
Haoxiang Qin, Wenlei Bai, Yi Xiang*, Fangqing Liu, Yuyan Han, and Ling Wang

Abstract—Handling energy resource management (ERM) in today’s energy systems is complex and challenging due to uncertainties arising from the high penetration of distributed energy resources. Such penetration introduces various uncertain factors, such as renewable energy, energy storage, and electric vehicles, making it difficult for traditional mathematical methods to find effective solutions. However, Evolutionary Algorithms (EAs) have shown good performance in solving this problem. Therefore, in this paper, a self-adaptive collaborative differential evolution algorithm (SADEA) is proposed to solve the ERM problem under uncertainty. In SADEA, a three-stage adaptive collaboration strategy, includes boundary randomization stage, knowledge-assisted collaboration stage, and range restructuring stage, is used to generate collaborative solutions. The collaborative solutions generated in the above stages will jointly participate in the perturbation of DE strategies to explore promising solutions. In addition, different DE strategies are selected according to count values and random factors. At the end of the algorithm, boundary control, elite solution and retention are used to ensure the legitimacy and robustness of solutions. The proposed SADEA is compared to several state-of-the-art algorithms on a real-world distribution network located in Salamanca, Spain. The results show that SADEA is superior to its competitors in terms of the objective function, ranking index, and convergence. In summary, the proposed algorithm is effective to handle the ERM problem under uncertainty.

Index Terms—Energy resource management, smart grid, uncertainty scheduling, differential evolution, optimization.

I. INTRODUCTION

In recent years, with the continuous development of technology, the power grid has evolved into an advanced electric grid, namely smart grid (SG) [1]. The SG is the power grid that uses advanced commutation technology and control programs, such as demand response (DR), voltage optimization control, fault location, isolation and service restoration. Its main feature is the large penetration of distributed energy resources (DERs). However, DERs (e.g., wind and solar generation) are, by nature, intermittent, which makes it difficult to operate SG reliably and economically [2], [3]. Thus, for embracing DERs, the energy resources management (ERM) system is proposed to automatically negotiate actions, with the goal of achieving a dynamic supply-demand balance [4].

In SG, ERM simply refers to a real-time control system which is used to manage energy resources. It can be modeled as a mixed integer nonlinear problem (MINLP) where profits are to be maximized through control and automation capabilities [5]. To achieve such a goal in real-world scenarios, ERM must interact with various resources under uncertainty, such as electricity market prices, DR procedures, renewable energy generation, energy storage systems (ESSs) and electric vehicles (EVs) [6], [7].

Currently, many studies have made various attempts to tackle the aforementioned uncertainties in resource management. Most of them use traditional mathematical methods, such as robust optimization models [8], [9] and stochastic models [10], [11]. Typically, these models are solved by using deterministic mathematical methods. Nevertheless, traditional mathematical methods face several challenges, including the management of large amounts of resources, high level of accuracy for uncertainty representation, and integration of nonlinear functions (e.g., alternating current power flow and generator quadratic functions) [12]. The above limitations drive researchers to seek for different approaches to tackle this problem, and it has been found that Evolutionary Algorithms (EAs) are capable of effectively solving this problem [13].

Inspired by biological evolutionary mechanisms, EAs adopt operations like crossover, mutation, selection, abandonment, and retention to improve solutions. Well-known EAs include Genetic Algorithms (GA) [14], [15], Differential Evolution (DE) algorithms [16], Particle Swarm Optimization (PSO) algorithms [17], and Iterated Greedy (IG) algorithms [18]–[20]). However, population-based GA, DE, and PSO cannot focus all their efforts on a particular solution during the evolution. The IG algorithm, a single-individual EA, focuses all its efforts on a single solution, which is one of the main reasons for its strong local search ability [21]. However, existing IG algorithms tend to achieve good results in solving discrete problems, and yet not continuous problems such as the ERM.
problem under uncertainty. One of the obvious drawbacks is its poor global search ability due to the lack of solution diversity \[^{[22]}\]. Therefore, if using a single-individual mechanism, it is necessary to design perturbation strategies that can improve the global search ability of the algorithm. Inspired by \[^{[12]}\], we find that the DE strategy is effective in improving the global search ability of the IG algorithm. Therefore, we utilize the single-individual mechanism with the DE strategies, hoping to solve the ERM problem, efficiently.

The major challenges for the ERM problems are summarized as follows.

- Extreme scenarios may endanger the operation or even destroy the aggregator. This requires a risk-based optimization model to protect aggregators from events with low probability and yet high impacts.
- So far, only a few EAs have considered risk-based methodologies when solving ERM under uncertainty. EAs are very subject to the constraints of the ERM. If certain conditions change, the search ability of EAs cannot be fully utilized.
- Population-based EAs may not be able to focus on a promising neighborhood of current solutions, which will distract the computational performance of algorithms and result in a stagnation of the entire evolution. In addition, according to the results in \[^{[23]}\], the global search ability of the perturbation strategy also needs to be further improved.

Nevertheless, DE strategies still have many advantages, such as good perturbation and stable convergence. At present, the variants of DE, e.g., HyDE \[^{[12]}\] and ReSaDE \[^{[24]}\] (won first place in the WCCI (CEC)/ECCO 2022 competition) have been used to solve the uncertain ERM problem and achieved good results. Therefore, this paper designs the methods based on DE strategies. Given the above challenges, our study transforms the population-based DE into a single individual to avoid dispersed search. During the search process, collaborative solutions participate in the evolution by using DE strategies, which improve the local and global search abilities. Based on the above, a Self-Adaptive Collaborative Differential Evolution Algorithm (SADEA) is designed to solve the ERM under uncertainty. Contributions of this paper are summarized as follows.

- To choose the most appropriate DE strategy for a certain stage, a three-stage adaptive collaboration strategy is designed. In the first stage, three collaborative solutions are utilized to effectively improve solution diversity. In the second stage, a historical archive is used to explore potentially promising solutions while accelerating the search process. In the third stage, the range restructuring prevents solutions from falling into local optimum.
- To improve the local search ability, we use a single-individual mechanism. It allows the algorithm to focus all its efforts on a single solution. Compared to population-based optimization algorithms, it can enable the solution to evolve more quickly. Meanwhile, it can reduce the tedious steps, making it suitable for power dispatch in various scenarios.

Fig. 1. The ERM problem \[^{[25]}\]

- To overcome the weak global search abilities of single-individual EAs, three DE strategies are proposed. The proposed DE strategies significantly increase the diversity of solutions and prevent them from falling into local optimum.

The remainder of this paper is organized as follows. Section II introduces the related work of the ERM problem. The definitions of the problem are given in Section III. Section IV provides a case study of the distribution network, which is a real-world problem to be addressed in this paper. Section V lists details of SADEA. In Section VI, the results show that the proposed algorithm performs better than the state-of-the-art algorithms in solving the ERM problem. The conclusion and future work are given in Section VII.

II. RELATED WORK

In this section, the related work will be divided into two categories: A) Risk-based methodologies, B) EAs-based methodologies.

