sensors

MDPI

Article

An Improved Multioperator-Based Constrained Differential Evolution for Optimal Power Allocation in WSNs

Wei Li and Wenyin Gong *

School of Computer Science, China University of Geosciences, Wuhan 430074, China; liwei2@cug.edu.cn
* Correspondence: wygong@cug.edu.cn

Abstract: Optimal power allocation (OPA), which can be transformed into an optimization problem with constraints, plays a key role in wireless sensor networks (WSNs). In this paper, inspired by ant colony optimization, an improved multioperator-based constrained adaptive differential evolution (namely, IMO-CADE) is proposed for the OPA. The proposed IMO-CADE can be featured as follows: (i) to adaptively select the proper operator among different operators, the feedback of operators and the status of individuals are considered simultaneously to assign the selection probability; (ii) the constrained reward assignment is used to measure the feedback of operators; (iii) the parameter adaptation is used for the parameters of differential evolution. To extensively evaluate the performance of IMO-CADE, it is used to solve the OPA for both the independent and correlated observations with different numbers of sensor nodes. Compared with other advanced methods, simulation results clearly indicate that IMO-CADE yields the best performance on the whole. Therefore, IMO-CADE can be an efficient alternative for the OPA of WSNs, especially for WSNs with a large number of sensor nodes.

Keywords: wireless sensor networks; optimal power allocation; constrained differential evolution; multioperator adaptation; ant colony optimization

check for updates

Citation: Li, W.; Gong, W. An Improved Multioperator-Based Constrained Differential Evolution for Optimal Power Allocation in WSNs. *Sensors* **2021**, *21*, 6271. https://doi.org/10.3390/s21186271

Academic Editor: Ramon Gonzalez Carvajal

Received: 26 July 2021
Accepted: 14 September 2021
Published: 18 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 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) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Due to their cost-effectiveness, easy deployment, and intelligence, wireless sensor networks (WSNs) have obtained considerable attention in the last few decades [1]. Nowadays, WSN is one of the promising technologies for real-world applications [2,3]. To improve their performance and lifetime, various computational intelligence techniques have been used recently for the design of WSNs [4], such as particle swarm optimization (PSO) [5], genetic algorithms (GAs), ant colony optimization (ACO) [6], machine learning algorithms [7], lion optimization [8], negatively correlated search [9], and Fibonacci tree optimization algorithm [10].

In WSNs, there are usually many sensor nodes, which have a very limited power supply. Therefore, among various design issues, optimal power allocation (OPA) plays a key role in WSNs [11–15]. The OPA can be transformed into an optimization problem with constraints. Due to its nonlinear and computationally expensive (for the correlated observations) nature, the optimization of the transformed problem is difficult for traditional/analytical methods; especially, there are a large number of sensor nodes with correlated observations due to the expensive computation cost [16]. In the literature, the use of nature-inspired optimization methods for solving the OPA has received more consideration [6]. For example, in [17], PSO was used for the OPA. Boussaïd et al. [18] presented a constrained bio-geography-based optimization (BBO) and differential evolution (DE)—namely, CBBO-DE—for the OPA. In [19], combined with the penalty technique, three nature-inspired methods—cat swarm optimization (CSO), cuckoo search (CS), and PSO—were compared for solving the OPA. In [20], Tsiflikiotis et al. developed a hybrid method with teaching–learning-based optimization and Jaya algorithms. Lee compared three

swarm intelligence algorithms—PSO, artificial bee colony (ABC), and continuous ACO (ACO_R)—for the OPA [21]. In [22], a constrained DE with multiple operator selection (PM-MDE) was presented, where a transformed fitness function was used for the constraints.

As mentioned above, various nature-inspired algorithms has been presented to solve the OPA in WSNs. However, there is still much room to improve the performance for the OPA by developing other advanced optimization techniques, especially for the OPA with a large number of sensor nodes and/or with the correlated observations. Based on this consideration and inspired by our previous work [22], in this paper, we propose an improved multioperator-based constrained adaptive DE, referred to as IMO-CADE, to try to effectively solve the OPA. IMO-CADE has the following three characteristics: (i) to adaptively select the proper operator among different operators, the feedback of operators and the status of individuals are considered simultaneously, such as the edge selection in ant colony optimization; (ii) the constrained reward assignment is used to measure the feedback of operators; (iii) the parameter adaptation is used for the parameters of DE. To evaluate the performance of IMO-CADE, extensive experiments have been conducted for the OPA on both the independent and correlated observations. Compared with other advanced methods, IMO-CADE can consistently provide the best results on the whole.

The rest of the paper is organized as follows: In Section 2, the background is briefly introduced, including the optimization problem formulation of the OPA, basic DE, and probability matching. The related work of nature-inspired optimization methods for the OPA are briefly reviewed in Section 3. Section 4 elaborates on the proposed IMO-CADE method. In Section 5, extensive simulations have been conducted to evaluate the performance of our proposal. Finally, Section 6 concludes the paper and points out some possible future work.

2. Background

In this section, first, the transformed optimization problem of the OPA is described. Then, the original DE is briefly introduced.

2.1. Problem Formulation

In this work, a WSN with a fusion center and K sensor nodes is considered. The optimization problem of the OPA can be formulated as follows (Since this paper mainly focuses on the development of efficient optimizer for the OPA, the data fusion problem formulation of the OPA in WSNs is not mentioned herein; the interested reader can refer to the work in [17] for the details.) [17,18]:

$$\min_{G_k \geq 0} f(\mathbf{x}) = \sum_{k=1}^{K} G_k^2, \tag{1}$$

subject to

$$\begin{aligned} &P(E) = Q\left(\tfrac{1}{2}\sqrt{m^2 \mathbf{e}^T \mathbf{A} \textstyle\sum_K^{-1} \mathbf{A}\mathbf{e}}\right) \leq \varepsilon, \\ &G_k \geq 0, \\ &k = 1, \cdots, K. \end{aligned} \tag{2}$$

where ε is the required fusion error probability threshold, m indicates the deterministic signal, G_k is the amplifier gain at node k, and $\mathbf{e}$ is the L-length vector with all ones. The covariance matrix is $\sum_K = \mathbf{A}^T \sum_v \mathbf{A} + \sum_w$, where $\mathbf{A} = diag(H_1 G_1, \ldots, H_K G_K)$, $\sum_v$, and $\sum_w$ are the observation and receiver noise covariances, respectively. H_k ($k = 1, \cdots, K$) is the channel fading coefficient. In this work, two situations (i.i.d. and correlated observations) are considered.

2.1.1. Independent Observations

If the local observations and the receiver noise are both i.i.d., the probability of fusion error can be simplified to

$$P(E) = Q\left(\frac{m}{2}\sqrt{\sum_{k=1}^{K}\frac{H_k^2 G_k^2}{\delta_v^2 H_k^2 G_k^2 + \delta_\omega^2}}\right) \leq \varepsilon. \tag{3}$$

The inequality in (2) can be expressed as follows:

$$\beta \leq \sqrt{\sum_{k=1}^{K}\frac{H_k^2 G_k^2}{\delta_v^2 H_k^2 G_k^2 + \delta_\omega^2}}, \tag{4}$$

where $\beta = \frac{2}{m}Q^{-1}(\varepsilon)$ and $Q(\cdot)$ is the complementary Gaussian cumulative distribution function. δ_v means the variances of the observation noise and δ_w represents the receiver noise. In this work, the channel fading coefficient H_k follows an exponential distribution (i.e., Rayleigh fading) with unit mean [18].

2.1.2. Correlated Observations

If the sensor observations are spatially correlated, the observation noise covariance matrix $\sum_v$ with the correlation degree ρ can be formulated as follows:

$$\sum\nolimits_v = \delta_v^2 \begin{pmatrix} 1 & \rho^d & \cdots & \rho^{d(K-2)} & \rho^{d(K-1)} \\ \rho^d & 1 & \cdots & \rho^{d(K-3)} & \rho^{d(K-2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho^{d(K-1)} & \rho^{d(K-2)} & \cdots & \rho^d & 1 \end{pmatrix}. \tag{5}$$

The inequality in (2) is

$$\beta \leq \sqrt{\mathbf{e}^T \mathbf{A} \sum\nolimits_K^{-1} \mathbf{A}\mathbf{e}}, \tag{6}$$

where $d_j = d(j-1), j = 1, \ldots K$, which means the sensor nodes are equally spaced along a straight line. Since $\sum_v$ is not diagonal, it is difficult to evaluate $\sum_K^{-1}$ in closed form. Note that, under the correlated situation, the computational complexity for the calculation of Equation (5) is $O(K^2)$, which means that the OPA with correlated observations is computationally expensive when K is large.

2.2. Differential Evolution

DE is a simple and efficient global optimizer for numerical optimization problems [23]. The flowchart of the original DE is shown in Figure 1, where there are four main procedures—i.e., population initialization, differential mutation, crossover, and selection.

