



An Efficient Multi-objective Approach Based on Golden Jackal Search for Dynamic Economic Emission Dispatch

Keyu Zhong^{1,2} · Fen Xiao³ · Xieping Gao^{1,4}

Received: 13 October 2023 / Revised: 1 March 2024 / Accepted: 5 March 2024 / Published online: 22 April 2024
© Jilin University 2024

Abstract

Dynamic Economic Emission Dispatch (DEED) aims to optimize control over fuel cost and pollution emission, two conflicting objectives, by scheduling the output power of various units at specific times. Although many methods well-performed on the DEED problem, most of them fail to achieve expected results in practice due to a lack of effective trade-off mechanisms between the convergence and diversity of non-dominated optimal dispatching solutions. To address this issue, a new multi-objective solver called Multi-Objective Golden Jackal Optimization (MOGJO) algorithm is proposed to cope with the DEED problem. The proposed algorithm first stores non-dominated optimal solutions found so far into an archive. Then, it chooses the best dispatching solution from the archive as the leader through a selection mechanism designed based on elite selection strategy and Euclidean distance index method. This mechanism can guide the algorithm to search for better dispatching solutions in the direction of reducing fuel costs and pollutant emissions. Moreover, the basic golden jackal optimization algorithm has the drawback of insufficient search, which hinders its ability to effectively discover more Pareto solutions. To this end, a non-linear control parameter based on the cosine function is introduced to enhance global exploration of the dispatching space, thus improving the efficiency of finding the optimal dispatching solutions. The proposed MOGJO is evaluated on the latest CEC benchmark test functions, and its superiority over the state-of-the-art multi-objective optimizers is highlighted by performance indicators. Also, empirical results on 5-unit, 10-unit, IEEE 30-bus, and 30-unit systems show that the MOGJO can provide competitive compromise scheduling solutions compared to published DEED methods. Finally, in the analysis of the Pareto dominance relationship and the Euclidean distance index, the optimal dispatching solutions provided by MOGJO are the closest to the ideal solutions for minimizing fuel costs and pollution emissions simultaneously, compared to the latest published DEED solutions.

Keywords Dynamic economic emission dispatch · Multi-objective optimization · Golden jackal · Euclidean distance index

✉ Fen Xiao
xiaof@xtu.edu.cn

✉ Xieping Gao
xpgao@hunnu.edu.cn

Keyu Zhong
zhongkyu@126.com

- ¹ Key Laboratory of Computing and Stochastic Mathematics of Ministry of Education, Hunan Normal University, Changsha 410081, China
- ² School of Mathematics and Statistics, Hunan Normal University, Changsha 410081, China
- ³ School of Computer Science, Xiangtan University, Xiangtan 411105, China
- ⁴ Hunan Provincial Key Laboratory of Intelligent Computing and Language Information Processing, Hunan Normal University, Changsha 410081, China

1 Introduction

As the social economy continued to develop, the demand for electricity in various daily activities and production sharply increased. This high demand for electricity rendered Conventional Economic Emission Dispatch (CEED) inadequate in adapting to actual electricity's dynamic load changes [1]. Dynamic Economic Emission Scheduling (DEED) emerged based on CEED as a result. The DEED problem is currently a hot research topic in the field of power system optimization, which is essentially a dynamic multi-objective optimization problem. The primary goal of DEED is to schedule the output power of different units within a specific time range, to achieve optimal control of power generation costs and emissions [2]. The significant feature of DEED is that the power scheduling of the previous time

period will directly affect the scheduling of the next time period, which is dynamic. In addition, DEED must also consider the ramp limit of the generator in adjacent time periods in addition to meeting the constraints of unit output and power balance, because excessive power rise or fall will seriously affect the service life of the generator rotor [3]. Considering the valve point effect, network transmission loss, unit ramp limit, time variability of the scheduling process and conflicts between objectives in DEED, the DEED problem presents complex characteristics such as high dimension, nonlinear, non-convex, non-differentiable and non-smooth.