A. Risk-based Methodologies

Risk-based methodologies have been commonly used in electrical systems to reduce natural disasters or utility outages \[^{[26]}–[28]}\]. The emergence of extreme scenarios can have a huge impact on the management of electric network operators \[^{[29]}\]. Recently, several studies have used risk-based methodologies for aggregators, such as wind power generation \[^{[30]}\], conventional hydropower \[^{[31]}\], and the uncertainty of microgrid planning \[^{[32]}\]. The work in \[^{[33]}\] considered the uncertainty of load and pool market prices, and proposed a risk-based approach to achieve equal cost in an uncertain scenario. This approach increases the retailers’ cost but reduces the risk to almost zero. In other words, although the cost increases to a certain extent, it is still worthwhile to reduce the risk to zero for extreme scenarios.

In recent years, risk-based measurement mechanisms such as value-at-risk (VaR\(_\alpha\)) have been proposed. However, VaR\(_\alpha\) can only measure risk when the expected cost does not exceed the confidence level \(\alpha\) for all scenarios. On the contrary,
conditional $VaR_\alpha$ ($CVaR_\alpha$) allows measurement of risk beyond confidence level scenarios [34, 35]. It shows that, by introducing $CVaR_\alpha$, the impact of extreme scenarios will be minimized. Therefore, in this paper, both $VaR_\alpha$ and $CVaR_\alpha$ risk assessment parameters are considered to protect the aggregator. Fig. [1] gives the energy resource Buy/Sell in ERM, and the communication link to the aggregator [25]. The final optimization objective is to maximize the profit, or, to minimize the difference between operational costs (OC) and incomes (In) in each scenario, which takes into account penalty values and risk assessment parameters.

B. EAs Utilized to Solve the Uncertainty ERM

In EAs, CUMDANCau {ch 36} generates new individuals not only by sampling from the learned distribution but also by using the individual neighborhoods in a ring cell structure. It achieves better performance compared to variants of PSO or variable neighborhood search algorithms [37–39]. In [23], the Cooperative Co-evolution Strategies with Time-dependent Grouping (CCSTG), inspired by cooperative co-evolution frameworks, groups the decision variables into subcomponents. It generates new individuals from the univariate marginal distribution in the subpopulation. Population regeneration star-guided optimization (Presto) (which won the PES-GM/CEC/GECO 2021 competition) [23] is inspired by PSO and DE algorithms and seeks to increase the diversity of solutions. Hybrid-Adaptive Differential Evolution (HyDE) algorithm [12] uses multiple individuals for a series of evolutions, and although it increases the diversity of solutions, to a certain extent, it sacrifices local search ability. In addition, after one evolution, only improved solutions replace the original ones, while unimproved solutions continue to participate in the next evolution. Therefore, the increase of the solution diversity is limited, and evaluations on multiple solutions will disperse the algorithm’s search ability. In addition, the aforementioned EAs do not consider risk-based methodologies.

Restart-assisted Self-adaptive Differential Evolution (ReSaDE) algorithm, which won the first place in the WCCI (CEC)/ECCO 2022 competition [24], divides the variables into four different groups based on their bounded upper and lower bounds and improves them separately. Although this algorithm is tested in ERM with risk assessment parameters, it has several limitations. For example, the grouping is subject to the range of variables. It may lead to performance degradation and scalability issues. In addition, it is difficult to transplant due to its complex structure and strong dependence on problems. Therefore, this paper proposes SADEA to fill these gaps. Compared with ReSaDE, SADEA has a simpler structure, making it easier to solve other similar problems.

III. PROBLEM DESCRIPTION

Risk-based ERM under uncertainty is formulated as a MINLP where the objective for the aggregator is to generate a day-ahead energy schedule for DERs, EVs, ESSs, and DR to maximize the expected profits, subject to constraints such as resource limits, energy balance [40]. To protect aggregators from impacts caused by uncertainty, the risk assessment parameters $VaR_\alpha$ and $CVaR_\alpha$ are introduced. In general, $VaR_\alpha$ deals with the scenarios where events with a high probability of occurrence but a reasonably small effects (such as small error in renewable forecast). $CVaR_\alpha$ considers events with a low probability, but have high impacts (such as hurricanes, thunderstorms, and errors in the forecast).

In this section, the ERM problem will be divided into three parts to illustrate: A) risk-based ERM and fitness function, B) scenario generation, and C) encoding of the solution.

A. Risk-based ERM and Fitness Function

Due to space limitations, the notation of the ERM problem has been included in the supplementary material. Readers can refer to the original literature [35] for the supplementary material for details.

As shown in [35], the risk-neutral strategy considers the uncertain behavior of renewable energy generation, electric vehicles, market prices, load consumption, etc. These stochastic behaviors are considered using various methods that account for different scenarios and their associated probabilities of occurrence. When a risk is not taken into account, the aggregator formulates its schedule based on an expected scenario. In this case, the cost and the objective function (OF) values are based on the expected cost, which is formulated as follows:

$$Z_s^{tot} = Z_s^{OC} - Z_s^{In} + P_s$$

$$Z_s^{Ex} = \sum_{s=1}^{N_s} (\rho_s \cdot Z_s^{tot})$$

where $Z_s^{tot}$ is the total OF value of each scenario $s$; $Z_s^{OC}$ represents the operational costs; $Z_s^{In}$ represents the income and $P_s$ indicates the penalty for bound violations. In Equation (2), $Z_s^{Ex}$ indicates the expected OF; $\rho_s$ denotes the probability of the corresponding scenario $s$, and $N_s$ denotes the total number of scenarios.

Risk-aversion strategies consider the risks associated with uncertainty. In (1-$\alpha$)% of the scenarios with the highest costs, $CVaR_\alpha$ is an additional cost added to $Z_s^{Ex}$. Its calculation is as follows.

$$CVaR_\alpha (Z_s^{tot}) = VaR_\alpha (Z_s^{tot}) + \frac{1}{1-\alpha} \sum_{s=1}^{N_s} (\rho_s \cdot \varphi)$$

where:

$$\varphi = \begin{cases} Z_s^{tot} - Z_s^{Ex} - VaR_\alpha (Z_s^{tot}), & Z_s^{tot} \geq Z_s^{Ex} + VaR_\alpha (Z_s^{tot}) \\ 0, & \text{otherwise} \end{cases}$$

$$VaR_\alpha (Z_s^{tot}) = z\text{-score (}\alpha\text{)} \cdot \text{std (}Z_s^{tot}\text{)}$$

When the total costs of scenario $s$ exceed the sum of expected costs and $VaR_\alpha$, $\varphi$ indicates the cost in the worst scenarios. Otherwise, the value of $\varphi$ equals 0. In addition, the z-score is calculated using the norminv() function in MATLAB with $\alpha = 95\%$. std() represents the standard deviation function. Considering the parameter $CVaR_\alpha$, the
is used to generate a large number of scenarios, and the probability Carlo Simulation (MCS) method [3] to solve the uncertainty of this to find an exact or perfect solution. Therefore, this paper uses Monte for the stochastic nature of the above resources, it is almost impossible market prices, electric vehicle charging, and load demands. Due to certainty of multiple resources, such as renewable energy generation, averse to provide the safest solution in the worst extreme scenarios. Conversely, when \( \beta \) is merely equal to the expected cost and it’s a risk-neutral strategy. \( \beta \) takes values from 0 to 1. When \( \beta \) takes the value of 0, OF value is achieved by grouping statistically similar scenarios and reducing the number of scenarios imply higher accuracy, but also higher memory and time. Therefore, it is necessary to apply scenario-reduction strategies to eliminate scenarios with a low probability of occurrence. This is achieved by grouping statistically similar scenarios and reducing the number of scenarios. Similarly, the number of constraints is reduced. Details of this strategy can be found in [41].