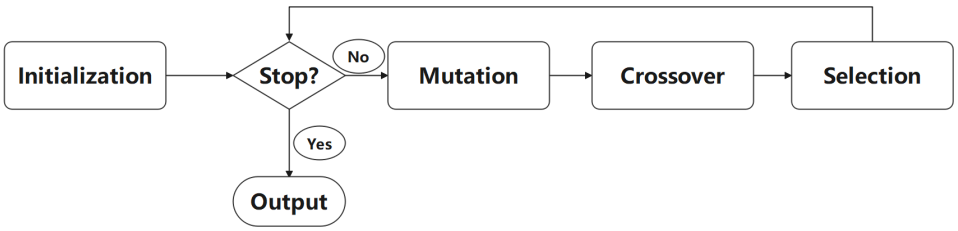


Figure 1. The flowchart of DE.

2.2.1. Initialization

Suppose the population contains Np solutions. For the OPA, each solution $\mathbf{x}^i = (x_1^i, \cdots, x_K^i)$, $i = 1 \cdots, Np$, is a K-dimensional real-valued vector. Initially, each variable $x_{i,k}$ is randomly generated as

$$x_k^i = \text{rndreal}(0, 10), \tag{7}$$

where $i = 1 \cdots, Np$ and $k = 1, \cdots, K$. The lower and upper bounds of x_k^i are 0 and 10, respectively. rndreal$(0, 10)$ is a real-valued random number generator from $(0, 10)$. Note that, for the OPA, x_k^i represents the amplifier gain at node k, i.e., G_k.

2.2.2. Mutation

In DE, the core operator is the *differential mutation*. The commonly used "DE/rand/1" can be formulated as

$$\mathbf{v}^i = \mathbf{x}^{r1} + F(\mathbf{x}^{r2} - \mathbf{x}^{r3}), \tag{8}$$

where $\mathbf{v}^i$ is generated mutant vector; $r1, r2, r3 \in \{1, Np\}$ satisfy $r1 \neq r2 \neq r3 \neq i$, and $F \in (0, 1)$ is a scaling factor.

2.2.3. Crossover

After mutation, the crossover is performed between the target vector $\mathbf{x}^i$ and the mutant vector $\mathbf{v}^i$. The binomial crossover is

$$u_k^i = \begin{cases} v_k^i, & \text{if rndreal}(0, 1) \leq Cr \text{ or } rand_j == k \\ x_k^i, & \text{otherwise} \end{cases}, \tag{9}$$

where $\mathbf{u}^i$ is the trial vector, $rand_j \in \{i, K\}$ is a randomly generated integer, and $Cr \in [0, 1]$ is the crossover rate.

2.2.4. Selection

After generating the trial vector, DE performs the one-to-one tournament selection to select the solution to survive into the next generation:

$$\mathbf{x}^i = \begin{cases} \mathbf{u}^i, & \text{if } \mathbf{u}^i \text{ is better than } \mathbf{x}^i \\ \mathbf{x}^i, & \text{otherwise} \end{cases}. \tag{10}$$

The mutation, crossover, and selection will repeat until the termination criterion is met.

3. Related Work

As stated in [17], if the observations are i.i.d, the OPA can be derived analytically in closed form. However, if the observations are spatially correlated, the evaluation of $\sum_K^{-1}$ in (6) cannot be obtained analytically in closed form. Therefore, recently, various nature-inspired methods are used for the OPA.

In [17,24], PSO was employed to obtain the OPA with the constraint of a required fusion error probability threshold. To deal with the constraint, the exterior penalty function with two additional parameters (i.e., positive penalty parameter r_k and non-negative constant q) is used. In [18], Boussaïd et al. hybridized BBO with DE for the OPA, where the penalty function presented in [25] was used to handle the constraint. In [19], with the dynamically modified penalty functions, CSO, CS, and PSO were empirically compared for the OPA. The results indicated that (i) when the number of the sensor nodes is small, PSO gets better results; however, (ii) CSO outperforms CS and PSO when the network consists of a large number of sensors. Tsiflikiotis et al. [20] extended their previous work [19] and presented a hybrid TLBO and Jaya (namely, TLBO-Jaya) to solve the OPA, where the proposed TLBO-Jaya was compared with TLBO, Jaya, heat transfer search algorithm, PSO, and BBO-DE. In [21], Lee used PSO, ABC, and ACO_R for optimizing the OPA, where Deb's three feasibility rules [26] were used to replace the penalty function for the constraint

handling. Li et al. [22] presented a constrained DE with multiple mutation strategies, where probability matching and constrained credit assignment techniques were used.

The abovementioned methods obtained promising results for solving the OPA. However, most of them are used to solve the OPA with a small number of sensor nodes, e.g., $K \leq 50$. In addition, the penalty functions usually introduce new parameters, which may influence the performance significantly. Therefore, there is still much room to develop advanced, effective, and efficient numerical methods for the OPA, especially for the OPA with a large number of sensor nodes.

4. Our Approach: IMO-CADE

With the main objective of finding an alternative solution for minimizing the total power consumed by the WSN, in this section, the proposed IMO-CADE is introduced in detail, including the motivations, operator pool, constrained reward assignment, improved multioperator selection technique, and parameter adaptation.

4.1. Motivations

As mentioned in Sections 1 and 3, different nature-inspired methods are presented to solve the OPA. To deal with the constraints, most methods use penalty functions. However, it is difficult to set the penalty coefficients for different problems. In [21], it adopts the feasibility rules [26]. However, the feasibility rules favor the feasible solutions, which may result in trapping into the local feasible regions.

In [27], the empirical study stated that the performance of DE may significantly be improved by the mutation operators for the OPA. It is difficult to choose the most suitable mutation operators for different cases in the OPA. In our previous work [22], probability matching was used for operator selection. However, to assign the probabilities of different operators, only operator feedback is considered. Indeed, individual information should also be considered for the probability assignment.

Based on the above considerations, in this work, we propose an improved multi-operator selection technique by considering both the operator feedback and individual information. In addition, to further improve the performance, parameter adaptation of DE is also used to release the setting of parameters by the users. The main techniques used in IMO-CADE are elaborated in the following subsections.

4.2. Operator Pool

In the DE literature, various mutation operators have been proposed [28]. In this work, to efficiently solve the OPA, two mutation operators presented in [29,30] were chosen as the operator pool—"DE/current-to-*p*best/1" with archive and "DE/rand-to-*p*best/1" with archive:

- "DE/current-to-*p*best/1" with archive:

$$\mathbf{v}^i = \mathbf{x}^i + F^i(\mathbf{x}^{pbest} - \mathbf{x}^i) + F^i(\mathbf{x}^{r1} - \hat{\mathbf{x}}^{r2}), \tag{11}$$

- "DE/rand-to-*p*best/1" with archive:

$$\mathbf{v}^i = \mathbf{x}^{r0} + F^i(\mathbf{x}^{pbest} - \mathbf{x}^{r0}) + F^i(\mathbf{x}^{r1} - \hat{\mathbf{x}}^{r2}), \tag{12}$$

where $r0, r1, r2 \in \{1, Np\}$ and $r0 \neq r1 \neq r2 \neq i$. F^i is the scaling factor of the i-th solution; $\mathbf{x}^{pbest}$ is the "*p*best" solution randomly chosen from the top p% solutions of the current population; $\hat{\mathbf{x}}^{r2}$ is a randomly selected solution from the union of the current population $\mathbb{P}$ and the archive $\mathbb{A}$.

4.3. Boundary Constraint Handling

After generating the mutant vector with the mutation operator, some variables may be out of their boundary constraints, i.e., $v_k^i \notin [0, 10]$. In this situation, the following boundary constraint handling technique (BCHT) is used:

$$v_k^i = \begin{cases} (0 + x_k^i)/2, & \text{if } v_k^i < 0 \\ (10 + x_k^i)/2, & \text{if } v_k^i > 10 \\ v_i^k, & \text{otherwise} \end{cases}. \tag{13}$$

4.4. Constrained Reward Assignment

For solving the constrained optimization problems, there are three situations of the *combined* parent population $\mathbb{P}$ and child population $\mathbb{C}$.

(i) *Infeasible situation*: All solutions in $\mathbb{P} \cup \mathbb{C}$ are infeasible. Under this situation, the fitness $\mathcal{F}(\mathbf{x})$ of each solution $\mathbf{x}$ is its overall constraint violation (CV),

$$\mathcal{F}(\mathbf{x}) = CV(\mathbf{x}) = P(E) - \epsilon. \tag{14}$$

(ii) *Semifeasible situation*: $\mathbb{P} \cup \mathbb{C}$ contains both the infeasible and feasible solutions. In this situation, the solutions in the parent and child populations are combined. Then, for each solution, its objective function and CV are normalized as suggested in [31]. Afterwards, the fitness is as follows:

$$\mathcal{F}(\mathbf{x}) = f_{\text{nor}}(\mathbf{x}) + CV_{\text{nor}}(\mathbf{x}), \tag{15}$$

where $f_{\text{nor}}(\mathbf{x})$ and $CV_{\text{nor}}(\mathbf{x})$ are the normalized objective function and CV, respectively. The details can be found in [31].