There are two main categories of methods for solving DEED: the first category is traditional mathematical methods, such as quadratic programming [4], linear programming [5], and dynamic programming [6]. These methods usually require the problem model to be smooth and derivable, so it is difficult for traditional mathematical methods to solve complex DEED problems efficiently. The second category of methods is metaheuristic algorithms. These methods, with their advantages such as collective search patterns and no specific requirements for the objective function of the optimization problem, are more suitable for solving multi-objective optimization problems like DEED. To address this challenge, Guo et al. [7] proposed a new multi-objective group search optimizer to solve the DEED problem. To obtain the best compromise solution, a Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) method was used to select the best compromise solution from a series of non-dominated optimal solution sets. Arul et al. [8] proposed three different variants of the harmony search algorithm based on chaotic mapping and differential mutation strategies to solve this problem. Furthermore, the Fuzzy decision-making method was utilized to choose the best compromise solution. Mason et al. [9] compared the performance of the basic Particle Swarm Algorithm (PSO) and the PSO with Avoidance of Worst Locations (PSOAWL) for solving the DEED problem. They researched the performance of PSO and its variants under DEED constraints by using different topology structures. Li et al. [10] improved the search performance of the Tunicate Swarm Algorithm for solving DEED problems in power systems of different scales by using tent mapping and levy flight strategy. The Tent mapping strategy provided a high-quality initial population for subsequent optimization processes, and the Levy flight strategy effectively improved the search direction of the population, accelerating the convergence speed of the algorithm. Experimental results in different cases indicated that the enhanced Tuned Tunicate Swarm algorithm achieved solutions with lower fuel costs and emissions. Mason et al. [11] proposed a new multi-objective optimization method, called MONNDE, which combines differential evolution algorithm with neural

networks to solve DEED problems. Differential evolution algorithm is mainly used to train neural networks and help find better Pareto-optimal solutions. The comprehensive results demonstrated that MONNDE obtained better compromise solutions in optimizing both fuel costs and emissions, outperforming other algorithms. Zhu et al. [12] presented an improved multi-objective evolutionary algorithm based on decomposition for this problem. Shao et al. [13] suggested a multi-objective proximal policy optimization to tackle DEED problem. Nourianfar and Abdi [14] combined the time-varying acceleration coefficient and particle swarm algorithm with the exchange market algorithm for DEED problem. Wu et al. [15] presented an Improved Non-Dominated Ranking Genetic Algorithm III (INSGA-III) for this problem. A crossover operator based on normal distribution was introduced to enhance the exploration and exploitation performance of NSGA-III. In addition, an angle-based association strategy was employed to achieve a more uniform distribution of the solution sets. Qiao et al. [16] proposed a new nondominated sorting differential evolution algorithm with an efficient constraint handling mechanism to solve the DEED problem. Roy et al. [17] presented a new multi-objective chemical reaction optimization algorithm for this problem. The convergence speed was significantly improved by mixing this algorithm with differential evolution. In Ref. [18], the Multi-Objective Multiverse Optimization Algorithm (MOMVO) was applied for the DEED problem. The authors adopted an effective constraint handling mechanism to handle the various constraints in DEED, which can well confine candidate solutions within the feasible domain boundaries.

Although the above multi-objective optimization algorithms perform well in solving the DEED problem, their optimization performance often fails to achieve the expected results when dealing with power scheduling with dynamic load changes. This is mainly because they cannot effectively balance the convergence and diversity of non-dominated optimal scheduling solutions. Most DEED methods focus on improving the convergence of non-dominated dispatching solutions while neglecting their diversity, which may lead them to fall into local optima. Overemphasizing the diversity of non-dominated solutions while overlooking the convergence of the solution set can produce many low-quality schedules. To obtain high-quality non-dominated optimal scheduling solutions, it is crucial to effectively balance the convergence and diversity of the non-dominated optimal scheduling solution set. Therefore, designing an effective trade-off mechanism between the convergence and diversity of non-dominated optimal solutions is essential for efficiently solving the DEED problem. If too much emphasis is placed on the diversity of non-dominated solutions while neglecting the convergence of the solution set, many poor-quality solutions may be generated. To obtain high-quality

non-dominated optimal solutions, it is crucial to effectively balance the convergence and diversity of the solution set. Therefore, designing an effective mechanism to balance convergence and diversity is essential for efficiently solving the DEED problem.

