e^{-0.0005\phi}, \gamma(\mathbf{x}) = \sum_{i=1}^d (1 - |x_i - x_i^*|) \end{aligned} \quad (15)$$

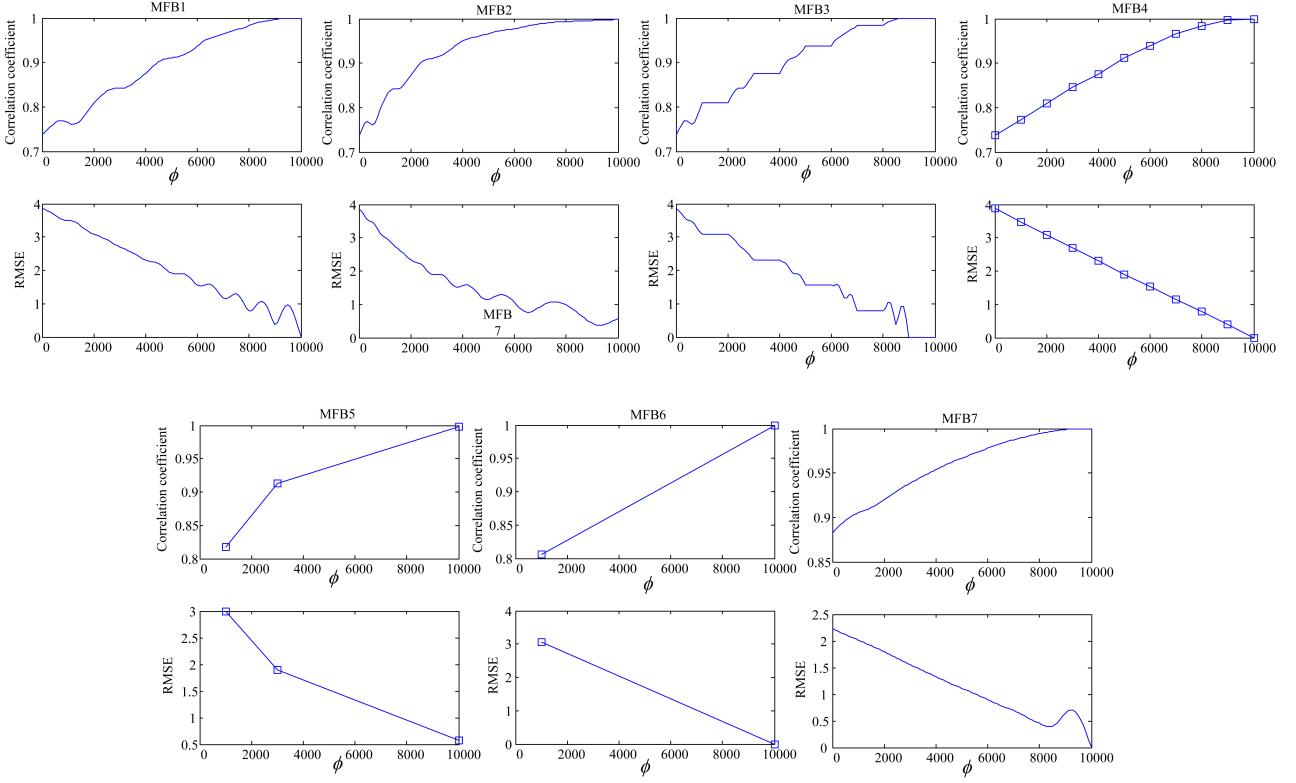
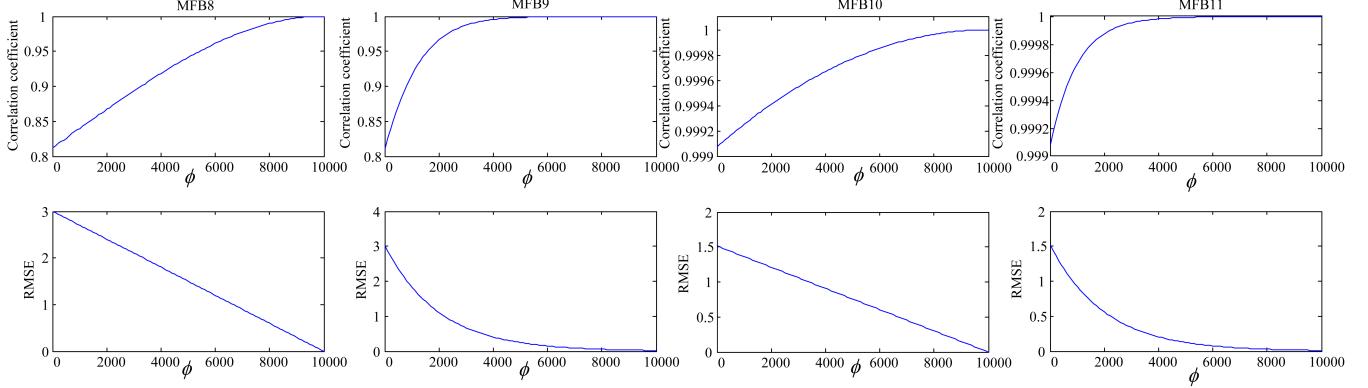
In the above equations, $\vartheta^1(\phi)$ and $\vartheta^3(\phi)$ are linear decreasing functions, $\vartheta^2(\phi)$ and $\vartheta^4(\phi)$ are nonlinear decreasing functions. $e_s^1(\mathbf{x}, \phi)$ and $e_s^2(\mathbf{x}, \phi)$ are independent of \mathbf{x} , whereas $e_s^3(\mathbf{x}, \phi)$ and $e_s^4(\mathbf{x}, \phi)$ are dependent on \mathbf{x} by $\gamma(\mathbf{x})$, which increases the deceptiveness of the global optimum \mathbf{x}^* .

In this paper, we construct four Type II test problems with stochastic errors. Table II presents the details of MFB8-MFB11. Since they are not stationary functions, it is hard to show their fitness landscape as in Fig. 6. Therefore, we quantitatively measure ρ and the RMSE of 30-dimensional MFB8-11 with the changing fidelity levels over 30 independent times. The experimental settings are the same as in the experiments in Section III-B1.

TABLE II
DEFINITIONS OF TYPE II MFB PROBLEMS (MFB8-11)

Problem	$f(\mathbf{x})$	Ψ	$e(\mathbf{x}, \phi)$	$c(\phi)$
MFB8	$f_e(\mathbf{x})$	$[0, 10000]$	$e_s^1(\mathbf{x}, \phi)$	$c_l(\phi)$
MFB9	$f_e(\mathbf{x})$	$[0, 10000]$	$e_s^2(\mathbf{x}, \phi)$	$c_{nl}(\phi)$
MFB10	$f_e(\mathbf{x})$	$[0, 10000]$	$e_s^3(\mathbf{x}, \phi)$	$c_l(\phi)$
MFB11	$f_e(\mathbf{x})$	$[0, 10000]$	$e_s^4(\mathbf{x}, \phi)$	$c_{nl}(\phi)$

The average correlation coefficient and RMSE of 30-dimensional MFB8-11 with the changing fidelity levels are shown in Fig. 8. MFB8 is a problem with linear e_s^1 stochastic errors. The correlation coefficient almost linearly increases with a growing ϕ , and the RMSE linearly decreases with a growing ϕ . MFB9 is a nonlinear version of MFB8. MFB10 and MFB11 are harder than MFB8 and MFB9, because $\mu(\mathbf{x}, \phi)$ is dependent on \mathbf{x} , which increases the deceptiveness in locating the true global optimum of both test problems.

Fig. 7. Average correlation coefficient and RMSE of Type I MFB problems ($d = 30$) with changing fidelity levels.Fig. 8. Average correlation coefficient and RMSE of Type II MFB problems ($d = 30$) with changing fidelity levels.

3) *Type III: MFB12-MFB13:* We adopt a function to simulate instability errors as follows:

$$e_{ins}(\phi) = \begin{cases} l & r \leq p(\phi) \\ 0 & r > p(\phi) \end{cases} \quad (16)$$

where l is an extremely large number, r is a random number in $[0, 1]$, and $p(\phi)$ is the probability of generating an outlier in the simulations.