C. Encoding of a Solution

This section describes the encoding method of solutions. It is known that depending on the problem characteristics, information is encoded into a solution with different dimensions to measure their performance. Undoubtedly, the information contained in the solution is related to resources, including generator active power, generator’s state, and EVs charge/discharge. Fig. 2 shows the encoding of a solution, in which all variables should be between the minimum and maximum values. Except for the state of the generator (represented by the binary state, where 1 indicates a connection to the SG and 0 indicates no connection), all variables are continuous and changed according to the specified boundaries. There are 24 periods in a solution. In each period, there are 570 sequentially repeated variables. Thus, the total number of variables in one solution is 24*570=13680, with 21 binary variables indicating the generator’s state, 21 continuous variables constituting the generator’s active power, and 500 continuous variables representing EVs charge/discharge. Load reduction uses 25-dimension continuous variables to represent its situation. The dimension number of the ESSs charge/discharge and electricity market is 2 and 1, respectively. It should be noted that DR is only assumed to be load reduction. For electricity market variables, assume that positive values are electricity sold in the market and negative values are electricity purchased.

V. METHODOLOGY

In our case study, the smart city medium voltage (MV) and distribution network (DN) located in the BISITE Laboratory in Salamanca, Spain is selected [42]. In this DN, there is a 30 megavolt-amperes (MVA) substation, 15 DG units (13 photovoltaic power plants and 2 wind farms), and four 1 MVar capacitor banks on bus 1 (these capacitor banks are not included in this problem because reactive power is not considered). In terms of consumption, the DN has 25 different loads, consisting of office buildings, residences, shopping centers, fire stations, and hospitals. In addition, the high penetration of EVs and renewable energy sources are considered in this case study. Smart city has seven charging stations to charge EVs, with a 50kW fast charging station at each connection point and four 7.2 kW slow charging stations. Fig. 3 shows the 13-bus 30 kV single-line diagram. First, 5,000 scenarios are generated, and are subsequently reduced to 150 scenarios using the GAMS/SCENRED [25] technique. Further, it is reduced to 15 scenarios to save computational time. Aggregators must manage various resources and meet consumption by purchasing electricity from external suppliers and purchasing/selling energy in the market. Table 1 considers the energy resource data related to the aggregator of the previous day’s formula in extreme scenarios. Noteworthy, m.u. means monetary unit. To solve the aggregator energy resource management problem, we propose SADEA. Details of its implementation will be introduced in the next section.
TABLE I
Energy resource data [35]

<table>
<thead>
<tr>
<th>Energy Resources</th>
<th>Prices (m.u./MWh)</th>
<th>Capacity (MW)</th>
<th>Forecasted (MW)</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wind</td>
<td>29-29</td>
<td>0.00-0.81</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Photovoltaic</td>
<td>31-31</td>
<td>0.30-3.07</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>External supplier</td>
<td>50-90</td>
<td>0.00-30.00</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ESSs</td>
<td>Charge</td>
<td>0.00-1.25</td>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Discharge</td>
<td>0.00-1.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EVs</td>
<td>Charge</td>
<td>0.01-0.05</td>
<td>0.01-2.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discharge</td>
<td>0.01-0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demand response</td>
<td>Load reduction</td>
<td>0.00-1.21</td>
<td>110</td>
<td>25</td>
</tr>
<tr>
<td>Load</td>
<td>0-0</td>
<td>0.01-2.38</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Electricity Market</td>
<td>44.78-156.91</td>
<td>0.00-1.25</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

Algorithm 1 The framework of SADEA

Input: self-adaptive control parameters $F, F_{CR}$, the total number of generations $G$

Output: $OF_{gbest}$

Begin:

/*Initialization*/
1. $x_c \leftarrow$ BoundaryRandomization $(1, D)$;
2. $x_{best} \leftarrow x_c$;
3. $OF_{gbest} \leftarrow f(x_c)$;
4. $\vartheta \leftarrow$ zeros $(2, D)$;
5. $gen = 1, e = 0, count = 0$;

/*Evolution*/
6. while $gen \leq G$ do
7.   $\{F', F_{CR}'\} \leftarrow jDE (F, F_{CR})$;
8.   $x_{new} \leftarrow TSAC (F', F_{CR}', \vartheta, x_c, x_{best}, count)$;
9.   $OF_{current} \leftarrow f(x_{new})$;
10. if $OF_{current} < OF_{gbest}$ then
11.     $OF_{gbest} \leftarrow OF_{current}$;
12.     $x_c \leftarrow x_{new}$;
13.     $x_{best} \leftarrow x_c$;
14.     $count = 0$;
15.     if $e < Ex_count$ then
16.       Put the solution into $\vartheta$;
17.       $e = e + 1$;
18.     else
19.       Use this solution $x_{new}$ to replace the worst one in $\vartheta$;
20.     end if
21.   end if
22.   count = count + 1;
23. end if
24. $gen = gen + 1$;
25. end while

End
seen in Lines 7-8, jDE is used to adjust parameters $F$ and $F_{CR}$, then, the three-stage adaptive collaboration strategy (TSAC) is used to obtain collaborative solutions based on the adjusted parameters $F'$ and $F'_{CR}$. As shown in Lines 10-24, elite selection and retention will be executed. If the current solution is better than the global optimal solution, the current solution will replace it, and $\vartheta$ will be updated. If the current solution is inferior to the global optimal solution, no operation will be taken, but the value of $\text{count}_1$ will be increased by 1.

Algorithm 2 jDE

Input: $F$, $F_{CR}$
Output: $F'$, $F'_{CR}$
Begin:
1: $F' = \begin{cases} F_{\text{lower}} + \text{rand}_1 \cdot F_{\text{upper}}, & \text{if } \text{rand}_2 < \tau_1; \\ F_1, & \text{otherwise} \end{cases}$
2: $F'_{CR} = \begin{cases} \text{rand}_3, & \text{if } \text{rand}_4 < \tau_2; \\ F_{CR}, & \text{otherwise} \end{cases}$
End

B. Self-adaptive Adjustment Strategy

The self-adaptive adjustment strategy removes the need for parameter adjustment and usually shows good performance for different types of problems. As a useful self-adaptive adjustment strategy, jDE was proposed in [16] and successfully applied in [12]. Results demonstrate that adjusting parameters is conducive to find promising solutions. Therefore, this paper uses jDE as the parameter-tuning method. In each generation, the calculation of $F$ and $F_{CR}$ is shown in Algorithm 2.