The Golden Jackal Optimization (GJO) algorithm is a swarm intelligence algorithm proposed by Nitish Chopra and Muhammad Mohsin Ansari [19], inspired by the hunting behavior of golden jackals. The GJO algorithm has gained wide application in various problems due to its simple structure, general principles, and high performance. Ref. [19] used the GJO algorithm to solve the economic load dispatch problem, resulting in lower fuel costs compared to other published methods. Ref. [20] proposed a GJO algorithm based on opposition-based learning to solve the skin cancer image segmentation problem. Rezaie et al. [21] combined particle swarm optimization algorithm with GJO algorithm to solve the parameter identification problem of proton exchange membrane fuel cells. Ref. [22] proposed a GJO algorithm based on optimal control strategy to solve the operation optimization problem of hybrid wave energy and photovoltaic system boost stations. Zhang et al. [23] introduced the elite opposition-based learning and simplex technique to enhance the search performance of GJO, which was applied to adaptively identify infinite pulse response systems. While GJO algorithm outperformed other algorithms in solving engineering problems such as economic load dispatch problem, it also suffers from insufficient search in the decision space [19, 20, 22]. This is mainly attributed to the ineffective control of the linear parameter, the energy descent factor, on the optimization process of GJO, leading to the population's insufficient exploration in many promising areas. However, the complex characteristics of the DEED problem result in a complex and variable search space. Therefore, adjusting the energy factor to guide the population search strength in the optimization process is necessary to improve the algorithm's optimization efficiency on DEED problems.

The "no free lunch" theorem [24] suggests that while a specific optimization method may perform well in solving one type of optimization problem, it may exhibit poor performance in solving different types of optimization problems. In other words, it is always possible to design new algorithms to efficiently solve real-world optimization problems. In this paper, we propose an effective multi-objective algorithm based on GJO, called MOGJO, to better address DEED. To improve the quality of the solution set, we propose a leader selection mechanism based on an elite selection strategy and Euclidean distance measurement method. This mechanism selects the best solution as the jackal leader from the non-dominated optimal solutions, prompting the algorithm to search for better solutions in a wide space. The elite selection

method prioritizes retaining solutions in the non-crowding region by evaluating the crowding degree of each solution in the archive to ensure diversity in the non-dominated optimal solutions. The Euclidean distance measurement method selects the best solution as the leader to guide the algorithm towards reducing fuel costs and emissions, thus improving the convergence of the non-dominated optimal solutions. To overcome the insufficient search problem of GJO, we use a non-linear energy descent factor based on the cosine function to enhance GJO's adaptive exploration of the dispatching space and fully explore non-dominated optimal solutions existing in promising areas. The nonlinear parameter guides the population to search in the direction of multiple optimal solutions by controlling the magnitude of energy descent, effectively improving the optimization efficiency of the algorithm for the DEED problem. In MOGJO, the nonlinear parameter and leader selection mechanism are both indispensable. The former focuses on quickly finding non-dominated optimal solutions in the dispatching space, while the latter is responsible for improving the convergence and diversity of the Pareto solution set. In the experiment, we first validate the multi-objective optimization performance of the proposed MOGJO algorithm on the latest CEC test functions and compared it with the state-of-the-art algorithms. Then, we use MOGJO to solve the DEED problem, and compare the provided best compromise solutions with the best compromise dispatching solutions reported in other published literature using Pareto dominance relationship and Euclidean distance index. The experimental results indicate that the proposed MOGJO algorithm outperforms the state-of-the-art algorithms for the DEED problems and can effectively handle the conflict between the two objectives of fuel cost and pollution emission.