The probability $p(\phi)$ decreases roughly as ϕ grows. Below, we provide two types of instability errors:

$$e_{ins}^1(\phi) = \begin{cases} l^1 & r \leq p^1(\phi) \\ 0 & r > p^1(\phi) \end{cases} \quad (17)$$

where

$$l^1 = 10d, p^1(\phi) = 0.1(1 - 0.0001\phi)$$

$$e_{ins}^2(\phi) = \begin{cases} l^2 & r \leq p^2(\phi) \\ 0 & r > p^2(\phi) \end{cases} \quad (18)$$

$$\text{where } l^2 = 10d, p^2(\phi) = e^{-0.001\phi-0.1}$$

where $p^1(\phi)$ is a linear decreasing function and $p^2(\phi)$ is a nonlinear decreasing function. The outliers l^1 and l^2 are set to be $10d$, which is a very large value compared with the range of the exact fitness function $f_e(\mathbf{x})$.

In this work, we construct two Type III test problems with instability errors. Table III provides a detailed description of MFB12-13. We quantitatively measure ρ and the RMSE of 30-dimensional MFB12 and MFB13 with a changing fidelity levels over 30 independent times. The experiments are conducted using the same settings as in the experiments in Section

III-B1.

TABLE III
DEFINITIONS OF TYPE III MFB PROBLEMS (MFB12-13)

Problem	$f(\mathbf{x})$	Ψ	$e(\mathbf{x}, \phi)$	$c(\phi)$
MFB12	$f_e(\mathbf{x})$	$[0, 10000]$	$e_{ins}^1(\phi)$	$c_l(\phi)$
MFB13	$f_e(\mathbf{x})$	$[0, 10000]$	$e_{ins}^2(\phi)$	$c_{nl}(\phi)$

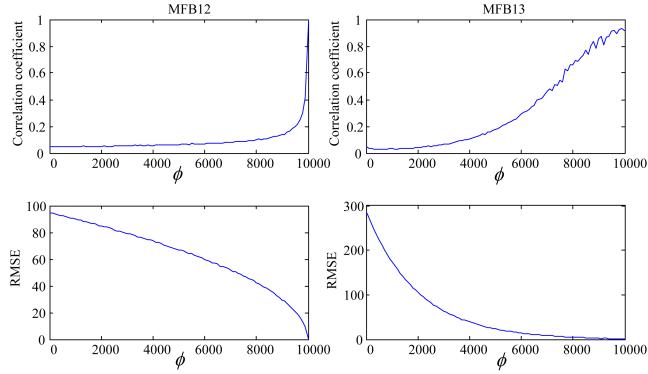


Fig. 9. Average correlation coefficient and RMSE of Type III MFB problems ($d = 30$) with changing fidelity levels.

The average correlation coefficient and RMSE of 30-dimensional MFB12-13 with a changing fidelity level are shown in Fig. 9. MFB12 and MFB13 are two test problems with instability errors (e_{ins}^1 and e_{ins}^2). The RMSEs of both problems decrease as ϕ increases. For MFB12, ρ is relatively small when ϕ is smaller than 8000, however, it increases rapidly after $\phi = 8000$. Compared with MFB12, ρ of MFB13 is smoother.

IV. FIDELITY ADJUSTMENT STRATEGIES FOR MULTI-FIDELITY OPTIMIZATION

In this section, we propose a number of strategies for fidelity control for multi-fidelity optimization. In this work, we use PSO as the optimizer, as it is easy to implement but has been shown to be a powerful optimizer for single-objective optimization [60], [61]. PSO starts with a swarm of random solutions in the decision space. In each generation g , solution \mathbf{x} changes its position by adding the velocity \mathbf{v} according to the following equations:

$$\mathbf{x}^{g+1} = \mathbf{x}^g + \mathbf{v}^{g+1} \quad (19)$$

$$\mathbf{v}^{g+1} = w\mathbf{v}^g + c_1 r_1 (\mathbf{x}_{pbest} - \mathbf{x}^g) + c_2 r_2 (\mathbf{x}_{gbest} - \mathbf{x}^g) \quad (20)$$

where \mathbf{x}_{pbest} is the personal best, \mathbf{x}_{gbest} is the global best, r_1 and r_2 are two random numbers in $[0, 1]$. According to [60], c_1 and c_2 are set to 1.49445, w is set to 0.729 in this work.

In the following, we propose three fidelity control strategies to be embedded in PSO, one generation-based, one individual-based, and one hybrid strategy. We then compare these three multi-fidelity PSO algorithms with the PSO algorithm using the exact fitness function only.

A. Generation-based Fidelity Adjustment Strategy

The basic assumption in designing generation-based multi-fidelity adjustment strategy is that in the early stage of the search, no high fidelity fitness evaluations are necessary. Therefore, low fidelity fitness evaluations can be more often used. In the later search stage, however, higher fidelity fitness evaluations become important for accurately locating the global optimum. Based on the above hypothesis, in the following, we propose a PSO with a generation-based adaptive fidelity adjustment strategy (called as PSO-AFAG). As shown in Fig. 10, PSO-AFAG follows the general framework of PSO, except for the components described by the shaded blocks. PSO-AFAG starts with the lowest fidelity fitness evaluations. In each generation, PSO-AFAG estimates the state of the swarm and decides whether the current fidelity level should be changed to the next fidelity level until the highest level is reached. For those test problems with an infinite number of fidelity levels, ϕ is uniformly discretized into N_f levels before running the algorithm.

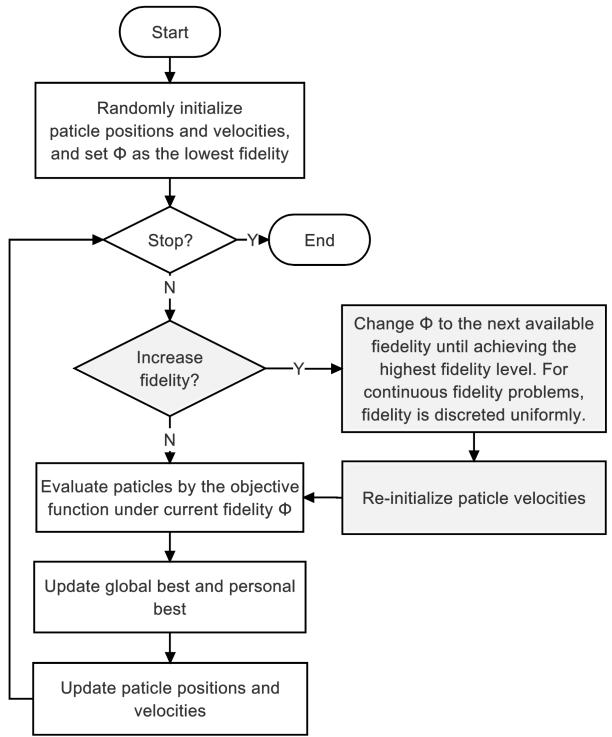


Fig. 10. Framework of PSO-AFAG.

PSO-AFAG adjusts the fidelity level to use according to the search status. The optimizer, PSO, concentrates on a small promising region (local optimum) after a number of generations for searching on $\tilde{f}(\mathbf{x}, \phi)$, then a higher fidelity level should be used to refine the search.