(iii) *Feasible situation*: all solutions in $\mathbb{P} \cup \mathbb{C}$ are feasible. In this situation, the fitness is the objective function:

$$\mathcal{F}(\mathbf{x}) = f(\mathbf{x}) = \sum_{k=1}^{K} G_k^2. \tag{16}$$

According to the above fitness of each solution under situations, the relative fitness improvement is calculated as [22]

$$\hat{\mathcal{F}}^i = \frac{\mathcal{F}^{best}}{\mathcal{F}(\mathbf{u}^i)} \left(\mathcal{F}(\mathbf{x}^i) - \mathcal{F}(\mathbf{u}^i) \right), \tag{17}$$

where $i = 1, \cdots, Np$, $\mathcal{F}^{best}$ is the fitness of the best-so-far solution, $\mathbf{x}^i$ is the target solution, and $\mathbf{u}^i$ is its corresponding trial solution.

Then, based on the relative fitness improvement, the reward $\mathcal{R}_o$ of the o-th operator is calculated by

$$\mathcal{R}_o = \frac{\sum_{i=1}^{|\mathbb{S}_o|} \mathbb{S}_o(i)}{|\mathbb{S}_o|}, \tag{18}$$

where $\mathbb{S}_o$ is the set of all relative fitness improvements $\hat{\mathcal{F}}^i$ of an operator $o (o = 1, \cdots, N_o)$, N_o is the number of operators in the pool, $|\mathbb{S}_o|$ is the size of $\mathbb{S}_o$, and $\mathbb{S}_o(i)$ is the i-th relative fitness improvement of operator o saved in $\mathbb{S}_o$.

4.5. Improved Multioperator Selection

In order to adaptively select a suitable operator from the pool, the selection probability of each operator is assigned. In this work, both the operator feedback and individual information are considered simultaneously.

4.5.1. Probability Based on Operator Feedback

The operator feedback is measured by its constrained reward, as shown in Equation (18). Based on the reward, probability matching [32] is used to assign the probability of each operator. First, for each operator o, its quality is updated as

$$q_o = q_o + \alpha\big(\mathcal{R}_o - q_o\big), \tag{19}$$

where $\alpha \in (0, 1]$ is the adaptation rate. Then, the probability is calculated as

$$p'_o = p_{\min} + (1 - N_o \cdot p_{\min}) \frac{q_o}{\sum_{i=1}^{N_o} q_i}, \tag{20}$$

where $p_{\min} \in (0, 1)$ is the minimal probability of each operator.

4.5.2. Probability Based on Individual Information

Generally, for different solutions in the population, different mutation operators may be more suitable. In this work, two mutation operators, i.e., "DE/current-to-$p$best/1" with archive ($o = 1$) and "DE/rand-to-$p$best/1" with archive ($o = 2$), are used in the pool. "DE/current-to-$p$best/1" with archive performs the local search around the target solution, and hence, is more suitable to the *better* solutions. "DE/rand-to-$p$best/1" with archive is able to provide more diversity, and hence, is more suitable to the *worse* solutions. Based on this consideration, first, the solutions in $\mathbb{P}$ are sorted based on the feasibility rules [26] from the best to the worst. Then, according to the individual information, the probability is assigned as

$$p''_{o,i} = \begin{cases} 0.9, & \text{if } 1 \le i \le Np/2 \ \& \ o == 1 \\ 0.1, & \text{if } 1 \le i \le Np/2 \ \& \ o == 2 \\ 0.1, & \text{if } Np/2 < i \le Np \ \& \ o == 2 \\ 0.9, & \text{if } Np/2 < i \le Np \ \& \ o == 1, \end{cases} \tag{21}$$

where $p''_{o,i}$ is the probability for the o-th operator of the i-th solution, $o = 1, 2$, and $i = 1, \cdots, Np$.

4.5.3. Final Probability Calculation

Finally, inspired by ACO [33], by combining operator feedback and individual information, the probability is calculated by

$$p_{o,i} = \frac{p'_o \cdot p''_{o,i}}{\sum_{j=1}^{N_o} p'_j \cdot p''_{j,i}}. \tag{22}$$

4.6. Parameter Adaptation

To further improve the performance of IMO-CADE for the OPA (Algorithm 1), the parameter adaptation technique proposed in [29] is also used. Initially, for each target solution $\mathbf{x}^i$, its associated parameters Cr^i and F^i are generated as

$$Cr^i = \text{Gaussian}(\mu_{Cr}, 0.1), \tag{23}$$

and

$$F^i = \text{Cauchy}(\mu_F, 0.1), \tag{24}$$

where "Gaussian$(\mu_{Cr}, 0.1)$" is a Gaussian random number generator with mean μ_{Cr} and standard deviation 0.1, and "Cauchy$(\mu_F, 0.1)$" is a Cauchy random number generator with location μ_F and scale 0.1. At the first generation, $\mu_{Cr} = 0.5$ and $\mu_F = 0.5$.

Algorithm 1 Pseudo-code of IMO-CADE

Input: Algorithm parameters: $Np, \mu_{Cr}, \mu_F, c, p_{\min}, \alpha, NFEs_{\max}$; WSN parameters: $K, \epsilon, \rho, \gamma_0$
Output: The best solution $\mathbf{x}^{\text{best}} = \{G_1^{\text{best}}, \cdots, G_K^{\text{best}}\}$

1: Initialize the population $\mathbb{P}$ with Np solutions;
2: Evaluate the population $\mathbb{P}$ and set $NFEs = Np$;
3: For each operator, set $q_o = 0$ and $p'_o = 1/2$;
4: Set $\mathbb{A} = \Phi$;
5: **while** $NFEs < NFEs_{\max}$ **do**
6: Set $\mathbb{S}_{Cr} = \Phi$, $\mathbb{S}_F = \Phi$;
7: Sort the solutions in $\mathbb{P}$ based on the feasibility rules from the best to the worst;
8: For each solution in $\mathbb{P}$, set $p''_{o,i}$ according to Equation (21);
9: For each solution, calculate $p_{o,i}$ based on Equation (22);
10: **for** $i = 1$ to Np **do**
11: Select the operator o^i based on the selection probability;
12: Calculate Cr^i and F^i based on Equations (23) and (24), respectively;
13: Generate $\mathbf{u}^i$ and deal with the violated variables based on Equation (13);
14: Evaluate $\mathbf{u}^i$, $NFEs = NFEs + 1$, and $\mathbb{C} \leftarrow \mathbf{u}^i$;
15: **end for**
16: $\mathbb{M} \leftarrow \mathbb{P} \cup \mathbb{C}$;
17: For each solution $\mathbf{x}$ in $\mathbb{M}$, calculate $\mathcal{F}(\mathbf{x})$ under the current situation;
18: **for** $i = 1$ to Np **do**
19: **if** $\mathcal{F}(\mathbf{u}^i) \leq \mathcal{F}(\mathbf{x}^i)$ **then**
20: $\mathbb{A} \leftarrow \mathbf{x}^i$;
21: Calculate $\hat{\mathcal{F}}^i$ based on Equation (17) and save it into $\mathbb{S}_o(i)$;
22: $\mathbb{S}_{Cr} \leftarrow Cr^i$ and $\mathbb{S}_F \leftarrow F^i$;
23: Replace $\mathbf{x}^i$ with $\mathbf{u}^i$;
24: **else**
25: Set $\hat{\mathcal{F}}^i = 0$ and save it into $\mathbb{S}_o(i)$;
26: **end if**
27: **end for**
28: Update μ_{Cr} and μ_F;
29: **for** $o = 1$ to 2 **do**
30: Calculate $\mathcal{R}_o$ based on Equation (18);
31: Update p'_o based on Equation (20);
32: **end for**
33: **end while**

After that, if the trial solution $\mathbf{u}^i$ is better than its target solution $\mathbf{x}^i$, the parameters Cr_i and F_i are saved in $\mathbb{S}_{Cr}$ and $\mathbb{S}_F$, respectively. Subsequently, μ_{Cr} and μ_F are updated as follows:

$$\mu_{Cr} = (1 - c) \cdot \mu_{Cr} + c \cdot \text{mean}_A(\mathbb{S}_{Cr}), \tag{25}$$

and

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}_L(\mathbb{S}_F), \tag{26}$$

where $c = 0.1$ is a parameter, $\text{mean}_A(\mathbb{S}_{Cr})$ calculates the arithmetic mean of $\mathbb{S}_{Cr}$, and $\text{mean}_L(\mathbb{S}_F)$ calculates the Lehmer mean of $\mathbb{S}_F$ [29].