In the jDE strategy, $\text{rand}_j$, $j \in \{1,2,3,4\}$ are generated uniformly at random in the range $[0,1]$. $\tau_1$ and $\tau_2$ are probability factors. $F_{\text{lower}}$ and $F_{\text{upper}}$ represent the low and upper boundary vectors of $F$, respectively. In this paper, the same parameter settings as in [16] are used, i.e., $\tau_1 = \tau_2 = 0.1$. All values in $F_{\text{lower}}$, $F_{\text{upper}}$ are set to 0.1 and 0.9, respectively.

C. Three-stage Adaptive Collaboration Strategy

In the ERM problem under uncertainty, each variable value in the solution directly affects the final profit. Therefore, how to perturb these variables is important. The SADEA is designed based on DE strategies, which are used to perturb the current solution. The quality of the collaborative solutions, as part of the co-evolution, will have a large impact on the subsequent evolutions. However, in many existing EAs, the perturbation of the solution is not satisfactory. As a result, the collaboration to the current solution may not achieve the desired effect. Therefore, in this paper, we use a single-individual mechanism, which has strong local search ability (demonstrated in [43], [44]), and combine it with DE strategies to improve the effect of SADEA on the perturbation of variable values.

Algorithm 3 gives details of the three-stage adaptive collaboration strategy. Note that in the algorithm, $x_i$, $x_{i1}$, $x_{i2}$, and $x_{i3}$ denote the current and three other collaborative solutions, respectively; $\text{minbound}$ and $\text{maxbound}$ are boundaries composed of the minimum and maximum values of all variables. Three different DE strategies are designed to perturb the current solution to generate a new solution. The details of the DE strategies are given in Equations (8)-(10).

DE/target-to-perturbed best:

$$x_{\text{new}} = F_1 (x_{\text{best}} (F_2 + \text{rand}(D)) - x_c) + x_c + F_3 (x_{i1} - x_{i2})$$

(9)

As shown in above equations, DE strategies generate a new solution $x_{\text{new}}$ by perturbing $x_{i1}$, $x_{i2}$ and $x_{i3}$. $F_1$, $F_2$, and $F_3$ are three factors obtained from the jDE strategy, which are included in $F$ and take values in the range $[0,1]$; $x_{\text{best}}$ is the new solution. $\text{rand}(D)$ represents a $D$-dimensional random vector whose data is all in the range $[0,1]$; $x_{\text{best}}$ and $F$ are involved in the perturbation of variables in the solution $x_c$. As shown in Algorithm 3, the related counting parameters $\text{count}_1$ and $\text{count}_2$ are suggested to set to 6 and 7, respectively (see the experiment in Section VI). In addition, $R$ is a pre-set fixed value in $[0,1]$.

Algorithm 3 Three-stage adaptive collaboration strategy

Input: $F'$, $F'_{CR}$, the size of the historical archive $\vartheta$, the current solution $x_c$, the best solution $x_{\text{best}}$, $\text{count}$
Output: $x_{\text{new}}$
Begin:
$F = F'$, $F_{CR} = F'_{CR}$;
/*boundary randomization stage*/
1: if $\text{count} < \text{count}_1$ then
2: Randomly generate collaborative solutions $x_{r1}$, $x_{r2}$, and $x_{r3}$.
3: if $\text{rand}(\cdot) \leq R$ then
4: Generate $x_{\text{new}}$ using Eq.(9).
5: else
6: Generate $x_{\text{new}}$ using Eq.(10).
7: end if
/*knowledge-assisted collaboration stage*/
8: else if $\text{count}_1 \leq \text{count} < 11$ then
9: Permute randomly the solutions in $\vartheta$, and assign the first three solutions to $x_{r1}$, $x_{r2}$, and $x_{r3}$, respectively. If the size of $\vartheta$ is less than 3, then $x_{r3} = 0$.
10: if $\text{count} \geq \text{count}_2$ then
11: Generate $x_{\text{new}}$ using Eq.(8).
12: else
13: if $\text{rand}(\cdot) \leq R$ then
14: Generate $x_{\text{new}}$ using Eq.(9).
15: else
16: Generate $x_{\text{new}}$ using Eq.(10).
17: end if
18: end if
/*range restructuring stage*/
19: else
20: Calculate the difference between $\text{maxbound}$ and $\text{minbound}$, the range is obtained, and then $x_{r1} = (\text{range}/2 \cdot \text{i}(i-1)) + \text{minbound}$, $i = 1,2,3$.
21: Generate $x_{\text{new}}$ using Eq.(8).
22: end if
/*Crossover*/
23: $v = \text{rand}(D) < F_{CR}$;
24: Generate $v'$ according to the $v$;
25: $x_{\text{new}} = v' \cdot x_c + v \cdot x_{\text{new}}$;
26: $x_{\text{new}} \leftarrow \text{BoundaryControl}(x_{\text{new}})$;
End

DE/target-to-perturbed best2:

$$x_{\text{new}} = F_1 (x_{\text{best}} (F_2 + \text{rand}(D)) - x_c) + x_c + F_3 (x_{i1} - x_{i2}) + x_c + F_3 (x_{i1} - x_{i2})$$

(10)

Fig. 4 shows the DE strategies used in different stages with different colors. In the boundary randomization stage of SADEA
(Algorithm 3 Lines 2-7), first, three random solutions are generated randomly. Then, they are used by DE/target-to-perturbed best/1 and 2 strategies to perturb the current solution (red rectangular area). When several invalid evolutions occur in the boundary randomization stage, it moves to the next stage. In the knowledge-assisted collaboration stage (Algorithm 3 Lines 9-18), successfully updated solutions from recent generations are stored into \( \vartheta \). Then, DE/target-to-perturbed best/1 and 2 and DE/target-to-minimum boundary best strategies use \( x_{1} \), \( x_{2} \), and \( x_{3} \) to further enhance the local search of SADeA, and find promising solutions in the neighborhood of the current solution (blue rectangular area). The purpose of this design is to maintain a certain balance between global search and local search abilities, and not to completely improve the performance of one side at the expense of the other. In the range restructuration stage (Algorithm 3 Lines 20-21), DE/target-to-minimum boundary best is used to reduce the number of expensive calculations and the number of quality evaluations (green rectangular area). In the Crossover (Algorithm 3 Lines 23-25), the operation of generating random solutions is inspired by [12]. \( v \) and \( v' \) are vectors composed of 0 and 1, and the pair of vectors are reciprocal. If the value of certain variable in \( v \) is 1, the value of the corresponding variable in \( v' \) is 0, and vice versa.