In fact, the search status of the swarm can be described by a pair of performance indicators of the swarm, namely, convergence and diversity. Taking a 1-D problem $\tilde{f}(\mathbf{x}, \phi)$ with a fidelity level ϕ in Fig. 11 as an example, where the solid line denotes the landscape of $\tilde{f}(\mathbf{x}, \phi_{max})$, the dotted line denotes the landscape of $\tilde{f}(\mathbf{x}, \phi)$, and the points are the particles of the swarm for optimizing $\tilde{f}(\mathbf{x}, \phi)$. In Fig. 11,

panel A illustrates an early stage of PSO, where the swarm distributes diversely, although no solution near the global optimum has been found. Panel B presents a scenario in the mid-stage of the search, where the swarm distributes within a smaller region, and the fitness of the particles has been improved. To continue optimizing $\tilde{f}(\mathbf{x}, \phi)$, PSO changes its status from panel D to panel C for approaching the global optimum of $\tilde{f}(\mathbf{x}, \phi)$, where both the swarm diversity and convergence on $\tilde{f}(\mathbf{x}, \phi_{\max})$ degrade while the convergence on $\tilde{f}(\mathbf{x}, \phi)$ improves. It is clear that the PSO is misled to a wrong region by the low fidelity simulation in panel C. In such cases, it is hard for the PSO to escape from the wrong region due to a reduced diversity, thus further search in this region might waste computational resources. In this case PSO-AFAg should switch to a higher fidelity level of simulation by increasing ϕ . The search status of the four situations illustrated in Fig. 11 can be plotted in Fig. 12 in terms of the swarm diversity and convergence on $\tilde{f}(\mathbf{x}, \phi_{\max})$. The status in panel C can be denoted as a degenerated status, in which both the convergence on $\tilde{f}(\mathbf{x}, \phi_{\max})$ and the swarm diversity degrade compared with those in the previous generations. Such a search status can be detected using the Pareto dominance [62], i.e., when the convergence and diversity pair of the current status is dominated by that in the previous generations.

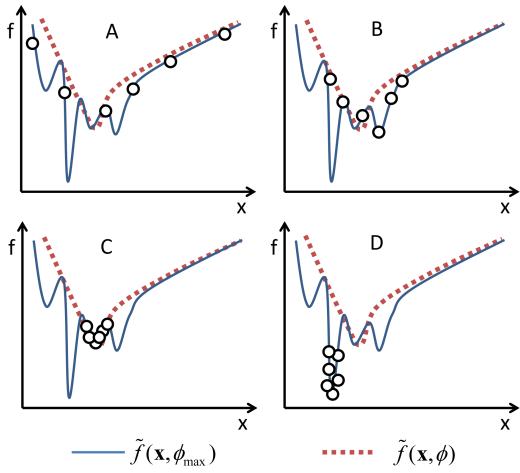


Fig. 11. Illustration of various states of a PSO search on a 1D problem $f(\mathbf{x}, \phi)$ with a fixed ϕ .

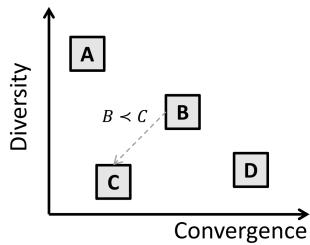


Fig. 12. Relationship between the convergence on $\tilde{f}(\mathbf{x}, \phi_{\max})$ and the swarm diversity in a search process for $\tilde{f}(\mathbf{x}, \phi)$.

In PSO-AFAg, the convergence is measured by the fitness of the current global best \mathbf{x}_{gbest} using the maximum fidelity

level:

$$\min M_c = \tilde{f}(\mathbf{x}_{gbest}, \phi_{\max}) \quad (21)$$

and the diversity is measured by the average of all the velocities in different dimensions:

$$\max M_d = \frac{1}{nd} \sum_{i=1}^n \sum_{j=1}^d v_i^j. \quad (22)$$

It is worth noting that M_c is evaluated using $\tilde{f}(\mathbf{x}, \phi_{\max})$ rather than $\tilde{f}(\mathbf{x}, \phi)$. In this case, degenerated status happens when the algorithm is in the stage when the local optimum of $\tilde{f}(\mathbf{x}, \phi)$ has been found, as further search using this fidelity level will lead to degeneration of $\tilde{f}(\mathbf{x}, \phi_{\max})$. In this particular situation, the fidelity level should be increased.

To avoid wasting computational budget on the degenerated status, a threshold is defined. When the accumulative computational cost on the degenerated status C_{deg} is larger than the threshold, a new fidelity will be employed. C_{deg} is set to zero immediately after a new fidelity level is used or a non-degenerated state occurs. The threshold for C_{deg} is assigned dependent on the fidelity levels as follows:

$$C_t(\phi) = \frac{C_{\max}}{N_f} \frac{n \times c(\phi) + c(\phi_{\max})}{n \times c(\phi_{\max})} \quad (23)$$

where C_{\max} is the total cost budget for PSO-AFAg, N_f is the number of available fidelity levels, n is the swarm size. Thus, a smaller threshold is assigned for a lower fidelity level to avoid unnecessary exploitation using low fidelity fitness evaluations, which is proportional to the ratio between the computational cost for the current fidelity and that for the highest fidelity level in each generation. When C_{deg} reaches $C_t(\phi)$, PSO-AFAg increases the fidelity level by one.

The implementation of AFAg in one generation is given in Algorithm 1. All the previous non-degenerated states are saved in set S . In each generation, the state pair $s = (M_c, M_d)$ of the current swarm is calculated. If s is dominated by any member in S , the accumulated cost for a degenerated state C_{deg} is added by the cost for fitness evaluations in this generation, which is $n \times c(\phi) + c(\phi_{\max})$, where $n \times c(\phi)$ is the cost for fitness evaluations, and $c(\phi_{\max})$ is the fitness evaluation cost to calculate M_c . Otherwise, C_{deg} is reset to zero. If C_{deg} is larger than the cost tolerance $C_t(\phi)$ for the current fidelity, PSO-AFAg changes ϕ to a higher level of fidelity and reinitializes the velocity of all the particles for the search using the new fidelity level.

B. Individual-based Fidelity Adjustment Strategy

The search process is not affected by different fidelity fitness functions as long as all the comparisons are ranked correctly. Given two solutions, if their low fidelity fitness difference is large, the ranking based on the high fidelity fitness is unlikely to be inconsistent with that based on the low fidelity fitness values. If the comparison based on the low fidelity fitness function is able to correctly rank them, there is no need to further evaluate them based on the high fidelity fitness function [2]. Therefore, it is possible to individually choose the fidelity level according to the fitness difference in the comparisons.

Algorithm 1 Pseudo code of AFAg in one generation.

Input: ϕ -the current fidelity model, S -state set, C_{deg} - accumulated cost for degenerated state.

- 1: Calculate the current state pair $s = (M_c, M_d)$ by Equations (21) and (22).
- 2: **if** s dominates any member in S **then**
- 3: Delete the dominated member from S .
- 4: Add s to S .
- 5: $C_{deg} = 0$.
- 6: **else**
- 7: **if** s is dominated by any member in S **then**
- 8: $C_{deg} = C_{deg} + n \times c(\phi) + c(\phi_{max})$.
- 9: **else**
- 10: Add s to S .
- 11: $C_{deg} = 0$.
- 12: **end if**
- 13: **end if**
- 14: **if** $C_{deg} >= C_t(\phi)$ **then**
- 15: Change ϕ to the next available level.
- 16: Re-initialize the velocities.
- 17: **end if**

Output: ϕ , S , and C_{deg} .

An adaptive fidelity adjustment strategy based on individuals, termed multi-fidelity evolutionary algorithm (MFEA) [2], is designed for partially converged simulations. When MFEA pairwise compares solutions, the fidelity level increases until the expected probability of rank reversal (i.e. the comparison inconsistency between the fitness evaluations of two fidelity levels) is smaller than a pre-specified threshold (set to 0.05 in MFEA), and the expected probability is predicted using a logistic regression model. However, the full cost of the re-evaluation is used when the fidelity level is changed in the MFB problems, i.e. another simulation with a new fidelity level is conducted, which is different from the problems in partially converged simulations where only the cost difference of two fidelity levels is added. Because partially converged simulations can be paused for re-evaluations. To implement the individual-based fidelity adjustment strategy in PSO on MFB problems, we modify the fidelity adjustment strategy in MFEA and embed it to PSO (called as PSO-AFAi).