4.7. Framework of IMO-CADE

By integrating the abovementioned techniques, the pseudo-code of IMO-CADE is shown in Algorithm 1, where $NFEs$ is the current number of fitness evaluations and $NFEs_{\max}$ is the maximal $NFEs$. First, the population $\mathbb{P}$ is initialized randomly; then, each solution is evaluated. The initial quality of each operator is set to $q_o = 0$ and $p'_o = 1/2$. The archive $\mathbb{A}$ is set to be empty. In the main loop, it works as follows:

- At each generation, $\mathbb{S}_{Cr}$ and $\mathbb{S}_F$ are set to be empty.
- In lines 8–9, the selection probability $p_{o,i}$ is calculated.
- In line 11, for each solution, one operator o^i is selected based on the selection probability and roulette wheel selection.

- In lines 12–14, the trial solution is generated according to the selected operator and the generated parameters. The violated variables are handled based on the BCHT.
- In lines 16–17, the parent population $\mathbb{P}$ and the child population $\mathbb{C}$ are combined. Then, the feasibility situation is checked, followed by the transformed fitness calculation by Equations (14)–(16).
- In line 19, the trial solution $\mathbf{u}^i$ is compared with its target solution $\mathbf{x}^i$ based on the transformed fitness.
- If $\mathbf{u}^i$ is better than $\mathbf{x}^i$, in lines 20–22, the worse $\mathbf{x}^i$ is saved into archive $\mathbb{A}$. Note that, when $|\mathbb{A}| > Np$, $|\mathbb{A}| - Np$ solutions are randomly removed from $\mathbb{A}$ to keep $|\mathbb{A}| = Np$. The relative fitness improvement and the successful parameters Cr^i, F^i are saved.
- In line 23, $\mathbf{x}^i$ is replaced by $\mathbf{u}^i$ for the next generation.
- In lines 28–32, μ_{Cr}, μ_F, $\mathcal{R}_o$, and p'_o are updated accordingly.

Remarks

It is worth noting that IMO-CADE is an improved version of our previous work (PM-MDE) in [22]. However, there are several important differences between IMO-CADE and PM-MDE:

(1) The core difference is that, in IMO-CADE, both the operator feedback and individual information are considered together to update the operator selection probability, whereas in PM-MDE, only the operator feedback is used.
(2) In IMO-CADE, a new BCHT is developed for the OPA.
(3) The operators used in the operator pool are different between IMO-CADE and PM-MDE.

5. Results and Analysis

In this section, the proposed IMO-CADE is used to solve the OPA with different sensor nodes under independent or correlated observations.

5.1. Parameter Settings

For IMO-CADE, the following parameter settings are used:

- $Np = 100$;
- $\mu_{Cr} = 0.5$, $\mu_F = 0.5$, and $c = 0.1$;
- $p_{\min} = 0.05$ and $\alpha = 0.3$;
- $NFEs_{\max} = 3000$;
- Number of independent runs = 30;
- Observation signal-to-noise ratio (SNR), $\gamma_0 = 10$ dB.

In this work, the WSN consists of a fusion center and many spatially separated sensors, which perform amplify-and-forward local processing of their observations independently. The OPA problem was considered for the decision fusion of a deterministic signal in an inhomogeneous WSN. The channel fading coefficients follow an exponential distribution (i.e., Rayleigh fading) with unit mean. All algorithms were implemented in C++. The simulations were executed on a desktop PC with an Intel Xeon E5-2620 processor @ 2.40 GHz, 32 GB RAM, using the Windows 10 64-bit OS.

5.2. Comparison with Other Advanced DEs

To evaluate the performance of IMO-CADE, it is compared with five other DE variants for the OPA with different sensor nodes under i.i.d. observations. The five DE methods are as follows: (i) SaDE [34], (ii) JADE [29], (iii) OrSHADE [35], (iv) COLSHADE [36], and (v) CADE. The first four DE methods obtained promising results in the literature (To make a fair comparison, the parameters of the four DE variants are set as the same as those used in the original literature. All methods use $NFEs_{\max} = 3000$), for example, COLSHADE is one of the winners in the CEC-2020 competition for the real-world constrained single-objective optimization. CADE is a variant of IMO-CADE. The only difference between IMO-CADE and CADE is that only the operator feedback is used for operator selection in CADE.

The detailed results are reported in Table 1 for $K = 10, 20, 50, 100, 150, 200$, and $\epsilon = 0.1, 0.01, 0.001$, where the "mean $\pm$ standard deviation" value is provided for each method on each case. All results are averaged over 30 runs. The best mean values are highlighted in **boldface**. Based on the Wilcoxon test at $\alpha = 0.05$ by the KEEL software [37], in the last row of Table 1, "$w/t/l$" indicates that IMO-CADE performs significantly better than, similar to, or significantly worse than its competitors in w, t, or l cases, respectively. Moreover, according to the multiple-problem analysis by the Friedman test [37], the averaging rankings of the six DE variants are plotted in Figure 2. The multiple-problem analysis by the Wilcoxon test for IMO-CADE vs. other DEs is given in Table 2.

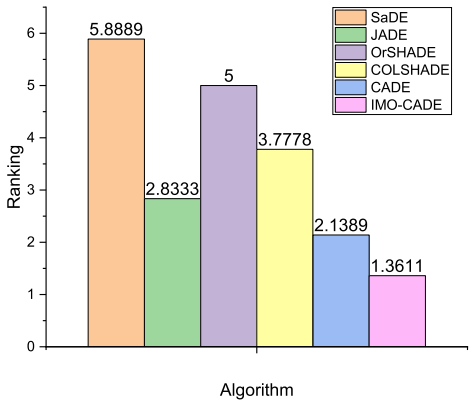


Figure 2. Average rankings of different DE variants obtained by the Friedman test for the OPA with i.i.d. observations, where the p-value computed by Iman and Daveport test is 0.

Table 1. Comparison of different DE variants for the OPA with different sensor nodes under i.i.d. observations.