The main contributions of this paper are as follows:

- A new multi-objective optimization approach based on the golden jackal search is proposed to tackle the dynamic multi-objective economic emission dispatch problems.
- A leader selection mechanism based on the elite selection strategy and the Euclidean distance index method is designed to adaptively improve the diversity and convergence of the non-dominated optimal dispatching solution set.
- Introducing a nonlinear control parameter based on the cosine function can enhance the thorough exploration of the dispatching space and improve the efficiency of MOGJO in searching for the optimal solutions.
- The performance of MOGJO is tested on the latest CEC2020 multi-objective benchmarking functions. Compared to the state-of-the-art multi-objective

algorithms, MOGJO provides Pareto optimal solutions and Pareto optimal fronts with better convergence and diversity.

- The best compromise solutions provided by MOGJO are compared with the latest published results in terms of Pareto dominance relationship and Euclidean distance index for DEED problems under various cases. The comprehensive results show that MOGJO outperforms the state-of-the-art DEED methods.

The remainder of this paper is organized as follows: Sect. 2 presents an introduction to the DEED problem model and constraints. Section 3 describes the basic GJO algorithm and the proposed MOGJO algorithm in detail. Section 4 presents the performance validation of MOGJO on the latest CEC test functions. The simulation results of MOGJO to solve the multi-objective DEED problem are given in Sect. 5. Finally, Sect. 6 concludes this work and envisions the future research.

2 Mathematical Model of DEED

2.1 Objective Functions

The DEED problem aims to minimize power generation cost and pollutant gas emission simultaneously as much as possible. The generation cost mainly consists of the fuel cost of the generators involved. The cost function considers quadratic and sinusoidal functions to model the valve point effect of the thermal generator. Due to the valve point effect, the cost function exhibits more complex nonconvex characteristics. The emission of sulfur oxide, nitrogen dioxide and carbon dioxide are modeled as a nonlinear function with quadratic and exponential terms. The bi-objective optimization function of DEED is described as follows [18]:

$$f_1(x) = \sum_{i=1}^T \sum_{i=1}^{NT} a_i P_{i,t}^2 + b_i P_{i,t} + c_i + \left| d_i \sin \left\{ e_i (P_i^{\min} - P_{i,t}) \right\} \right|, \quad (1)$$

$$f_2(x) = \sum_{i=1}^T \sum_{i=1}^{NT} \left[k_i P_{i,t}^2 + l_i P_{i,t} + m_i + n_i \exp(q_i P_{i,t}) \right], \quad (2)$$

where $f_1(\cdot)$ denotes the fuel cost function and $f_2(\cdot)$ represents the pollutant gas emission function. NT denotes the number of thermal power generators and T refers to 24 dispatch periods. a_i, b_i, c_i are the fuel cost coefficients and d_i, e_i are the valve point coefficients for the i -th generator. $P_{i,t}$ denotes the output power of the i -th generator at time t . P_i^{\min} refers to the lower bound for the output power of the i -th generator. k_i, l_i, m_i, n_i, q_i are the emission coefficients of the i -th generator.

2.2 Power Balance Constraint

The mathematical description of the power balance constraint for the DEED problem is as follows [16]:

$$\sum_{i=1}^{NT} P_{i,t} = P_{D,t} + P_{L,t}, \quad (3)$$

where $P_{D,t}$ stands for the power system load demand at time t . $P_{L,t}$ refers to the network transmission power loss, which can be calculated by Krons formula [18]:

$$P_{L,t} = \sum_{i=1}^{NT} \sum_{j=1}^{NT} P_{i,t} B_{ij} P_{j,t} + \sum_{i=1}^{NT} P_{i,t} B_{0i} + B_{00}, \quad (4)$$

where B_{ij}, B_{0i}, B_{00} are the network transmission loss coefficients.