Similar to MFEA, before the initialization of PSO-AFAi, 5d training data are generated using the Latin hypercube sampling method, and each sample is evaluated using the fitness functions with all available fidelity levels. With the training data, the relationship between the ϕ -fitness difference of two solutions $\Delta = |\tilde{f}(\mathbf{x}_1, \phi) - \tilde{f}(\mathbf{x}_2, \phi)|$ and the probability of a wrong ranking for a given fidelity level $P(\tilde{f}(\mathbf{x}_1, \phi_{max}) > \tilde{f}(\mathbf{x}_2, \phi_{max}) | \tilde{f}(\mathbf{x}_1, \phi) < \tilde{f}(\mathbf{x}_2, \phi))$ can be described by a logistic regression model as follows:

$$LR^\phi(\Delta) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \Delta)}} \quad (24)$$

In each generation, solutions are firstly evaluated using the lowest fidelity fitness function (ϕ_{min}). During the comparisons for updating the personal and global best, AFAi chooses a fidelity level to re-evaluate their solutions according to the

difference between the two fitness values. For two solutions \mathbf{x}_1 and \mathbf{x}_2 for comparison in Algorithm 2, the lowest fidelity level ϕ_{min} is used for comparison at first. If the fitness difference $\Delta = |\tilde{f}(\mathbf{x}_1, \phi_{min}) - \tilde{f}(\mathbf{x}_2, \phi_{min})|$ results in a high probability of wrong ranking (more than 0.05 as pre-specified in MFEA) according to the obtained logistic regression model for the lowest fidelity level $LR^{\phi_{min}}(\Delta)$, a higher estimated fidelity level ϕ' will be chosen, \mathbf{x}_1 and \mathbf{x}_2 are re-evaluated using the ϕ' -fitness function, Δ is updated to $|\tilde{f}(\mathbf{x}_1, \phi') - \tilde{f}(\mathbf{x}_2, \phi')|$. If the newly chosen higher fidelity level still results in a high probability of wrong ranking according to $LR^{\phi'}(\Delta)$, AFAi directly chooses the highest fidelity level ϕ_{max} to re-evaluate \mathbf{x}_1 and \mathbf{x}_2 for comparison. Therefore, three fidelity levels at most are involved for comparing two solutions.

Algorithm 2 Pseudo code of AFAi for comparing two solutions.

Input: ϕ_{min} -the minimal fidelity level, ϕ_{max} -the maximum fidelity level, \mathbf{x}_1 and \mathbf{x}_2 -two solutions for comparison.

- 1: **if** $LR^{\phi_{min}}(|\tilde{f}(\mathbf{x}_1, \phi_{min}) - \tilde{f}(\mathbf{x}_2, \phi_{min})|) < 0.05$ **then**
- 2: $\phi = \phi_{min}$.
- 3: **else**
- 4: Estimate ϕ' according to $|\tilde{f}(\mathbf{x}_1, \phi_{min}) - \tilde{f}(\mathbf{x}_2, \phi_{min})|$.
- 5: Evaluate \mathbf{x}_1 and \mathbf{x}_2 by the fitness function in ϕ' level.
- 6: **if** $LR^{\phi'}(|\tilde{f}(\mathbf{x}_1, \phi') - \tilde{f}(\mathbf{x}_2, \phi')|) < 0.05$ **then**
- 7: $\phi = \phi'$
- 8: **else**
- 9: Evaluate \mathbf{x}_1 and \mathbf{x}_2 with $\tilde{f}(\mathbf{x}, \phi_{max})$.
- 10: $\phi = \phi_{max}$.
- 11: **end if**
- 12: **end if**
- 13: Compare \mathbf{x}_1 and \mathbf{x}_2 based on $\tilde{f}(\mathbf{x}, \phi)$.

Unlike the strategy in MFEA, AFAi chooses the fidelity level based on the regression model. To estimate ϕ' in line 4 of Algorithm 2, a mapping relationship between the fidelity level and fitness difference Δ is constructed using the logistic regression model $LR^{\phi_{min}}(\Delta)$ as shown in Fig. 13, where $[\phi, \phi_{max}]$ is normalized to $[0.05, LR^\phi(0)]$. Taking the four fidelity levels in Fig. 13 as an example, when Δ falls in $[\Delta_{\phi_{min}}, \Delta_{\phi_1}]$, fidelity ϕ_1 is estimated as ϕ' ; when Δ falls in $[\Delta_{\phi_1}, \Delta_{\phi_2}]$, fidelity ϕ_2 is estimated as ϕ' ; and when Δ falls in $[\Delta_{\phi_2}, \Delta_{\phi_{max}}]$, fidelity ϕ_{max} is estimated as ϕ' .

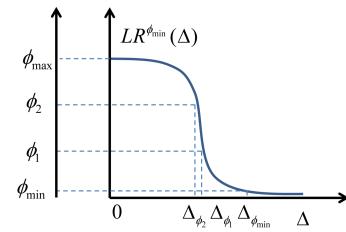


Fig. 13. Mapping the relationship between the fidelity level and the difference in their fitness values.

C. Hybrid Fidelity Adjustment Strategy

Generation- and individual-based fidelity adjustment strategies have their own advantages and disadvantages. To increase

the flexibility of fidelity adjustment strategies, we combine PSO-AFAg and PSO-AFAi to form a hybrid fidelity adjustment strategy and embed it in a PSO algorithm, PSO-AFAh for short.

PSO-AFAh follows the main framework of PSO-AFAi, but the highest fidelity level allowed in AFAi is adjusted by AFAg in every generation. Therefore, the fidelity adjustment strategy in PSO-AFAh is based on both generations and individuals. Consequently, PSO-AFAh adjusts the fidelity level according to the swarm status as PSO-AFAg does. Additionally, PSO-AFAh allows for various fidelity levels for different solutions in a generation as PSO-AFAi does.

V. EXPERIMENTAL RESULTS

In this section, we empirically examine the three multi-fidelity PSO variants on the proposed MFB problems. For a fair comparison, the PSO [60] using the fitness evaluations of the highest fidelity level only is employed as a base optimizer. All experiments in the following are conducted on the MFB problems with 30 decision variables. The same population size of 50 and computational budget ($C_{\max}=5e-09$) are applied to all compared algorithms. 30 independent runs are executed for each compared algorithm.

A. Further Analysis on MFB Problems

We compare PSO variants with different fixed fidelity levels on 30-dimensional MFB1-13 problems, where the fitness evaluations with a certain fidelity level $\tilde{f}(\mathbf{x}, \phi)$ are used in each PSO. For MFB4-6, all the available fidelity levels are tested; For other problems, every 1000 in the fidelity range is tested. Each PSO with 50 particles is run 30 times and stops by the cost budget 5e-09.

The average exact fitness values obtained by PSO with different fixed fidelity levels on Type I MFB problems are shown in Fig. 14. The higher fidelity fitness evaluations can generally lead PSO to better solutions. When ϕ is smaller than 4000, $\tilde{f}(\mathbf{x}, \phi)$ can provide a fine approximation to the exact objective function, especially in the area near the true global optimum. When ϕ is larger than 8000, $e(\mathbf{x}, \phi)$ reduces, better solutions are found by PSO with higher fidelity levels than that with lower fidelity levels. However, the search becomes less sufficient due to the limited budget, which is the reason why PSO with $\phi = 10000$ cannot perform better than that with $\phi = 9000$ on some problems.

The average exact fitness values obtained by PSO with different fixed fidelity levels on Type II MFB problems are shown in Fig. 15. For MFB8 and MFB9 with errors e_s^1 and e_s^2 , whose μ is independent on \mathbf{x} , PSO with higher fidelity levels can obtain better solutions than that with lower fidelity levels. However, the situation changes when μ is dependent on \mathbf{x} in MFB10 and MFB11, where the deceptive error from μ makes the two instances harder to be solved.

The average exact fitness values obtained by PSO with different fixed fidelity levels on Type III MFB problems are shown in Fig. 16. For both problems, a certain probability of outliers disturb the search of PSO, and PSO with various fixed fidelity levels have similar performance. Also, the outliers of MFB12 and MFB13 might mislead the surrogate construction.