K	ϵ	SaDE	JADE	OrSHADE	COLSHADE	CADE	IMO-CADE
10	0.1	$3.1762 \pm 1.45 \times 10^{-3}$	$3.1723 \pm 2.41 \times 10^{-5}$	$3.1742 \pm 6.00 \times 10^{-4}$	$3.1732 \pm 4.30 \times 10^{-4}$	$3.1723 \pm 1.15 \times 10^{-5}$	$3.1723 \pm 1.93 \times 10^{-5}$
	0.01	$15.1315 \pm 7.00 \times 10^{-4}$	$15.1303 \pm 7.19 \times 10^{-11}$	$15.1314 \pm 6.51 \times 10^{-4}$	$15.1304 \pm 1.30 \times 10^{-4}$	$15.1303 \pm 5.69 \times 10^{-12}$	$15.1303 \pm 5.55 \times 10^{-13}$
	0.001	$40.2599 \pm 6.18 \times 10^{-3}$	$40.2400 \pm 1.82 \times 10^{-4}$	$40.2469 \pm 2.63 \times 10^{-3}$	$40.2430 \pm 2.02 \times 10^{-3}$	$40.2400 \pm 1.57 \times 10^{-4}$	$40.2400 \pm 1.94 \times 10^{-4}$
20	0.1	$1.9541 \pm 5.47 \times 10^{-3}$	$1.9317 \pm 8.25 \times 10^{-5}$	$1.9379 \pm 1.34 \times 10^{-3}$	$1.9365 \pm 1.21 \times 10^{-3}$	$1.9317 \pm 2.11 \times 10^{-5}$	$1.9317 \pm 7.55 \times 10^{-5}$
	0.01	$9.1215 \pm 9.12 \times 10^{-3}$	$9.0970 \pm 6.50 \times 10^{-6}$	$9.1135 \pm 4.04 \times 10^{-3}$	$9.1111 \pm 5.12 \times 10^{-3}$	$9.0970 \pm 9.79 \times 10^{-6}$	$9.0970 \pm 9.67 \times 10^{-6}$
	0.001	$21.6245 \pm 1.28 \times 10^{-2}$	$21.5961 \pm 9.98 \times 10^{-5}$	$21.6252 \pm 6.77 \times 10^{-3}$	$21.6286 \pm 1.08 \times 10^{-2}$	$21.5962 \pm 2.27 \times 10^{-4}$	$21.5962 \pm 2.03 \times 10^{-4}$
50	0.1	$1.2003 \pm 4.88 \times 10^{-2}$	$0.8659 \pm 3.35 \times 10^{-4}$	$0.8804 \pm 2.42 \times 10^{-3}$	$0.8766 \pm 1.87 \times 10^{-3}$	$0.8660 \pm 3.39 \times 10^{-4}$	$0.8656 \pm 7.95 \times 10^{-5}$
	0.01	$4.8931 \pm 8.78 \times 10^{-2}$	$4.3288 \pm 4.22 \times 10^{-3}$	$4.3873 \pm 9.55 \times 10^{-3}$	$4.3834 \pm 8.23 \times 10^{-3}$	$4.3300 \pm 6.23 \times 10^{-3}$	$4.3254 \pm 3.90 \times 10^{-3}$
	0.001	$10.6059 \pm 1.11 \times 10^{-1}$	$9.8982 \pm 9.14 \times 10^{-3}$	$10.0035 \pm 1.62 \times 10^{-2}$	$10.0007 \pm 1.22 \times 10^{-2}$	$9.8955 \pm 1.05 \times 10^{-2}$	$9.8904 \pm 6.81 \times 10^{-3}$
100	0.1	$11.8999 \pm 1.52 \times 10^{0}$	$0.8762 \pm 1.77 \times 10^{-2}$	$0.9472 \pm 2.06 \times 10^{-2}$	$0.9032 \pm 1.14 \times 10^{-2}$	$0.8656 \pm 1.06 \times 10^{-2}$	$0.8539 \pm 1.25 \times 10^{-2}$
	0.01	$12.1699 \pm 1.30 \times 10^{0}$	$3.9931 \pm 5.92 \times 10^{-2}$	$4.1652 \pm 6.26 \times 10^{-2}$	$4.0969 \pm 3.84 \times 10^{-2}$	$3.9818 \pm 3.97 \times 10^{-2}$	$3.9329 \pm 2.35 \times 10^{-2}$
	0.001	$16.6726 \pm 6.26 \times 10^{-1}$	$8.6754 \pm 6.95 \times 10^{-2}$	$8.9279 \pm 6.85 \times 10^{-2}$	$8.8313 \pm 4.71 \times 10^{-2}$	$8.6435 \pm 4.76 \times 10^{-2}$	$8.5999 \pm 5.39 \times 10^{-2}$
150	0.1	$79.5028 \pm 9.10 \times 10^{0}$	$1.2345 \pm 9.28 \times 10^{-2}$	$1.4707 \pm 1.18 \times 10^{-1}$	$1.1482 \pm 3.41 \times 10^{-2}$	$1.1142 \pm 5.19 \times 10^{-2}$	$1.0252 \pm 4.80 \times 10^{-2}$
	0.01	$85.3694 \pm 1.02 \times 10^{1}$	$4.6995 \pm 1.14 \times 10^{-1}$	$5.0791 \pm 1.45 \times 10^{-1}$	$4.6995 \pm 5.62 \times 10^{-2}$	$4.5959 \pm 1.02 \times 10^{-1}$	$4.4696 \pm 1.10 \times 10^{-1}$
	0.001	$80.4239 \pm 1.54 \times 10^{1}$	$9.6689 \pm 1.51 \times 10^{-1}$	$10.2374 \pm 2.06 \times 10^{-1}$	$9.7378 \pm 9.40 \times 10^{-2}$	$9.5034 \pm 1.54 \times 10^{-1}$	$9.3211 \pm 1.18 \times 10^{-1}$
200	0.1	$224.0768 \pm 1.57 \times 10^{1}$	$2.0922 \pm 2.20 \times 10^{-1}$	$3.0493 \pm 4.45 \times 10^{-1}$	$1.6664 \pm 1.24 \times 10^{-1}$	$1.5992 \pm 1.62 \times 10^{-1}$	$1.3095 \pm 1.31 \times 10^{-1}$
	0.01	$239.8870 \pm 1.63 \times 10^{1}$	$4.8360 \pm 3.22 \times 10^{-1}$	$5.8158 \pm 4.71 \times 10^{-1}$	$4.3757 \pm 1.90 \times 10^{-1}$	$4.1938 \pm 2.22 \times 10^{-1}$	$3.8791 \pm 1.46 \times 10^{-1}$
	0.001	$237.0817 \pm 2.03 \times 10^{1}$	$8.5756 \pm 3.73 \times 10^{-1}$	$9.5102 \pm 5.91 \times 10^{-1}$	$8.0990 \pm 2.37 \times 10^{-1}$	$7.8208 \pm 2.74 \times 10^{-1}$	$7.4426 \pm 2.80 \times 10^{-1}$
$w/t/l$		16/0/0	12/4/0	16/0/0	16/0/0	11/5/0	-

Table 2. Results obtained by the Wilcoxon test for IMO-CADE vs. other DEs under i.i.d. observations. Hereinafter, $p \leq 0.05$ indicates that IMO-CADE significantly outperforms its competitors overall.

IMO-CADE vs.	R^+	R^-	p-Value
SaDE	171.0	0.0	7.63×10^{-6}
JADE	160.5	10.5	3.74×10^{-4}
OrSHADE	171.0	0.0	7.63×10^{-6}
COLSHADE	162.0	9.0	1.53×10^{-5}
CADE	160.5	10.5	3.74×10^{-4}

From Tables 1 and 2, and Figure 2, the following can be observed:

- IMO-CADE can obtain the best ranking based on the Friedman test, followed by CADE and JADE. In addition, the p-value computed by Iman and Daveport test is 0, which means that the performance of the six compared DEs are significantly different based on the multiple-problem analysis.
- According to the Wilcoxon test, IMO-CADE significantly outperforms SaDE, JADE, OrSHADE, COLSHADE, and CADE on 16, 12, 16, 16, and 11 cases, respectively. Compared with JADE and CADE, IMO-CADE obtains similar results on 4 and 5 cases, respectively. IMO-CADE is not outperformed by other DEs in any case.
- IMO-CADE obtains similar mean values compared with JADE and CADE when $K \leq 20$. However, when $K \geq 50$, IMO-CADE can consistently provide the best results in all cases.
- Compared with CADE, IMO-CADE obtains better results. This means that combining the operator feedback with the individual information for operator selection is really effective to improve the performance of CADE for the OPA.
- The results in Table 2 clearly show that IMO-CADE obtains significantly better results than other DEs based on multiple-problem analysis by the Wilcoxon test.

Therefore, based on the above analysis, it is clear that IMO-CADE yields the best results than other compared DE variants on the whole, especially for the OPA with a large number of sensor nodes.

5.3. Comparison with Reported Results

In this section, the results of IMO-CADE are compared with the reported results of other methods, i.e., CBBO [18], CDE [18], CBBO-DE [18], and PM-MDE [22], under both i.i.d. and correlated observations. Note that the five compared methods consume the same $NFEs_{\max} = 3000$ for fair comparison.

5.3.1. Under i.i.d. Observations

The results of the five compared methods for $K = 10, 20, 50$, and $\epsilon = 0.1, 0.01, 0.001$ are shown in Table 3. In addition, the results of the multiple-problem analysis by the Wilcoxon and Friedman tests (To calculate the statistical results, "NF" is set to be 1000.) are provided in Table 4 and Figure 3, respectively.

From Table 3, we can see that IMO-CADE can consistently obtain the best results compared with the other four methods. Additionally, the results in Table 4 show that our approach can perform significantly better than CBBO, CDE, and PM-MDE. Figure 3 confirms that IMO-CADE obtains the best ranking, followed by CBBO-DE and PM-MDE.

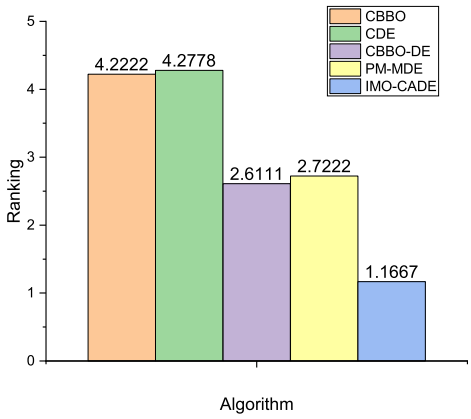


Figure 3. Average rankings of different methods obtained by the Friedman test for the OPA with the reported results under i.i.d. observations, where the p-value computed by Iman and Daveport test is 2.15×10^{-7}.

Table 3. Comparison with the reported results for the OPA under i.i.d. observations, where "NF" indicates that no feasible solution is obtained.