D. Boundary Control Strategy

Ensuring the legality and correctness of the obtained solution is an extremely important aspect. In this paper, after the perturbation of DE strategies, variables in the solution are changed. However, some values may exceed the minimum or maximum bounds during the evolution. Obviously, this is not in accordance with the constraints of the ERM. In this case, it is necessary to use the boundary control strategy (Algorithm 3 Line 26) to contain the out-of-bounds variables within a reasonable range. This is a simple and effective strategy, which is also used in HyDE [12]. If variable values are greater than the maximum boundary values, maximum boundary values are assigned to these variables. If the variable values are less than the minimum boundary values, minimum boundary values are used to replace these values.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the effectiveness of SADeA is to be verified through the following experiments: parameter sensitivity study of SADeA, performance validation of the proposed strategies, and comparison with the state-of-the-art algorithms. In these experiments, all algorithms are coded in Matlab 2020 and performed on a 2.60 GHz Pentium processor, Intel Core i7 under the Windows 11 operating system. The line and violin charts are drawn by Origin; the convergence curve is drawn by Matlab, and other statistical charts are plotted using Excel. The details can be found in the following subsections.

A. Design of the Experiments

In our experiment, SADeA, along with its variants and peer algorithms, is executed with 20 independent runs. In each run, 15 different scenarios are integrated into a problem black box (the requirements for testing can be found on the website: http://www.gecad.isep.ipp.pt/ERM-competitions/2023-2/). When conducting experiments with peer algorithms, we choose two termination conditions, i.e., \( G = 3,000 \) and \( G = 5,000 \). The reason is to check the stability and robustness of the algorithms. All parameters and strategies are the same as those used in the original paper or competition. In addition, the RI is used as the performance metric. It is obtained by calculating the average value of \( N \) independent runs. A smaller RI value indicates a better algorithm. The calculation formula of RI is shown below:

\[
RI = \frac{1}{N} \sum_{i=1}^{N} OF_i
\]

where \( N \) is the number of executed runs, and \( N = 20 \). \( OF_i \) denotes the value of the \( OF \) obtained in the \( i \)-th run.

B. Sensitivity Study of Parameters

This subsection will conduct a sensitivity study on different parameters:

- \( F \), which affects collaborative solutions and evolutionary trends.
- \( F_{CR} \), which controls the degree of crossover.
- \( Ex\_count \), which indicates the size of \( \vartheta \).
- \( count\_1 \), which selects the execution stage in the three-stage adaptive collaboration strategy.
- \( count\_2 \), which determines the execution of DE/target-to-minimum boundary best.
- The random factor \( R \), which helps to choose the DE/target-to-perturbed best/1 or 2.

Fig. 5 shows the variation trend of each parameter. Based on the results, we ultimately choose \( F = 0.4 \) (meaning that \( F_1 \), \( F_2 \), and \( F_3 \) are set to 0.4), \( F_{CR} = 0.5, R = 0.4, Ex\_count = 2, count\_1 = 6 \), and \( count\_2 = 7 \). As shown in Fig. 5, too large or too small \( F \) and \( R \) will lead to the degradation of the algorithm’s performance. The \( OF \) value gradually increases with respect to \( Ex\_count \). This indicates that there is no need for too many historical solutions. The reason may be that storing too many solutions might lead to the wrong evolutionary direction of the current solution, thereby hampering the performance of the strategy. For other parameters, \( F_{CR}, count\_1, \) and \( count\_2 \), the variation of \( OF \) values is irregular. However, in terms of the variation, the settings of these parameters also affect the performance of SADeA.

C. The effectiveness of the proposed strategies

To investigate the impact of the proposed strategies, ablation experiments are conducted in this section. We use the following abbreviations to distinguish between variants of SADeA: \( V\text{-bound} \) represents that there is no boundary randomization stage (\( \vartheta' \) means ‘variant’, ‘\( \vartheta \)’ means ‘remove’). \( V\text{-Hist} \) and \( V\text{-Re} \) indicate the absence of knowledge-assisted collaboration and range restructuration stages, respectively. \( V\text{-DE1}, V\text{-DE2}, \) and \( V\text{-DE3} \) mean that the designed DE strategies (\( DE\text{target-to-minimum boundary best}, DE\text{target-to-disturbed best/1} \) and 2) are not used, respectively, SADeA contains all the strategies. The results are given in Table II and Fig. 6. Note that, in Table II bold font means that the corresponding algorithm obtains the best \( OF \) or \( RI \) values among all the algorithms. As shown in Table II, SADeA obtains 10 out of 20 best values, larger than the numbers of the best OFs obtained by \( V\text{-bound} \) (0/20), \( V\text{-Hist} \) (4/20), \( V\text{-Re} \) (3/20), \( V\text{-DE1} \) (1/20), \( V\text{-DE2} \) (1/20), and \( V\text{-DE3} \) (2/20). Moreover, SADeA achieves the smallest RI value. The results imply that the proposed strategies are effective in improving the performance of SADeA. Fig. 6 gives a violin chart drawn based on the numeric results in Table II. In addition to showing the statistics, the advantage of using a violin is that it clearly illustrates the overall
TABLE II

PERFORMANCE COMPARISONS (REGARDING OF AND RI) BETWEEN THE VARIANTS WHERE ONE OF THE PROPOSED STRATEGIES IS REMOVED

<table>
<thead>
<tr>
<th>Row</th>
<th>V-Bound</th>
<th>V-His</th>
<th>V-Re</th>
<th>V-DE1</th>
<th>V-DE2</th>
<th>V-DE3</th>
<th>SADEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>95865</td>
<td>41604</td>
<td>4884</td>
<td>41434</td>
<td>41440</td>
<td>41509</td>
<td></td>
</tr>
<tr>
<td>Run 2</td>
<td>82323</td>
<td>68288</td>
<td>45190</td>
<td>45829</td>
<td>46134</td>
<td>46264</td>
<td></td>
</tr>
<tr>
<td>Run 3</td>
<td>100560</td>
<td>45892</td>
<td>42049</td>
<td>48037</td>
<td>45362</td>
<td>42064</td>
<td></td>
</tr>
<tr>
<td>Run 4</td>
<td>94768</td>
<td>62568</td>
<td>43743</td>
<td>61084</td>
<td>40024</td>
<td>39594</td>
<td></td>
</tr>
<tr>
<td>Run 5</td>
<td>92227</td>
<td>40680</td>
<td>43784</td>
<td>61743</td>
<td>40092</td>
<td>40499</td>
<td></td>
</tr>
<tr>
<td>Run 6</td>
<td>68288</td>
<td>45892</td>
<td>41541</td>
<td>42049</td>
<td>48037</td>
<td>45362</td>
<td></td>
</tr>
<tr>
<td>Run 7</td>
<td>100560</td>
<td>45892</td>
<td>41541</td>
<td>42049</td>
<td>48037</td>
<td>45362</td>
<td></td>
</tr>
<tr>
<td>Run 8</td>
<td>132880</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 9</td>
<td>101830</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 10</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 11</td>
<td>124160</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 12</td>
<td>103790</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 13</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 14</td>
<td>92227</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 15</td>
<td>97227</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 16</td>
<td>111640</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 17</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 18</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 19</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>Run 20</td>
<td>112420</td>
<td>40680</td>
<td>41021</td>
<td>45859</td>
<td>42649</td>
<td>48874</td>
<td></td>
</tr>
<tr>
<td>RI</td>
<td>99407</td>
<td>42572</td>
<td>45247</td>
<td>43855</td>
<td>49845</td>
<td>43336</td>
<td></td>
</tr>
</tbody>
</table>