B. Parameter Sensitivity Analysis

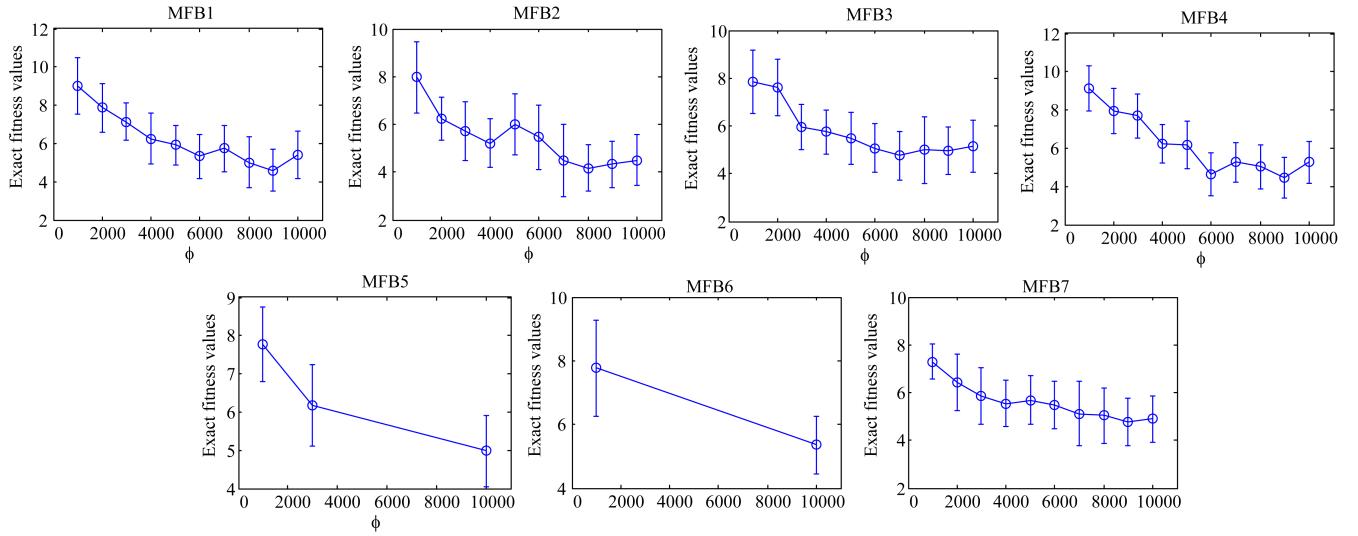
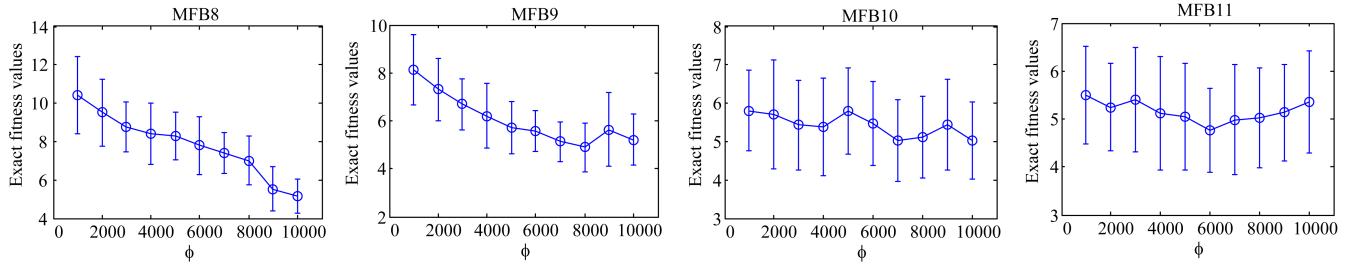
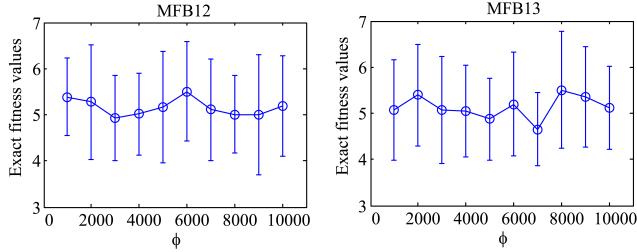
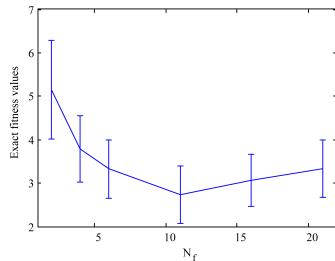
The number of discretized fidelity levels N_f is a parameter of PSO-AFA variants for those problems with infinite fidelity levels. We test PSO-AFAg with different N_f (2,4,6,11,16,21) on MFB1 ($d = 30$). The exact fitness values obtained by PSO-AFAg with different N_f are shown in Fig. 17. When N_f is small, the change between two fidelity levels is large. In this case, after a fidelity change, the obtained solutions using the previous fidelity fitness function are located far from the optimum of the higher fidelity fitness function, which is hard for PSO-AFAg to guide the swarm to the new promising region. That is the reason why PSO-AFAg often does not perform well. When N_f is too large, PSO-AFAg changes the fidelity level too frequently. As a result, the search for each fidelity is not sufficient, again resulting in poor performance of PSO-AFAg. From the results shown in Fig. 17, we can find that PSO-AFAg with $N_f = 11$ achieves the best performance. According to the above findings, we set N_f to 11 in PSO-AFA variants in the following experiments.

C. Comparative Studies

We compare PSO-AFA variants ($N_f = 11$) and PSO with the highest fidelity level ($\phi = 10000$), on the thirteen 30-dimensional MFB. In addition, a surrogate-assisted EA, Gaussian process surrogate model assisted evolutionary algorithm for medium-scale expensive problems, GPEME for short [63] is included in the comparisons. GPEME employs a Kriging model as the surrogate to approximate the highest-fidelity evaluations ($\phi = 10000$). The settings of GPEME are the same as those in [63], except that the Hooke & Jeeves method [64] instead of a GA is employed for optimization of the hyperparameters of the Kriging model. The results are summarised in Table IV. From these results, we see that PSO-AFAh and PSO-AFAg outperform PSO-AFAi, PSO, and GPEME on all the test problems.

Although GPEME employs a Kriging model to replace $\tilde{f}(\mathbf{x}, \phi_{\max})$ for saving computational cost, it performs worse than the PSO without using surrogate models, probably due to the poor quality of the Kriging model that is trained using 100 newest samples. Therefore, adopting a single surrogate model might be inefficient and inadequate, which further supports our hypothesis that using multi-fidelity evaluations is more flexible to find the best trade-off between computational cost and performance. The reason why PSO-AFAh and PSO-AFAg outperform PSO is that AFAh and AFAg adaptively assign the computational budget. At the early search stage, PSO-AFAh and PSO-AFAg use relatively low fidelity levels to find the promising region in which the global optimum might be located. In the latter search stage, they use high fidelity simulations to perform fine search for the optimum. By contrast, PSO always uses the maximum fidelity level during the whole optimization process, which might waste some computational budget at the beginning of the search.

On MFB1 to MFB7 with resolution errors, PSO-AFAh performs better than PSO-AFAg on five of the seven test problems. On MFB4 and MFB6, the advantage of PSO-AFAg over PSO-AFAh is not statistically significant. PSO-

Fig. 14. Average exact fitness values obtained by PSO with different fixed fidelity levels on MFB1-7 ($d = 30$).Fig. 15. Average exact fitness values obtained by PSO with different fixed fidelity levels on MFB8-11 ($d = 30$).Fig. 16. Average exact fitness values obtained by PSO with different fixed fidelity levels on MFB12-13 ($d = 30$).Fig. 17. Average exact fitness values obtained by PSO-AFAg with different N_f (2,4,6,11,16,21) on MFB1 ($d = 30$).