K	ϵ	CBBO		CDE		CBBO-DE		PM-MDE		IMO-CADE	
10	0.1	3.1991	1.64×10^{-2}	3.1732	5.83×10^{-4}	3.1725	1.18×10^{-6}	3.1727	9.28×10^{-5}	3.1723	1.93×10^{-5}
	0.01	15.1680	3.31×10^{-4}	15.1310	2.77×10^{-8}	11.5300	8.07×10^{-10}	15.1300	3.54×10^{-8}	15.1303	5.55×10^{-13}
	0.001	NF	NF	40.2450	5.59×10^{-6}	40.2450	5.59×10^{-6}	40.3170	3.95×10^{-4}	40.2400	1.94×10^{-4}
20	0.1	2.0406	3.91×10^{-2}	2.0485	1.25×10^{-1}	1.9333	1.49×10^{-3}	1.9343	1.38×10^{-3}	1.9317	7.55×10^{-5}
	0.01	9.2443	7.93×10^{-2}	9.1201	1.51×10^{-2}	9.0985	6.79×10^{-4}	9.1009	4.03×10^{-3}	9.0970	9.67×10^{-6}
	0.001	21.7400	6.07×10^{-2}	21.6260	2.98×10^{-2}	21.5980	6.42×10^{-4}	21.6010	4.30×10^{-3}	21.5962	2.03×10^{-4}
50	0.1	1.4513	1.57×10^{-1}	3.7790	4.12×10^{-1}	2.7516	3.91×10^{-1}	1.1192	6.22×10^{-2}	0.8656	7.95×10^{-5}
	0.01	4.9231	1.48×10^{-1}	6.8290	3.59×10^{-1}	6.2178	5.08×10^{-1}	4.7101	8.96×10^{-2}	4.3254	3.90×10^{-3}
	0.001	10.4960	1.49×10^{-1}	14.3240	1.49×10^{0}	11.0930	6.01×10^{-1}	10.3060	1.20×10^{-1}	9.8904	6.81×10^{-3}

Table 4. Results obtained by the Wilcoxon test for IMO-CADE vs. other methods with reported results under i.i.d. observations.

IMO-CADE vs.	R^+	R^-	p-Value
CBBO	45.0	0.0	3.91×10^{-3}
CDE	45.0	0.0	3.91×10^{-3}
CBBO-DE	36.0	9.0	1.29×10^{-1}
PM-MDE	40.5	4.5	7.81×10^{-3}

5.3.2. Under Correlated Observations

As mentioned in Section 2.1.2, when the observations are correlated, i.e., $\rho \neq 0$, the optimization of OPA is more complicated and computationally expensive. In this section, IMO-CADE is compared with CBBO, CDE, CBBO-DE, and PM-MDE with $K = 10, 20, 50$, $\rho = 0.01, 0.1, 0.5$, and $\epsilon = 0.1, 0.01, 0.001$. The detailed results are reported in Table 5. Moreover, the multiple-problem statistical results based on the Wilcoxon and Friedman tests are provided in Table 6 and Figure 4, respectively.

The results in Tables 4 and 5, and Figure 4 again confirms that IMO-CADE can get the best overall results in all cases. It is able to significantly outperform CBBO, CDE, CBBO-DE, and PM-MDE based on the Wilcoxon test. Further, it obtains the best ranking based on the Friedman test.

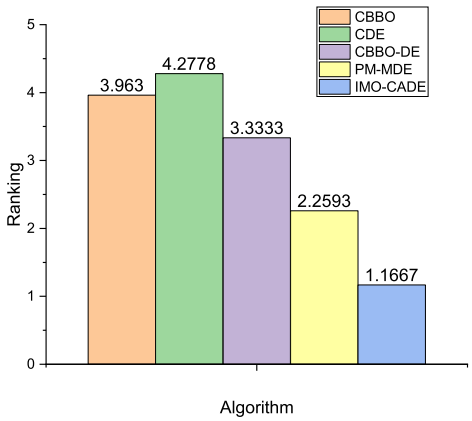


Figure 4. Average rankings of different methods obtained by the Friedman test for the OPA with the reported results under correlated observations, where the p-value computed by Iman and Daveport test is 0.

Based on the comparison with the reported results of other methods under both i.i.d. and correlated observations with different situations (i.e., number of sensor nodes, fusion

errors, and correlation degrees), the results clearly show that the proposed IMO-CADE can provide significantly better results than the compared methods. Therefore, IMO-CADE can be an effective and efficient alternative to solve the OPA of WSNs.

Table 5. Comparison with the reported results for the OPA under correlated observations.

					$K = 10$					
ρ	ϵ	**CBBO**		**CDE**		**CBBO-DE**		**PM-MDE**		**IMO-CADE**	
0.01	0.1	3.2129	2.17×10^{-2}	3.1847	6.74×10^{-4}	3.1833	3.86×10^{-4}	3.1834	1.60×10^{-4}	3.1830	9.53×10^{-6}
	0.01	15.3070	7.97×10^{-4}	15.2560	2.39×10^{-8}	15.2550	1.89×10^{-9}	15.2550	3.34×10^{-8}	15.2547	6.89×10^{-13}
	0.001	NF	NF	40.9860	3.00×10^{-6}	41.0460	4.63×10^{-4}	40.9800	3.45×10^{-4}	40.9795	2.94×10^{-4}
0.1	0.1	3.3100	1.77×10^{-2}	3.2900	8.43×10^{-4}	3.2800	1.12×10^{-4}	3.2792	9.96×10^{-5}	3.2789	4.19×10^{-5}
	0.01	16.6000	4.30×10^{-4}	16.6000	2.21×10^{-7}	16.6000	3.56×10^{-9}	16.4890	4.61×10^{-6}	16.4885	1.05×10^{-9}
	0.001	NF	NF	48.9850	1.49×10^{-7}	49.0770	7.81×10^{-4}	48.6440	1.12×10^{-10}	48.6440	3.38×10^{-13}
0.5	0.1	3.8800	1.87×10^{-2}	3.8600	1.39×10^{-3}	3.8600	2.32×10^{-4}	3.5839	8.56×10^{-4}	3.5824	5.77×10^{-5}
	0.01	3.5100	6.13×10^{-3}	34.4000	1.85×10^{-6}	34.3000	8.08×10^{-7}	22.8030	8.88×10^{-4}	22.8014	9.16×10^{-6}
	0.001	NF	NF	734.2400	1.88×10^{-4}	735.1400	6.63×10^{-2}	107.7800	6.79×10^{0}	105.5153	6.72×10^{-13}
					$K = 20$					
ρ	ϵ	**CBBO**		**CDE**		**CBBO-DE**		**PM-MDE**		**IMO-CADE**	
0.01	0.1	2.0292	4.15×10^{-2}	2.0127	1.51×10^{-2}	1.9396	2.14×10^{-3}	1.9394	1.79×10^{-3}	1.9373	1.73×10^{-5}
	0.01	9.3053	7.68×10^{-2}	91.7670	1.03×10^{-2}	9.1607	8.98×10^{-4}	91.6340	3.74×10^{-3}	9.1588	2.00×10^{-6}
	0.001	2.1980	4.97×10^{-2}	21.8600	1.92×10^{-2}	21.8420	3.99×10^{-3}	21.8420	3.14×10^{-3}	21.8383	8.06×10^{-6}
0.1	0.1	2.0913	4.71×10^{-2}	2.0799	1.29×10^{-1}	1.9905	1.32×10^{-3}	1.9908	3.58×10^{-3}	1.9871	9.65×10^{-6}
	0.01	99.2370	5.59×10^{-2}	9.8126	3.08×10^{-2}	9.7894	9.19×10^{-4}	97.5540	5.49×10^{-3}	9.7484	9.59×10^{-5}
	0.001	24.5070	8.38×10^{-2}	24.3400	1.49×10^{-2}	24.3240	1.96×10^{-3}	24.1820	3.48×10^{-3}	24.1772	6.46×10^{-6}
0.5	0.1	2.3633	2.55×10^{-2}	2.3406	3.53×10^{-2}	2.3026	2.19×10^{-3}	2.1879	7.40×10^{-3}	2.1780	1.43×10^{-3}
	0.01	15.9850	5.07×10^{-2}	15.8750	7.75×10^{-3}	15.8650	3.27×10^{-3}	12.5470	8.55×10^{-2}	12.4527	1.34×10^{-3}
	0.001	63.8450	9.39×10^{-1}	60.9090	2.30×10^{0}	60.6850	6.51×10^{-2}	36.2470	9.15×10^{-2}	36.1418	1.26×10^{-2}
					$K = 50$					
ρ	ϵ	**CBBO**		**CDE**		**CBBO-DE**		**PM-MDE**		**IMO-CADE**	
0.01	0.1	1.5072	1.35×10^{-1}	3.5357	4.37×10^{-1}	2.7742	5.61×10^{-1}	1.2346	1.11×10^{-1}	0.8687	2.08×10^{-4}
	0.01	4.9139	1.36×10^{-1}	6.8366	2.26×10^{-1}	6.0532	7.98×10^{-1}	4.8249	1.31×10^{-1}	4.3582	2.40×10^{-3}
	0.001	10.6430	2.23×10^{-1}	12.0540	2.28×10^{-1}	10.7460	3.31×10^{-1}	10.5210	7.40×10^{-2}	9.9713	2.49×10^{-3}
0.1	0.1	1.4947	1.64×10^{-1}	3.7939	3.94×10^{-1}	3.0813	4.30×10^{-1}	1.2406	9.45×10^{-2}	0.8960	4.22×10^{-5}
	0.01	5.2413	1.67×10^{-1}	7.0517	2.90×10^{-1}	6.6532	1.01×10^{0}	5.1088	8.14×10^{-2}	4.6572	5.00×10^{-3}
	0.001	11.3850	1.54×10^{-1}	13.0250	2.32×10^{-1}	11.6690	5.81×10^{-1}	11.2840	1.20×10^{-1}	10.7169	2.02×10^{-3}
0.5	0.1	1.6223	1.80×10^{-1}	4.1081	6.54×10^{-1}	2.9672	4.70×10^{-1}	1.3432	8.87×10^{-2}	1.0055	2.86×10^{-3}
	0.01	7.1915	1.50×10^{-1}	8.5854	1.20×10^{-1}	7.4210	3.24×10^{-1}	6.2096	1.13×10^{-1}	5.6704	5.32×10^{-2}
	0.001	18.8410	1.20×10^{-1}	19.3060	4.42×10^{-1}	18.4490	3.68×10^{-2}	14.8240	1.54×10^{-1}	13.8922	4.27×10^{-2}