D. The Performance Comparison of All Algorithms

This subsection compares SADEA with the state-of-the-art algorithms for solving ERM problems: HyDE [12], CUMDANCauchy, CCSTG, Presto, HC2RCEDUMDA, and ReSaDE, which participated in the 2020, 2021, and 2022 competitions on evolutionary computation in the energy domain: smart grid scheduling applications. Among them, CUMDANCauchy, CCSTG, and ReSaDE won the first place in 2020, 2021, and 2022 competitions, respectively. The proposed SADEA adopts the parameters tuned in Subsection VI.B, while other

variant has the worst OF values. In addition, range restructuration and DE/target-to-disturbed best/2 have a large impact on the performance of SADEA. After removing these two strategies, the search ability of SADEA becomes poor, resulting in unstable OF values. Thus, it is necessary to use strategies with strong perturbation ability, range restructuration and DE/target-to-perturbed best/2 precisely to meet this requirement. For other strategies, they have a relatively small impact on SADEA. As can be seen from the Table II and Fig. 6 if these strategies are removed, the search performance of SADEA decreases to a certain degree.
algorithms use the parameters suggested by their developers. Tables III and IV list the OF values obtained by all algorithms in 20 independent runs. Like in Table II, the best OF and RI values are highlighted in bold. As shown in Tables III and IV, SADEA achieves the best results in all the runs, showing clearly its superiority over the peer algorithms.

To demonstrate the differences between these algorithms, we plotted the RI values, min/max OF values, and accumulated OF values in Figs. 7, 8, and 9 (G = 3,000) and Figs. 10, 11, and 12 (G = 5,000). The RI values (Figs. 7 and 10) are drawn in ascending order according to Tables III and IV. Each color rectangle indicates an algorithm. These plots provide an intuitive comparison of the overall performance of the algorithms. From these plots, it can be seen that SADEA has the best performance. The min/max values, as shown in Figs. 8 and 11, are drawn based on the minimum and maximum values in Tables III and IV. These two graphs reflect the fluctuation of OF values in all runs. It can be seen that although SADEA has the largest fluctuation, it still obtains smaller min/max values than the peer algorithms. The max value of the SADEA is smaller than the min value of the second-best algorithm HC2RCDUMDA. In addition, Figs. 9 and 12 give the accumulated OF values for SADEA and peer algorithms. It can be seen from these figures, SADEA achieves the minimum OF values and saves a significant amount of costs. Comparing Figs. 13 and 14, we can find that the solutions obtained by SADEA get more centralized when G is increased from 3,000 to 5,000, while solutions of other algorithms have small changes. It indicates that SADEA can continuously evolve, while other algorithms may fall into local optimum. From the overall perspective of the box plots, the OF values of SADEA are lower than other algorithms, further demonstrating the effectiveness of SADEA.

The reason for the good performance of SADEA may be that boundary randomization and range restructuration effectively enhance the algorithm’s global search ability. knowledge-assisted collaboration saves historical solutions and uses them to explore promising solutions in neighborhoods. This strategy aims at improving the diversity of solutions while stabilizing the local search ability. However, SADEA focuses its attention on a solution, which inevitably leads to more ‘precise’ evolution of the solution. It will lead to an issue: if the perturbation ability of SADEA is not strong, it is likely to fall into the local optimum. Therefore, it is necessary to design strategies with strong perturbation to improve the global search ability of SADEA, helping the solution jump out of the local optimum. The proposed DE strategies, combined with the three-stage adaptive collaboration strategy, achieve the desired goal. They overcome the shortcomings of traditional strategies in local search abilities. Furthermore, as demonstrated in Section VI-C, the proposed boundary randomization greatly improves the search ability of SADEA.

E. Statistical Validation and Convergence Curves

In this subsection, to check whether there exist significant differences between SADEA and peer algorithms, we performed statistical
### TABLE III

**Performance Comparisons (regarding OF and RI) between the peer algorithms when G = 3,000**

<table>
<thead>
<tr>
<th>Row</th>
<th>SADEA</th>
<th>HC2RCEDUMDA</th>
<th>ReSaDE</th>
<th>HyDE</th>
<th>CUMDANCauchy</th>
<th>Presto</th>
<th>CCSTG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>47771</td>
<td>72517</td>
<td>88159</td>
<td>101260</td>
<td>156100</td>
<td>178850</td>
<td>182710</td>
</tr>
<tr>
<td>Run 2</td>
<td>41648</td>
<td>66596</td>
<td>82663</td>
<td>94049</td>
<td>154800</td>
<td>178890</td>
<td>180750</td>
</tr>
<tr>
<td>Run 3</td>
<td>44239</td>
<td>69077</td>
<td>78367</td>
<td>97599</td>
<td>157740</td>
<td>178880</td>
<td>180770</td>
</tr>
<tr>
<td>Run 4</td>
<td>48659</td>
<td>66590</td>
<td>78959</td>
<td>87998</td>
<td>157680</td>
<td>178890</td>
<td>180830</td>
</tr>
<tr>
<td>Run 5</td>
<td>42024</td>
<td>65801</td>
<td>79883</td>
<td>94551</td>
<td>157680</td>
<td>178890</td>
<td>180950</td>
</tr>
<tr>
<td>Run 6</td>
<td>42190</td>
<td>74884</td>
<td>81738</td>
<td>99533</td>
<td>159350</td>
<td>178880</td>
<td>180630</td>
</tr>
<tr>
<td>Run 7</td>
<td>44115</td>
<td>69077</td>
<td>85055</td>
<td>91380</td>
<td>160650</td>
<td>178910</td>
<td>181270</td>
</tr>
<tr>
<td>Run 8</td>
<td>44561</td>
<td>66590</td>
<td>79195</td>
<td>94551</td>
<td>160690</td>
<td>178930</td>
<td>182290</td>
</tr>
<tr>
<td>Run 9</td>
<td>42541</td>
<td>70743</td>
<td>81151</td>
<td>91445</td>
<td>160690</td>
<td>178920</td>
<td>181350</td>
</tr>
<tr>
<td>Run 10</td>
<td>65734</td>
<td>68911</td>
<td>83439</td>
<td>98171</td>
<td>145450</td>
<td>178890</td>
<td>182140</td>
</tr>
<tr>
<td>Run 11</td>
<td>48271</td>
<td>68420</td>
<td>80783</td>
<td>87865</td>
<td>157790</td>
<td>178910</td>
<td>181270</td>
</tr>
<tr>
<td>Run 12</td>
<td>43324</td>
<td>66628</td>
<td>84231</td>
<td>102680</td>
<td>157520</td>
<td>178890</td>
<td>180530</td>
</tr>
<tr>
<td>Run 13</td>
<td>42568</td>
<td>73408</td>
<td>81357</td>
<td>98085</td>
<td>156100</td>
<td>178870</td>
<td>181350</td>
</tr>
<tr>
<td>Run 14</td>
<td>42446</td>
<td>65954</td>
<td>78773</td>
<td>90092</td>
<td>155070</td>
<td>178850</td>
<td>181190</td>
</tr>
<tr>
<td>Run 15</td>
<td>42059</td>
<td>67907</td>
<td>76726</td>
<td>93571</td>
<td>160720</td>
<td>178900</td>
<td>181500</td>
</tr>
<tr>
<td>Run 16</td>
<td>51161</td>
<td>73224</td>
<td>86613</td>
<td>84695</td>
<td>160170</td>
<td>178910</td>
<td>182090</td>
</tr>
<tr>
<td>Run 17</td>
<td>42817</td>
<td>69812</td>
<td>80934</td>
<td>84690</td>
<td>158380</td>
<td>178930</td>
<td>181400</td>
</tr>
<tr>
<td>Run 18</td>
<td>48806</td>
<td>67911</td>
<td>80608</td>
<td>86338</td>
<td>159340</td>
<td>178860</td>
<td>180870</td>
</tr>
<tr>
<td>Run 19</td>
<td>51921</td>
<td>70281</td>
<td>86703</td>
<td>92194</td>
<td>157190</td>
<td>178960</td>
<td>181910</td>
</tr>
<tr>
<td>Run 20</td>
<td>56333</td>
<td>67198</td>
<td>88030</td>
<td>100880</td>
<td>159660</td>
<td>178910</td>
<td>182350</td>
</tr>
</tbody>
</table>