AFAi performs comparably well with PSO, partly due to the fact that the logic regression models are not accurate because

of the local errors. In addition, PSO-AFAh and PSO-AFAg do not perform as well on MFB5 and MFB6 as on other test problems, because there are only two or three available fidelity levels. From the experiment described in Section V-B, it is clear that PSO-AFAg is not suited for problems with a smaller number of fidelity levels, because the difference between the objective functions in two fidelity levels is too large to be captured by PSO-AFAg. PSO-AFAg has the similar performance on other problems with resolution errors, we can conclude that PSO-AFAg can adaptively deal with different types of resolution errors. With the individual-based AFA strategy, PSO-AFAh can further improve the performance of PSO-AFAg. For MFB8-11 with stochastic errors, PSO-AFAh and PSO-AFAg perform better than PSO-AFAi and PSO. Even for MFB10 and MFB11 with deceptive errors, PSO-AFAh and PSO-AFAg are still able to exhibit good performance. Comparing PSO-AFAh with PSO-AFAg, PSO-AFAh outperforms PSO-AFAg on MFB problems with stochastic errors. For MFB12 and MFB13 with the instability errors, PSO-AFAh and PSO-AFAg are less sensitive to the outliers in fitness evaluations and outperform PSO-AFAi and PSO. Overall, PSO-AFAg performs better than PSO-AFAh.

The previous experimental results indicate that PSO-AFAh and PSO-AFAg are two well-performing algorithms. To further understand the three proposed fidelity adjustment strategies, we plot the average computational cost and the obtained exact

TABLE IV

EXACT FITNESS VALUES OBTAINED BY PSO-AFA VARIANTS, PSO WITH THE HIGHEST FIDELITY FITNESS EVALUATION ($\phi = 10000$), AND GPEME ON MFB1-13 ($d = 30$). THE BEST AND SECOND BEST FITNESS VALUES AMONG ALL THE COMPARED ALGORITHMS FOR EACH PROBLEM ARE HIGHLIGHTED IN GRAY AND LIGHT GRAY.

Problem	PSO-AFAg	PSO-AFAi	PSO-AFAh	PSO	GPEME
MFB1	2.86e+00±6.67e-01	5.02e+00±9.39e-01	2.59e+00±6.19e-01	4.85e+00±1.04e+00	1.96e+01±2.20e+00
MFB2	3.25e+00±6.89e-01	4.51e+00±9.58e-01	2.88e+00±6.42e-01	4.79e+00±1.00e+00	1.86e+01±2.46e+00
MFB3	2.87e+00±5.52e-01	4.93e+00±9.77e-01	2.85e+00±7.38e-01	5.08e+00±7.95e-01	1.96e+01±2.14e+00
MFB4	2.76e+00±6.49e-01	5.08e+00±9.04e-01	2.80e+00±6.49e-01	4.99e+00±1.06e+00	2.05e+01±2.93e+00
MFB5	3.36e+00±6.97e-01	4.86e+00±1.09e+00	3.24e+00±8.25e-01	4.72e+00±1.07e+00	1.86e+01±2.21e+00
MFB6	4.11e+00±1.02e+00	5.10e+00±9.89e-01	4.21e+00±8.44e-01	4.90e+00±1.22e+00	2.04e+01±2.33e+00
MFB7	2.84e+00±6.69e-01	5.13e+00±1.22e+00	2.42e+00±6.30e-01	5.19e+00±8.19e-01	1.95e+01±2.08e+00
MFB8	4.53e+00±8.89e-01	5.02e+00±1.09e+00	4.01e+00±9.89e-01	5.12e+00±9.09e-01	1.94e+01±2.95e+00
MFB9	3.49e+00±6.03e-01	5.48e+00±1.03e+00	3.33e+00±5.39e-01	5.32e+00±1.34e+00	2.01e+01±3.07e+00
MFB10	3.22e+00±6.14e-01	5.21e+00±1.21e+00	3.08e+00±7.41e-01	5.27e+00±1.03e+00	2.01e+01±2.14e+00
MFB11	2.84e+00±7.78e-01	4.87e+00±1.13e+00	2.79e+00±5.18e-01	5.14e+00±1.04e+00	1.99e+01±1.82e+00
MFB12	2.93e+00±7.84e-01	5.10e+00±9.23e-01	3.07e+00±6.93e-01	5.35e+00±1.10e+00	1.99e+01±2.50e+00
MFB13	2.78e+00±6.47e-01	5.10e+00±8.78e-01	4.60e+00±1.18e+00	4.82e+00±1.04e+00	1.93e+01±2.16e+00

fitness values using different single fidelity levels in PSO-AFAh and PSO-AFAg on MFB1 in Fig. 18. It is clear that PSO-AFAh and PSO-AFAg allocate computational budget in a different way. For the first fidelity level, PSO-AFAh consumes more computational budget than PSO-AFAg for building the logic regression models. They use at first four fidelity levels with a small amount of computational cost to approach the region near the global optimum. When $\phi = 4000$, PSO-AFAh and PSO-AFAg achieve the same performance of PSO with the whole computational budget. Then, they use the rest budget for higher fidelity levels to refine the solution. In summary, both PSO-AFAh and PSO-AFAg are able to adaptively adjust the fidelity level based on the state of their swarm to minimize the computational cost.

of the thirteen MFB problems. Finally, AFAg is able to more effectively save more computational cost than AFAi.

It should be pointed out that the above experimental conclusions hold on the MFB problems. Further research is needed to better understand the search performance of the two fidelity control strategies. AFAi will be able to work better if more precise regression models are built, at the expense of more computational costs to train those models.

Note also that the proposed AFA strategies are based on very intuitive methodologies for fidelity adjustment and no surrogate model is employed. In other words, there is much more room for performance improvement in multi-fidelity optimization when surrogates are combined with simulations of various fidelity levels.

VI. CONCLUDING REMARK

Multi-fidelity optimization problems are commonly seen in simulation-driven optimization. So far, little work has been done in the evolutionary computational community with a few exceptions. This paper aims to design synthetic benchmark problems on the basis of the characteristics found in real-world multi-fidelity optimization problems. Three different types of errors, namely, resolution errors, stochastic errors and instability errors are considered, based on which thirteen generic, scalable, and extensible multi-fidelity benchmark (MFB) problems are formulated for multi-fidelity optimization. In addition, we proposed three fidelity adjustment strategies, which are embedded in a PSO algorithm and compared on the thirteen MFB problems. Our empirical results demonstrate that the generation-based fidelity adjustment strategy is effective in reducing computational time.

Although the proposed MFB problems aim to capture the main characteristics and reflect the challenges in real-world multi-fidelity optimization problems, many issues remain to be addressed in the future. For example, more rigorous comparative studies need to be carried out to verify to what an extent the proposed benchmark problems can resemble multi-fidelity simulations. Second, the proposed fidelity adjustment strategies do not take surrogates into account, which represent another level of approximation. Third, the correlation between the objective functions of different fidelity levels needs further

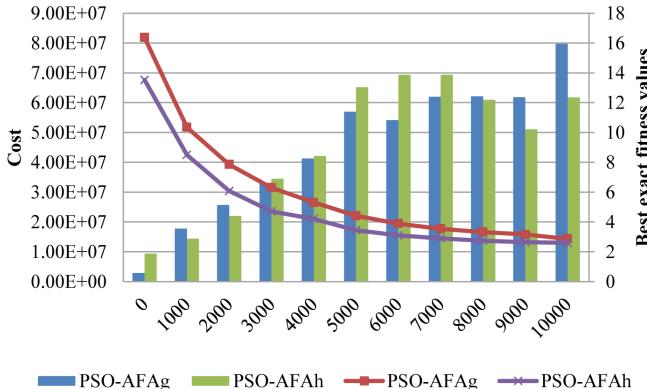


Fig. 18. Average cost and obtained exact fitness values in various fidelity levels in PSO-AFAh and PSO-AFAg on MFB1 ($d = 30$).

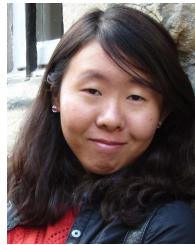
From the above experiments, the following observations can be made. First, adopting a single surrogate model for the highest-fidelity objective function only might be insufficient, because a single surrogate is not efficient to strike the optimal trade-off between the performance and computational cost. Second, using multi-fidelity models can save computational cost, provided that there is a correlation between the high fidelity simulations and low fidelity simulations. Third, the generation-based fidelity adjustment strategy (AFAg) and the individual-based fidelity adjustment strategy (AFAi) perform differently. Overall, AFAg outperforms AFAi on a majority

analysis for a more efficient adjustment of fidelity levels. Furthermore, there is hardly any study on the performance assessment of multi-fidelity optimization algorithms. A reasonable assessment system should evaluate both performance and computational cost. Finally, research on design of multi-objective multi-fidelity benchmark problems and algorithms will also be important for solving real-world multi-fidelity optimization problems.