Table 6. Results obtained by the Wilcoxon test for IMO-CADE vs. other methods with reported results under correlated observations.

IMO-CADE vs.	R^+	R^-	p**-Value**
CBBO	335.0	43.0	1.84×10^{-4}
CDE	378.0	0.0	1.49×10^{-8}
CBBO-DE	364.5	13.5	2.98×10^{-8}
PM-MDE	373.0	5.0	1.49×10^{-7}

5.4. Discussions

In previous subsections, the superiority of IMO-CADE is verified on both the i.i.d. and correlated observations with different numbers of sensors. Herein, the gain allocated to each sensor of different methods is discussed. Table 7 reports the results comparison between analytical and different numerical methods with $K = 10$, $\gamma_0 = 10$ dB, and $\rho = 0$. Table 8 reports the results of different numerical methods with $K = 10$, $\gamma_0 = 10$ dB, and $\rho = 0.1$.

The best results of $f(\mathbf{x})$ are highlighted in boldface. In the two tables, IMO-CADE is compared with the reported results of CBBO, CDE, and CBBO-DE in [18].

Table 7. Comparison between analytical and different numerical methods to gain allocation with $K = 10$, $\gamma_0 = 10$ dB, and $\rho = 0$.

Sensor	$\epsilon = 0.1$					$\epsilon = 0.01$				
	Analytical	**CBBO**	**CDE**	**CBBO-DE**	**IMO-CADE**	**Analytical**	**CBBO**	**CDE**	**CBBO-DE**	**IMO-CADE**
G1	1.0362	1.0823	1.0392	1.0353	1.0361	1.6172	1.5500	1.5894	1.5925	1.5926
G2	0.9972	0.9838	0.9985	0.9977	0.9972	1.5888	1.6000	1.5826	1.5821	1.5821
G3	0.8834	0.8619	0.8853	0.8826	0.8836	1.5555	1.5185	1.5469	1.5496	1.5483
G4	0.4823	0.5219	0.4623	0.4812	0.4825	1.4666	1.4610	1.4423	1.4381	1.4379
G5	0.3021	0.1330	0.3061	0.3066	0.3010	1.4616	1.4231	1.4069	1.4050	1.4050
G6	0	0.0031	0.0655	0.0135	0.0111	1.4107	1.3738	1.3600	1.3606	1.3605
G7	0	0.0656	0.0117	0.0142	0.0029	1.1231	1.3503	1.3405	1.3404	1.3420
G8	0	0.0067	0.0006	0.0033	1.14×10^{-4}	0	0.0056	0.0070	0.0053	1.20×10^{-6}
G9	0	0.0020	0.0035	0.0004	8.97×10^{-5}	0	0.0173	0.0037	0.0058	1.78×10^{-7}
G10	0	0.0076	0.0010	0.0046	4.28×10^{-5}	0	0.0016	0.0032	0.0013	1.10×10^{-7}
$f(\mathbf{x})$	3.1723	3.1766	3.1725	3.1723	3.1723	15.09782	15.13894	15.13036	15.13032	15.13027

Table 8. Comparison of different numerical methods to gain allocation with $K = 10$, $\gamma_0 = 10$ dB, and $\rho = 0.1$.

Sensor	$\epsilon = 0.1$				$\epsilon = 0.01$			
	CBBO	**CDE**	**CBBO-DE**	**IMO-CADE**	**CBBO**	**CDE**	**CBBO-DE**	**IMO-CADE**
G1	1.0942	1.0507	1.0475	1.0442	1.6645	1.6733	1.6751	1.6717
G2	0.8958	0.9567	0.9606	0.9574	1.5635	1.5863	1.5869	1.5754
G3	0.8978	0.8850	0.8784	0.8818	1.5626	1.5775	1.5771	1.5751
G4	0.5006	0.5091	0.5015	0.5103	1.5491	1.4767	1.4805	1.5086
G5	0.4439	0.4246	0.4431	0.4508	1.5218	1.4797	1.4809	1.4817
G6	0.1227	0.1997	0.1959	0.1471	1.4327	1.4504	1.4429	1.4812
G7	0.1421	0.0371	0.0743	0.0952	1.4659	1.5110	1.5111	1.4374
G8	0.0263	0.0197	0.0067	0.0012	0.0019	0.0245	0.0130	3.84×10^{-5}
G9	0.0304	0.0141	0.0022	0.0007	0.0211	0.0044	0.0048	9.83×10^{-6}
G10	0.0103	0.0142	0.0019	0.0001	0.0100	0.0080	0.0005	3.69×10^{-6}
$f(\mathbf{x})$	3.2902	3.2839	3.2833	3.2789	16.5744	16.5625	16.5623	16.4885

From the results in Tables 7 and 8, the following can be seen:

- Based on the gain allocated to each sensor, the sensors with poor channels can be turned off to save system power. Based on the gain allocated to each node, we can decide that the sensors with good channel fading coefficients are assigned to more power; on the other hand, sensors with poor channels are allocated less power.
- For the i.i.d. observations, the numerical results of CDE, CBBO-DE, and IMO-CADE closely match with the analytical results.
- When the observations are correlated, the sensors need more power compared with the i.i.d. observations.

- For both i.i.d. and correlated observations, IMO-CADE provides the best results of $f(\mathbf{x})$ compared with other methods.

6. Conclusions

The OPA is one of key issues in developing WSNs. Hence, the design of effective optimization techniques for the OPA is important and obtains more attention. Based on this consideration, in this paper, we propose the IMO-CADE for the OPA. To effectively solve the OPA, in IMO-CADE, both operator feedback and individual information are considered simultaneously for multioperator selection. In addition, the parameter adaptation, efficient BCHT, and constrained reward assignment techniques are developed. IMO-CADE is extensively compared with other methods under various situations. The results clearly indicate the following:

- The proposed modifications in IMO-CADE can improve its performance for the OPA under different situations.
- With respect to the performance of the overall system power $f(\mathbf{x})$, IMO-CADE is superior to other methods in all cases, especially for the WSN with a large number of sensor nodes.
- Considering gain allocation, the numerical results of IMO-CADE agree well with the analytical results.
- IMO-CADE can be an effective alternative for the OPA and other complex optimization problems of WSNs.

When the observations of sensor nodes are correlated, the OPA is computationally expensive—especially if there are a large number of sensor nodes. In the near future, combining IMO-CADE with surrogate models [38] is a promising way to efficiently solve the OPA under correlated observations. In addition, due to the similarity between the OPA and the lighting in long tunnels [39], IMO-CADE will be used for the lighting problem in long tunnels in future work. In this work, the simulations are performed to evaluate the performance of IMO-CADE. In future work, another interesting direction is using IMO-CADE in real networks.

The source code of IMO-CADE can be obtained from Dr. Gong upon request.