2.3 Generator Output Power Constraints

The output constraint of the generator can be described as [13]:

$$P_i^{\min} \leq P_{i,t} \leq P_i^{\max}, \quad (5)$$

where P_i^{\max} indicates the upper bound for the output power of the i -th generator.

2.4 Generator Ramp Limit Constraint

In the DEED problem, the magnitude of the power increase or decrease of the generator at two adjacent time periods cannot exceed the ramp limit [18].

$$\begin{cases} P_{i,t} - P_{i,t-1} \leq UR_i \Delta t \\ P_{i,t-1} - P_{i,t} \leq DR_i \Delta t \end{cases} \quad (6)$$

where UR_i and DR_i denote the ramp up and ramp down limits for the i -th generator, respectively. Δt is the dispatch period gap, which is normally set to 1.

2.5 Constraint Handling

For the ramp limit constraint in DEED, the upper and lower bounds of the output power for each generator are restricted to a security range [18].

$$P_{i,t}^{\min} = \begin{cases} P_i^{\min}, & \text{if } t = 1 \\ \max(P_i^{\min}, P_{i,t-1} - DR_i), & \text{if } t > 1 \end{cases},$$

$$P_{i,t}^{\max} = \begin{cases} P_i^{\max}, & \text{if } t = 1 \\ \min(P_i^{\max}, P_{i,t-1} + UR_i), & \text{if } t > 1 \end{cases}. \quad (7)$$

The real power of the system must be balanced with the power demand and the line losses. To handle this constraint, the violation (*PRD*) between the actual power and the latter two is first calculated. $|PRD|$ must be less than a tiny positive

number (*vio*) to guarantee that the constraint is handled efficiently. When $|PRD|$ is larger than *vio*, the constraint handling process is as described in Pseudocode 1.

1. Generate a random number r in range $[1, NT]$
2. $P_{i,t} = P_i^{\min} - rand * (P_i^{\min} - P_i^{\max})$, where $i \neq r$
3. $P_{r,t} = P_{D,t} - \sum_{i \in \alpha_n} P_{i,t}$, $\alpha_n = 1$ to NT excluding r
4. Boundary constraint handling for $P_{r,t}$
5. $PRD = P_{D,t} - \sum_{i=1}^{NT} P_{i,t}$ // Calculate the violation between real power and $P_{D,t}$
6. If $|PRD| > vio$, repeat steps 1-5 until $|PRD| \leq vio$
7. Calculate the line loss $P_{L,t}$ according to Eq. (4)
8. Generate a random number r in range $[1, NT]$
9. $P_{i,t} = P_i^{\min} - rand * (P_i^{\min} - P_i^{\max})$, where $i \neq r$
10. $P_{r,t} = P_{D,t} + P_{L,t} - \sum_{i \in \alpha_n} P_{i,t}$, $\alpha_n = 1$ to NT excluding r
11. Boundary constraint handling for $P_{r,t}$
12. $PRD = P_{D,t} + P_{L,t} - \sum_{i=1}^{NT} P_{i,t}$ // The violation between real power and $P_{D,t} + P_{L,t}$
13. If $|PRD| > vio$, repeat steps 7-12 until $|PRD| \leq vio$

Pseudocode 1 Constraint violation handling process

2.6 Selection of Compromise Solution

In a multi-objective optimization algorithm, a single run will produce multiple solutions that do not dominate each other. For decision makers, without clear choice preferences, it is difficult to select the best solution from a large number of non-dominated solution sets to support their decision-making process. To facilitate the decision-making to select an optimal trade-off solution from Pareto optimal solutions, we adopt the Fuzzy decision-making method [18]. The specific steps of the method are as follows.