REFERENCES

- [1] L. Leifsson and S. Koziel, "Variable-fidelity aerodynamic shape optimization," in *Computational Optimization, Methods and Algorithms*. Springer, 2011, pp. 179–210.
- [2] J. Branke, M. Asafuddoula, K. S. Bhattacharjee, and T. Ray, "Efficient use of partially converged simulations in evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 1, pp. 52–64, 2017.
- [3] Y. Cao, Y. Jin, M. Kowalczykiewicz, and B. Sendhoff, "Prediction of convergence dynamics of design performance using differential recurrent neural networks," in *International Joint Conference on Neural Networks*, 2008, pp. 529–534.
- [4] D. Lim, Y.-S. Ong, Y. Jin, and B. Sendhoff, "Evolutionary optimization with dynamic fidelity computational models," in *International Conference on Intelligent Computing*. Springer, 2008, pp. 235–242.
- [5] C. Smith, J. Doherty, and Y. Jin, "Recurrent neural network ensembles for convergence prediction in surrogate-assisted evolutionary optimization," in *Computational Intelligence in Dynamic and Uncertain Environments (CIDUE), 2013 IEEE Symposium on*. IEEE, 2013, pp. 9–16.
- [6] A. Mehmani, "Uncertainty-integrated surrogate modeling for complex system optimization." Ph.D. dissertation, Syracuse University, 2015.
- [7] S. Koziel, "Multi-fidelity multi-grid design optimization of planar microwave structures with sonnet," in *International Review of Progress in Applied Computational Electromagnetics*, 2010, pp. 26–29.
- [8] L. Leifsson and S. Koziel, "Multi-fidelity design optimization of transonic airfoils using physics-based surrogate modeling and shape-preserving response prediction," *Journal of Computational Science*, vol. 1, no. 2, pp. 98–106, 2010.
- [9] ———, "Aerodynamic shape optimization by variable-fidelity computational fluid dynamics models: A review of recent progress," *Journal of Computational Science*, vol. 10, pp. 45–54, 2015.
- [10] D. Peri and E. F. Campana, "High-fidelity models and multiobjective global optimization algorithms in simulation-based design," *Journal of Ship Research*, vol. 49, no. 3, pp. 159–175, 2005.
- [11] D. J. Toal, A. J. Keane, D. Benito, J. A. Dixon, J. Yang, M. Price, T. Robinson, A. Remouchamps, and N. Kill, "Multifidelity multidisciplinary whole-engine thermomechanical design optimization," *Journal of Propulsion and Power*, vol. 30, no. 6, pp. 1654–1666, 2014.
- [12] K. Yu, X. Yang, and Z. Mo, "Profile design and multifidelity optimization of solid rocket motor nozzle," *Journal of Fluids Engineering*, vol. 136, no. 3, pp. 1–6, 2014.
- [13] Y. Jin and B. Sendhoff, "A systems approach to evolutionary multiobjective structural optimization and beyond," *Computational Intelligence Magazine, IEEE*, vol. 4, no. 3, pp. 62–76, 2009.
- [14] S. Choi, J. J. Alonso, I. M. Kroo, and M. Wintzer, "Multifidelity design optimization of low-boom supersonic jets," *Journal of Aircraft*, vol. 45, no. 1, pp. 106–118, 2008.
- [15] S. Atamturktur, Z. Liu, S. Cogan, and H. Juang, "Calibration of imprecise and inaccurate numerical models considering fidelity and robustness: a multi-objective optimization-based approach," *Structural and Multidisciplinary Optimization*, vol. 51, no. 3, pp. 659–671, 2015.
- [16] X. Marduel, C. Tribes, and J.-Y. Trépanier, "Variable-fidelity optimization: efficiency and robustness," *Optimization and Engineering*, vol. 7, no. 4, pp. 479–500, 2006.
- [17] R. Sala, N. Baldanzini, and M. Pierini, "Representative surrogate problems as test functions for expensive simulators in multidisciplinary design optimization of vehicle structures," *Structural and Multidisciplinary Optimization*, vol. 54, no. 3, pp. 449–468, 2016.
- [18] J. Ren, A. Thelen, A. Amrit, X. Du, L. Leifsson, Y. Tesfahunegn, and S. Koziel, "Application of multi-fidelity optimization techniques to benchmark aerodynamic design problems," in *54th AIAA Aerospace Sciences Meeting*, 2016, p. 1542.
- [19] A. I. Forrester, A. Sóbester, and A. J. Keane, "Multi-fidelity optimization via surrogate modelling," in *Proceedings of the royal society of london a: mathematical, physical and engineering sciences*, vol. 463, no. 2088. The Royal Society, 2007, pp. 3251–3269.
- [20] N. M. Alexandrov, R. M. Lewis, C. R. Gumbert, L. L. Green, and P. A. Newman, "Optimization with variable-fidelity models applied to wing design," DTIC Document, Tech. Rep., 1999.
- [21] ———, "Approximation and model management in aerodynamic optimization with variable-fidelity models," *Journal of Aircraft*, vol. 38, no. 6, pp. 1093–1101, 2001.
- [22] J. Zheng, H. Qiu, and H. Feng, "The variable fidelity optimization for simulation-based design: a review," in *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2012.
- [23] M. K. Zahir and Z. Gao, "An improved parallel optimization framework for transonic airfoil design," *Reseach Journal of Applied Sciences, Engineering and Technology*, vol. 5, no. 22, pp. 5209–5216, 2013.
- [24] Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," *Soft Computing*, vol. 9, no. 1, pp. 3–12, 2005.
- [25] ———, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011.
- [26] Y. Xiong, W. Chen, and K.-L. Tsui, "A new variable-fidelity optimization framework based on model fusion and objective-oriented sequential sampling," *Journal of Mechanical Design*, vol. 130, no. 11, pp. 1–10, 2008.
- [27] S. E. Gano, *Simulation-based design using variable fidelity optimization*. World Scientific, 2005.
- [28] M. Eldred and D. Dunlavy, "Formulations for surrogate-based optimization with data fit, multifidelity, and reduced-order models," in *Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, number AIAA-2006-7117, Portsmouth, VA, vol. 199, 2006.
- [29] A. Diaz-Manríquez, G. Toscano, and C. A. C. Coello, "Comparison of metamodeling techniques in evolutionary algorithms," *Soft Computing*, vol. 21, no. 16, pp. 5647–5663, 2017.
- [30] H. Wang, Y. Jin, and J. Doherty, "Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2664–2677, 2017.
- [31] S. Ulaganathan, I. Couckuyt, F. Ferranti, T. Dhaene, and E. Laermans, "Variable-fidelity surrogate modelling with kriging," in *17th International Conference on Chemistry and Chemical Engineering, World Academy of Science, Engineering and Technology, International Science Index, Computer and Information Engineering (ICCCE 2015)*, 2015, pp. 514–518.
- [32] G. Sun, G. Li, M. Stone, and Q. Li, "A two-stage multi-fidelity optimization procedure for honeycomb-type cellular materials," *Computational Materials Science*, vol. 49, no. 3, pp. 500–511, 2010.
- [33] Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, and K. Y. Lum, "Combining global and local surrogate models to accelerate evolutionary optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 1, pp. 66–76, 2007.
- [34] S. Koziel, S. Ogurtsov, and S. Szczepanski, "Local response surface approximations and variable-fidelity electromagnetic simulations for computationally efficient microwave design optimisation," *IET microwaves, antennas & propagation*, vol. 6, no. 9, pp. 1056–1062, 2012.
- [35] Q. Zhou, X. Shao, P. Jiang, H. Zhou, and L. Shu, "An adaptive global variable fidelity metamodeling strategy using a support vector regression based scaling function," *Simulation Modelling Practice and Theory*, vol. 59, pp. 18–35, 2015.
- [36] Y. Jin, M. Olhofer, and B. Sendhoff, "A framework for evolutionary optimization with approximate fitness functions," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 5, pp. 481–494, 2002.
- [37] G. Sun, G. Li, S. Zhou, W. Xu, X. Yang, and Q. Li, "Multi-fidelity optimization for sheet metal forming process," *Structural and Multidisciplinary Optimization*, vol. 44, no. 1, pp. 111–124, 2011.
- [38] S. Ulaganathan, I. Couckuyt, F. Ferranti, E. Laermans, and T. Dhaene, "Performance study of multi-fidelity gradient enhanced kriging," *Structural and Multidisciplinary Optimization*, vol. 51, no. 5, pp. 1017–1033, 2015.
- [39] L. Le Gratiet, "Bayesian analysis of hierarchical multifidelity codes," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 1, no. 1, pp. 244–269, 2013.
- [40] D. J. Toal, "Some considerations regarding the use of multi-fidelity kriging in the construction of surrogate models," *Structural and Multidisciplinary Optimization*, vol. 51, no. 6, pp. 1223–1245, 2015.