Author Contributions: Conceptualization, methodology, validation, data curation, & writing—original draft preparation, W.L.; supervision, writing—review and editing, & funding acquisition, W.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant No. 62076225, and the Natural Science Foundation for Distinguished Young Scholars of Hubei under Grant No. 2019CFA081.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the reviewers and editor for their constructive comments.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACO	Ant Colony Optimization
BBO	Bio-geography-Based Optimization
BCHT	Boundary Constraint Handling Technique
CV	Constraint Violation

DE	Differential Evolution
IMO-CADE	Improved Multioperator-based Constrained Adaptive DE
OPA	Optimal Power Allocation
PSO	Particle Swarm Optimization
WSNs	Wireless Sensor Networks

References

1. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
2. Yick, J.; Mukherjee, B.; Ghosal, D. Wireless sensor network survey. *Comput. Netw.* **2008**, *52*, 2292–2330. [CrossRef]
3. Riaz, A.; Sarker, M.R.; Saad, M.H.M.; Mohamed, R. Review on Comparison of Different Energy Storage Technologies Used in Micro-Energy Harvesting, WSNs, Low-Cost Microelectronic Devices: Challenges and Recommendations. *Sensors* **2021**, *21*, 5041. [CrossRef]
4. Kulkarni, R.; Förster, A.; Venayagamoorthy, G. Computational Intelligence in Wireless Sensor Networks: A Survey. *IEEE Commun. Surv. Tutor.* **2011**, *13*, 68–96. [CrossRef]
5. Kulkarni, R.; Venayagamoorthy, G. Particle Swarm Optimization in Wireless-Sensor Networks: A Brief Survey. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2011**, *41*, 262–267. [CrossRef]
6. Adnan, M.A.; Razzaque, M.A.; Ahmed, I.; Isnin, I.F. Bio-Mimic Optimization Strategies in Wireless Sensor Networks: A Survey. *Sensors* **2014**, *14*, 299–345. [CrossRef]
7. Praveen Kumar, D.; Amgoth, T.; Annavarapu, C.S.R. Machine learning algorithms for wireless sensor networks: A survey. *Inf. Fusion* **2019**, *49*, 1–25. [CrossRef]
8. Singh, A.; Sharma, S.; Singh, J. Nature-inspired algorithms for Wireless Sensor Networks: A comprehensive survey. *Comput. Sci. Rev.* **2021**, *39*, 100342. [CrossRef]
9. Jiao, D.; Yang, P.; Fu, L.; Ke, L.; Tang, K. Optimal Energy-Delay Scheduling for Energy-Harvesting WSNs with Interference Channel via Negatively Correlated Search. *IEEE Internet Things J.* **2020**, *7*, 1690–1703. [CrossRef]
10. Wu, L.; Cai, H. Energy-Efficient Adaptive Sensing Scheduling in Wireless Sensor Networks Using Fibonacci Tree Optimization Algorithm. *Sensors* **2021**, *21*, 5002. [CrossRef]
11. Xu, H.; Gao, H.; Zhou, C.; Duan, R.; Zhou, X. Resource Allocation in Cognitive Radio Wireless Sensor Networks with Energy Harvesting. *Sensors* **2019**, *19*, 5115. [CrossRef] [PubMed]
12. Filomeno, M.D.L.; de Campos, M.L.R.; Poor, H.V.; Ribeiro, M.V. Hybrid Power Line/Wireless Systems: An Optimal Power Allocation Perspective. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 6289–6300. [CrossRef]
13. Ojo, F.K.; Akande, D.O.; Salleh, M.F.M. Optimal Power Allocation in Cooperative Networks with Energy-Saving Protocols. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5079–5088. [CrossRef]
14. Guo, F.; Xu, B.; Zhang, W.A.; Wen, C.; Zhang, D.; Yu, L. Training Deep Neural Network for Optimal Power Allocation in Islanded Microgrid Systems: A Distributed Learning-Based Approach. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, 1–13. [CrossRef]
15. Zhao, B.; Zhao, X. Deep Reinforcement Learning Resource Allocation in Wireless Sensor Networks with Energy Harvesting and Relay. *IEEE Internet Things J.* **2021**, 1. [CrossRef]
16. Wimalajeewa, T.; Jayaweera, S. Optimal Power Scheduling for Data Fusion in Inhomogeneous Wireless Sensor Networks. In Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, Sydney, Australia, 24 November 2006; pp. 73–78.
17. Wimalajeewa, T.; Jayaweera, S. Optimal Power Scheduling for Correlated Data Fusion in Wireless Sensor Networks via Constrained PSO. *IEEE Trans. Wirel. Commun.* **2008**, *7*, 3608–3618. [CrossRef]
18. Boussaïd, I.; Chatterjee, A.; Siarry, P.; Ahmed-Nacer, M. Hybridizing Biogeography-Based Optimization with Differential Evolution for Optimal Power Allocation in Wireless Sensor Networks. *IEEE Trans. Veh. Technol.* **2011**, *60*, 2347–2353. [CrossRef]
19. Tsiflikiotis, A.; Goudos, S.K. Optimal power allocation in wireless sensor networks using emerging nature-inspired algorithms. In Proceedings of the 2016 5th International Conference on Modern Circuits and Systems Technologies (MOCAST), Thessaloniki, Greece, 12–14 May 2016; pp. 1–4.
20. Tsiflikiotis, A.; Goudos, S.K.; Karagiannidis, G.K. Hybrid teaching-learning optimization of wireless sensor networks. *Trans. Emerg. Telecommun. Technol.* **2017**, *28*, e3194. [CrossRef]
21. Lee, J. Optimal power allocating for correlated data fusion in decentralized WSNs using algorithms based on swarm intelligence. *Wirel. Netw.* **2017**, *23*, 1655–1667. [CrossRef]
22. Li, Y.; Gong, W.; Cai, Z. Optimal Power Allocation of Wireless Sensor Networks with Multi-operator Based Constrained Differential Evolution. Artificial Life and Computational Intelligence. ACALCI 2017. *Lect. Notes Comput. Sci.* **2017**, *10142*, 339–352.
23. Storn, R.; Price, K. Differential Evolution–A Simple and Efficient Heuristic for Global Optimization Over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
24. Wimalajeewa, T.; Jayaweera, S. PSO for Constrained Optimization: Optimal Power Scheduling for Correlated Data Fusion in Wireless Sensor Networks. In Proceedings of the IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, 2007 (PIMRC 2007), Athens, Greece, 3–7 September 2007; pp. 1–5.

25. Coello, C.A.C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 1245–1287. [CrossRef]
26. Deb, K. An efficient constraint handling method for genetic algorithms. *Comput. Methods Appl. Mech. Eng.* **2000**, *186*, 311–338. [CrossRef]
27. Gong, W.; Cai, Z. An empirical study on differential evolution for optimal power allocation in WSNs. In Proceedings of the 2012 8th International Conference on Natural Computation, Chongqing, China, 20 May 2012; pp. 635–639.
28. Das, S.; Suganthan, P.N. Differential evolution: A survey of the state-of-the-art. *IEEE Trans. Evol. Comput.* **2011**, *15*, 4–31. [CrossRef]
29. Zhang, J.; Sanderson, A.C. JADE: Adaptive Differential Evolution with Optional External Archive. *IEEE Trans. Evol. Comput.* **2009**, *13*, 945–958. [CrossRef]
30. Zhang, J.; Sanderson, A.C. *Adaptive Differential Evolution: A Robust Approach to Multimodal Problem Optimization*; Springer: Berlin, Germany, 2009.
31. Wang, Y.; Cai, Z. Constrained Evolutionary Optimization by Means of $(\mu + \lambda)$-Differential Evolution and Improved Adaptive Trade-Off Model. *Evol. Comput.* **2011**, *19*, 249–285. [CrossRef] [PubMed]
32. Goldberg, D.E. Probability Matching, the Magnitude of Reinforcement, and Classifier System Bidding. *Mach. Learn.* **1990**, *5*, 407–425. [CrossRef]
33. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [CrossRef]
34. Qin, A.K.; Huang, V.L.; Suganthan, P.N. Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization. *IEEE Trans. Evol. Comput.* **2009**, *13*, 398–417. [CrossRef]
35. Mohamed, A.W.; Hadi, A.A.; Jambi, K.M. Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization. *Swarm Evol. Comput.* **2019**, *50*, 100455. [CrossRef]
36. Gurrola-Ramos, J.; Hernandez-Aguirre, A.; Dalmau-Cedeno, O. COLSHADE for Real-World Single-Objective Constrained optimization Problems. In Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, 19–24 July 2020; pp. 1–8.
37. Alcalá-Fdez, J.; Sánchez, L.; García, S.; del Jesus, M.J.; Ventura, S.; Garrell, J.M.; Otero, J.; Romero, C.; Bacardit, J.; Rivas, V.M.; et al. KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Comput.* **2009**, *13*, 307–318. [CrossRef]
38. Ong, Y.S.; Keane, A.J.; Nair, P.B. Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling. *Am. Inst. Aeronaut. Astronaut. J.* **2003**, *41*, 687–696. [CrossRef]
39. Mehri, A.; Sajedifar, J.; Abbasi, M.; Naimabadi, A.; Mohammadi, A.A.; Teimori, G.H.; Zakerian, S.A. Safety evaluation of lighting at very long tunnels on the basis of visual adaptation. *Saf. Sci.* **2019**, *116*, 196–207. [CrossRef]