#### RI
<table>
<thead>
<tr>
<th>Row</th>
<th>SADEA</th>
<th>HC2RCEDUMDA</th>
<th>ReSaDE</th>
<th>HyDE</th>
<th>CUMDANCauchy</th>
<th>Presto</th>
<th>CCSTG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>46659</td>
<td>69079</td>
<td>82615</td>
<td>93801</td>
<td>157110</td>
<td>178900</td>
<td>181440</td>
</tr>
</tbody>
</table>

### TABLE IV

**Performance Comparisons (regarding OF and RI) between the peer algorithms when G = 5,000**

<table>
<thead>
<tr>
<th>Row</th>
<th>SADEA</th>
<th>HC2RCEDUMDA</th>
<th>ReSaDE</th>
<th>HyDE</th>
<th>CUMDANCauchy</th>
<th>Presto</th>
<th>CCSTG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>41509</td>
<td>56722</td>
<td>81563</td>
<td>88676</td>
<td>152430</td>
<td>178760</td>
<td>182710</td>
</tr>
<tr>
<td>Run 2</td>
<td>40648</td>
<td>52413</td>
<td>85619</td>
<td>68679</td>
<td>154810</td>
<td>178750</td>
<td>180750</td>
</tr>
<tr>
<td>Run 3</td>
<td>42604</td>
<td>57521</td>
<td>79784</td>
<td>90745</td>
<td>152850</td>
<td>178750</td>
<td>180770</td>
</tr>
<tr>
<td>Run 4</td>
<td>39594</td>
<td>49503</td>
<td>82439</td>
<td>78569</td>
<td>153180</td>
<td>178760</td>
<td>180830</td>
</tr>
<tr>
<td>Run 5</td>
<td>40499</td>
<td>52351</td>
<td>83220</td>
<td>87263</td>
<td>153990</td>
<td>178740</td>
<td>180950</td>
</tr>
<tr>
<td>Run 6</td>
<td>48874</td>
<td>59645</td>
<td>80071</td>
<td>85397</td>
<td>153940</td>
<td>178730</td>
<td>180630</td>
</tr>
<tr>
<td>Run 7</td>
<td>39992</td>
<td>51477</td>
<td>74485</td>
<td>87365</td>
<td>155330</td>
<td>178750</td>
<td>181270</td>
</tr>
<tr>
<td>Run 8</td>
<td>40817</td>
<td>52670</td>
<td>83258</td>
<td>91225</td>
<td>152840</td>
<td>178730</td>
<td>182290</td>
</tr>
<tr>
<td>Run 9</td>
<td>40397</td>
<td>56896</td>
<td>80087</td>
<td>88930</td>
<td>153110</td>
<td>178760</td>
<td>180900</td>
</tr>
<tr>
<td>Run 10</td>
<td>40843</td>
<td>53481</td>
<td>80655</td>
<td>89126</td>
<td>151420</td>
<td>178730</td>
<td>182140</td>
</tr>
<tr>
<td>Run 11</td>
<td>43279</td>
<td>55064</td>
<td>68860</td>
<td>91198</td>
<td>155930</td>
<td>178770</td>
<td>181340</td>
</tr>
<tr>
<td>Run 12</td>
<td>40706</td>
<td>51754</td>
<td>80087</td>
<td>78894</td>
<td>155750</td>
<td>178740</td>
<td>181190</td>
</tr>
<tr>
<td>Run 13</td>
<td>39814</td>
<td>53854</td>
<td>76660</td>
<td>91797</td>
<td>143770</td>
<td>178720</td>
<td>181500</td>
</tr>
<tr>
<td>Run 14</td>
<td>41653</td>
<td>54845</td>
<td>80593</td>
<td>86023</td>
<td>153670</td>
<td>178760</td>
<td>182090</td>
</tr>
<tr>
<td>Run 15</td>
<td>42002</td>
<td>52120</td>
<td>84121</td>
<td>83458</td>
<td>148390</td>
<td>178790</td>
<td>181400</td>
</tr>
<tr>
<td>Run 16</td>
<td>40308</td>
<td>54500</td>
<td>85601</td>
<td>80524</td>
<td>153630</td>
<td>178730</td>
<td>180870</td>
</tr>
<tr>
<td>Run 17</td>
<td>41307</td>
<td>55376</td>
<td>85037</td>
<td>91209</td>
<td>155200</td>
<td>178730</td>
<td>181910</td>
</tr>
<tr>
<td>Run 18</td>
<td>45459</td>
<td>52775</td>
<td>82475</td>
<td>84846</td>
<td>148890</td>
<td>178760</td>
<td>182350</td>
</tr>
<tr>
<td>Run 19</td>
<td>41747</td>
<td>54094</td>
<td>80914</td>
<td>84773</td>
<td>152990</td>
<td>178750</td>
<td>181440</td>
</tr>
</tbody>
</table>

#### RI
<table>
<thead>
<tr>
<th>Row</th>
<th>SADEA</th>
<th>HC2RCEDUMDA</th>
<th>ReSaDE</th>
<th>HyDE</th>
<th>CUMDANCauchy</th>
<th>Presto</th>
<th>CCSTG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run 1</td>
<td>41747</td>
<td>54094</td>
<td>80914</td>
<td>84773</td>
<td>152990</td>
<td>178750</td>
<td>181440</td>
</tr>
</tbody>
</table>

---

**Fig. 12.** Accumulated OF values for SADEA and peer algorithms when G = 5,000

**Fig. 13.** Box plots for SADEA and the peer algorithms when G = 3,000
is not strong. Presto and CCSTG are trapped in a local optimum from the beginning of their execution and are unable to jump out of it. Compared with the above algorithms, SADEA can converge to a steady and better OF value more quickly. Overall, it exhibits a continuous evolution, albeit with a slower speed in later stages. The above convergence curves further confirm that SADEA outperforms other peer algorithms in improving the convergence. In summary, the proposed algorithm is effective for solving the ERM problem under uncertainty.