- [41] Z.-H. Han, S. Görtz, and R. Zimmermann, "Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function," *Aerospace Science and Technology*, vol. 25, no. 1, pp. 177–189, 2013.
- [42] S. E. Gano, J. E. Renaud, J. D. Martin, and T. W. Simpson, "Update strategies for kriging models used in variable fidelity optimization," *Structural and Multidisciplinary Optimization*, vol. 32, no. 4, pp. 287–298, 2006.
- [43] D. Huang, T. Allen, W. Notz, and R. Miller, "Sequential kriging optimization using multiple-fidelity evaluations," *Structural and Multidisciplinary Optimization*, vol. 32, no. 5, pp. 369–382, 2006.
- [44] Z.-H. Han, R. Zimmermann, and S. Görtz, "A new cokriging method for variable-fidelity surrogate modeling of aerodynamic data," *AIAA Paper*, vol. 1225, no. 2010, pp. 1–22, 2010.
- [45] N. Courrier, P.-A. Boucard, and B. Soulier, "Variable-fidelity modeling of structural analysis of assemblies," *Journal of Global Optimization*, vol. 64, no. 3, pp. 577–613, 2016.
- [46] C. Smith, J. Doherty, and Y. Jin, "Multi-objective evolutionary recurrent neural network ensemble for prediction of computational fluid dynamic simulations," in *IEEE Congress on Evolutionary Computation*, 2014.
- [47] T. Robinson, K. Willcox, M. Eldred, and R. Haimes, "Multifidelity optimization for variablecomplexity design," in *Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Portsmouth, VA, 2006.
- [48] A. Chaudhuri and K. Willcox, "Multifidelity uncertainty propagation in coupled multidisciplinary systems," in *18th AIAA Non-Deterministic Approaches Conference*, 2016, p. 1442.
- [49] E. Li, H. Wang, and F. Ye, "Two-level multi-surrogate assisted optimization method for high dimensional nonlinear problems," *Applied Soft Computing*, vol. 46, pp. 26–36, 2016.
- [50] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: a survey," *Control Engineering Practice*, vol. 10, no. 11, pp. 1223–1241, 2002.
- [51] B. Liu, S. Koziel, and Q. Zhang, "A multi-fidelity surrogate-model-assisted evolutionary algorithm for computationally expensive optimization problems," *Journal of Computational Science*, vol. 12, pp. 28–37, 2016.
- [52] W. L. Oberkampf and T. G. Trucano, "Verification and validation in computational fluid dynamics," *Progress in Aerospace Sciences*, vol. 38, no. 3, pp. 209–272, 2002.
- [53] K. McCarthy, P. McCarthy, N. Wu, A. Alleyne, J. Koeln, S. Patnaik, S. Emo, and J. Cory, "Model accuracy of variable fidelity vapor cycle system simulations," SAE Technical Paper, Tech. Rep., 2014.
- [54] H. Wang, Y. Jin, and J. O. Jansen, "Data-driven surrogate-assisted multiobjective evolutionary optimization of a trauma system," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 6, pp. 939–952, 2016.
- [55] B. D. Ripley, *Stochastic simulation*. John Wiley & Sons, 2009, vol. 316.
- [56] S. T. Rachev, S. V. Stoyanov, and F. J. Fabozzi, *Advanced stochastic models, risk assessment, and portfolio optimization: The ideal risk, uncertainty, and performance measures*. John Wiley & Sons, 2008, vol. 149.
- [57] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *The Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.
- [58] A. Slepoy, A. P. Thompson, and S. J. Plimpton, "A constant-time kinetic monte carlo algorithm for simulation of large biochemical reaction networks," *The Journal of Chemical Physics*, vol. 128, no. 20, pp. 1–8, 2008.
- [59] F. F. Abraham, D. Brodbeck, R. Rafey, and W. Rudge, "Instability dynamics of fracture: a computer simulation investigation," *Physical Review Letters*, vol. 73, no. 2, pp. 272–272, 1994.
- [60] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *IEEE Congress on Evolutionary Computation*, 1998, pp. 69–73.
- [61] X. Chen, M. Diez, M. Kandasamy, Z. Zhang, E. F. Campana, and F. Stern, "High-fidelity global optimization of shape design by dimensionality reduction, metamodels and deterministic particle swarm," *Engineering Optimization*, vol. 47, no. 4, pp. 473–494, 2015.
- [62] K. Miettinen, *Nonlinear multiobjective optimization*. Springer, 1999.
- [63] B. Liu, Q. Zhang, and G. G. Gielen, "A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 180–192, 2014.
- [64] J. S. Kowalik and M. R. Osborne, *Methods for unconstrained optimization problems*. North-Holland, 1968.



Handing Wang (S'10-M'16) received the B.Eng. and Ph.D. degrees from Xidian University, Xi'an, China, in 2010 and 2015, respectively.

She is currently a research fellow with the Department of Computer Science, University of Surrey, Guildford, UK.

Dr. Wang is a member of IEEE Computational Intelligence Society. Her research interests include nature-inspired computation, multiobjective optimization, multiple criteria decision making, surrogate-assisted evolutionary optimization, and real-world problems.



Yaochu Jin (M'98-SM'02-F'16) received the B.Sc., M.Sc., and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1988, 1991, and 1996 respectively, and the Dr.-Ing. degree from Ruhr University Bochum, Germany, in 2001.

He is a Professor in Computational Intelligence, Department of Computer Science, University of Surrey, Guildford, U.K., where he heads the Nature Inspired Computing and Engineering Group. He is also a Finland Distinguished Professor funded by the Finnish Agency for Innovation and a Changjiang

Distinguished Visiting Professor appointed by the Ministry of Education, China. His science-driven research interests lie in the interdisciplinary areas that bridge the gap between computational intelligence, computational neuroscience, and computational biology. He is also particularly interested in real-world problem solving using artificial intelligence and machine learning, including data-driven optimization, image identification, and interpretable and secure machine learning. He has (co)authored over 200 peer-reviewed journal and conference papers and been granted eight patents on evolutionary optimization. His research has been funded by EC FP7, UK EPSRC and international companies. He has delivered over 30 invited keynote speeches at international conferences.

He is the Editor-in-Chief of the IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS and Complex & Intelligent Systems. He is also an Associate Editor or Editorial Board Member of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON NANOBIO-SCIENCE, Evolutionary Computation, BioSystems, Soft Computing, and Natural Computing.

Dr Jin is an IEEE Distinguished Lecturer (2017–2019) and was the Vice President for Technical Activities of the IEEE Computational Intelligence Society (2014–2015). He is the recipient of the 2014 and 2016 IEEE Computational Intelligence Magazine Outstanding Paper Award, the 2017 IEEE Transactions on Evolutionary Computation Outstanding Paper Award, and the Best Paper Award of the 2010 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology. He is a Fellow of IEEE.



John Doherty received a B.Sc. from the University of Liverpool in 1988.

He is currently a Reader within the Centre for Aerodynamics and Environmental Flow, Department of Mechanical Engineering Sciences, University of Surrey, Guildford, U.K.

His research interests include aerodynamic and multi-disciplinary optimization under uncertainty, for aircraft, vehicle and propulsion installation design. He is a Fellow of the Royal Aeronautical Society.