VII. CONCLUSION AND FUTURE WORK

In this paper, a self-adaptive collaborative differential evolution algorithm, named SADEA, is proposed to solve the ERM problem under uncertainty. In SADEA, a three-stage adaptive collaboration strategy is designed to generate different collaborative solutions to help the evolution of the current solution. In boundary randomization, collaborative solutions are generated to perturb the current solution. In the knowledge-assisted collaboration, the collaborative solutions are generated using historical archive, whose purpose is to help improve the global and local search abilities of SADEA. In addition, three different DE strategies are proposed to reduce the costs of aggregators in the ERM problem. The generated superior solutions directly participate in the next collaboration after boundary control and elite selection. In the experiments, sensitivity study and analysis of parameters are conducted. Then, the effectiveness of all proposed strategies is verified. Next, to verify the stability of the algorithm, SADEA is compared with six peer algorithms. The results demonstrate that SADEA outperforms peer algorithms in terms of solutions’ diversity and convergence.

We found that single-individual mechanism is useful for solving the continuous optimization problem, especially the ERM problem addressed in this paper. In addition, based on the single-individual mechanism, the performance of the algorithm can be further enhanced by using strategies that can improve the global search ability. Last but not the least, to maintain a balance between global and local search abilities, it is effective to use a multi-stage adaptive collaboration framework.

In the future, we will consider further optimizing the structure of SADEA, and will reduce the number of parameters with fixed settings. Moreover, there is value in improving the local and global search abilities of the algorithm, particularly in the late evolution stage. Also, it is interesting to incorporate micro search [46], neural networks [47], or reinforcement learning [48] into the EA algorithms. Finally, it is possible to make use of the advantages of other EA algorithms, e.g., Discrete Bee Colony Optimization Algorithms [49] and Meme algorithms [50].

Subsequently, to check the convergence of each algorithm, we select 12 runs and then draw their curves to verify the convergence performance. When $G = 3,000$, we randomly chose the run numbers: 2, 5, 9, 12, 15, and 17. When $G = 5,000$, we randomly chose the run numbers: 1, 4, 7, 11, 14, and 18. As shown in Figs. 15 and 16, the second best algorithm, HC2RCEUMDA, falls into a local optimum in later evolutions. Other peer algorithms, such as ReSaDE and HyDE, are not as strong as SADEA, though they do not fall into local optimum in most cases. CUMDANCauchy is often trapped in local optimum, and occasionally jumps out of this state. However, from the convergence trend, the overall performance of this algorithm

Subsequently, to check the convergence of each algorithm, we select 12 runs and then draw their curves to verify the convergence performance. When $G = 3,000$, we randomly chose the run numbers: 2, 5, 9, 12, 15, and 17. When $G = 5,000$, we randomly chose the run numbers: 1, 4, 7, 11, 14, and 18. As shown in Figs. 15 and 16, the second best algorithm, HC2RCEUMDA, falls into a local optimum in later evolutions. Other peer algorithms, such as ReSaDE and HyDE, are not as strong as SADEA, though they do not fall into local optimum in most cases. CUMDANCauchy is often trapped in local optimum, and occasionally jumps out of this state. However, from the convergence trend, the overall performance of this algorithm
Fig. 15. Convergence curves of SADEA and peer algorithms when $G = 3,000$

Fig. 16. Convergence curves of SADEA and peer algorithms when $G = 5,000$


**Haoxiang Qin** received B.S. and M.S. degrees from the School of Computer Science, LiaoCheng University, Shandong, China. He is currently pursuing the Ph.D. degree in software engineering at School of Software Engineering, South China University of Technology, Guangzhou, China.

His current research interests include intelligent optimization algorithms and their applications, such as hybrid flow shop scheduling, flexible job-shop scheduling, and electric system scheduling.

**Wenlei Bai (M’15)** received the B.S. degree in Electrical Engineering in 2009 from Southwest University for Nationalities, M.S. degree in Engineering Technology in 2011 from West Texas A&M University, and Ph. D. degree in Electrical and Computer Engineering in 2017 from Baylor University. He is currently a principal application developer at Oracle Energy and Water to conduct research and development on advanced distribution management system.

His research interests include power system planning and operation, electricity market, supply chain optimization, computational intelligence, and their applications to power systems, and modeling, simulation, and control of micro-grids with renewable and distributed energy sources.

**Yi Xiang** received the B.Sc. and M.Sc. degrees in mathematics from Guangzhou University, Guangzhou, China, in 2010 and 2013, respectively, and the Ph.D. degree in computer science from Sun Yat-sen University, Guangzhou, in 2018. He is currently an associate professor within the School of Software Engineering, South China University of Technology, Guangzhou. His current research interests include software product lines, search-based software engineering, and multi-objective optimization.

**Fangqing Liu** received the B.E. degree and Ph.D degree in software engineering from South China University of Technology, Guangzhou, China, in 2016 and 2021 respectively. He is currently a post-doctor in software engineering at School of Software Engineering, South China University of Technology, Guangzhou, China. His current research interests in AIGC including applications in automated software test case generation and image data augmentation.

**Yuyan Han** received the M.S. degree from the School of Computer Science, Liaocheng University, Liaocheng, China, in 2012, and the Ph.D. degree in control theory and control engineering from the China University of Mining and Technology, Xuzhou, China, in 2016.

Since 2016, she has been an associate professor with the School of Computer Science, Liaocheng University. She has authored over 30 refereed papers. Her current research interests include evolutionary computation, multiobjective optimization, and flow-shop scheduling.

**Ling Wang** received the B.Sc. in automation and Ph.D. degrees in control theory and control engineering from Tsinghua University, Beijing, China, in 1995 and 1999, respectively.

Since 1999, he has been with the Department of Automation, Tsinghua University, where he became a full professor in 2008. His current research interests include intelligent optimization and production scheduling. He has authored five academic books and more than 260 refereed papers.

Professor Wang is a recipient of the National Natural Science Fund for Distinguished Young Scholars of China, the National Natural Science Award (Second Place) in 2014, the Science and Technology Award of Beijing City in 2008, the Natural Science Award (First Place in 2003, and Second Place in 2007) nominated by the Ministry of Education of China. Professor Wang now is the Editor-in-Chief of International Journal of Automation and Control, and the Associate Editor of IEEE Transactions on Evolutionary Computation.