(1) Calculate the satisfaction for each objective of the Pareto optimal solution:

$$\delta_{k,i} = \begin{cases} 1, & f_{k,i} \leq f_k^{\min} \\ \frac{f_k^{\max} - f_{k,i}}{f_k^{\max} - f_k^{\min}}, & f_k^{\min} \leq f_{k,i} \leq f_k^{\max} \\ 0, & f_{k,i} \geq f_k^{\max} \end{cases}, \quad (8)$$

where $\delta_{k,i}$ denotes the satisfaction for the k -th objective of the i -th solution. f_k^{\max} and f_k^{\min} refer to the maximum and minimum values of the k -th objective in the Pareto optimal front, respectively.

(2) Normalize the overall satisfaction of each Pareto optimal solution as follows:

$$\delta_i = \frac{\sum_{k=1}^M \delta_{k,i}}{\sum_{i=1}^I \sum_{k=1}^M \delta_{k,i}}, \quad (9)$$

where M and I are the number of objectives and Pareto optimal solutions, respectively.

(3) Select the solution with maximum δ_i as the best compromise solution.

3 Multi-objective Golden Jackal Algorithm

3.1 Basic Golden Jackal Algorithm

The GJO is a new population-based algorithm [19], which simulates the hunting behavior of golden jackals in nature and has strong search performance. The jackal population can be classified into male jackals and female jackals, which have a well-defined labor division in the population. The male jackal is responsible for hunting, while the female jackal follows the male jackal to search for prey. The specific steps of the GJO are described below.

3.1.1 Initialization Phase

The golden jackal population can be initialized as:

$$X_0 = X_{\min} + \mathbf{rand}(X_{\max} - X_{\min}), \quad (10)$$

where \mathbf{rand} is a random vector in range [0, 1]. X_{\max} and X_{\min} denote the upper and lower bounds of the decision variables, respectively. After initialization, the golden jackal population forms the \mathbf{Prey} matrix.

$$\mathbf{Prey} = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,D} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,D} \\ \vdots & \vdots & \vdots & \vdots \\ X_{N,1} & X_{N,2} & \cdots & X_{N,D} \end{bmatrix}, \quad (11)$$

where N is the number of jackals and D denotes the size of the decision variable. The male jackal is the most skilled hunter, and the female jackal follows. Each male and female jackal together grouped as a jackal pair.

3.1.2 Searching for Prey

In search for prey, male jackals lead the hunting behavior of the jackal population, and female jackals follow the males. In this phase, $|E| > 1$, which corresponds to the exploration of the algorithm. The position update formula of the jackal pair is:

$$X_1(t) = X_M(t) - E \cdot |X_M(t) - \mathbf{LF} \cdot \mathbf{Prey}(t)|, \quad (12)$$

$$X_2(t) = X_{FM}(t) - E \cdot |X_{FM}(t) - \mathbf{LF} \cdot \mathbf{Prey}(t)|, \quad (13)$$

where $X_M(t)$ and $X_{FM}(t)$ refer to the positions of the male and female jackals at the t -th iteration, respectively. $\mathbf{Prey}(t)$ denotes the prey position and LF represents the Levy flight function, which is described in [19]. E is the energy factor that is used to control the exploration and exploitation of the algorithm and can be calculated by:

$$E = E_0 * E_1, \quad E_0 = 2r - 1, \quad E_1 = s_1 * (1 - t/T), \quad (14)$$

where E_0 means the initial energy state, r is a random number in range [0,1]. E_1 represents the energy decline of the prey, s_1 is a constant in value of 1.5, t and T indicate the number of current iterations and the total number of iterations, respectively.

3.1.3 Hunting Prey Phase

In hunting prey phase, $|E| < 1$ represents the exploitation of GJO. Jackal pair co-attacks the prey in this phase, which is described mathematically as

$$X_1(t) = X_M(t) - E \cdot |\mathbf{LF} \cdot X_M(t) - \mathbf{Prey}(t)|, \quad (15)$$

$$X_2(t) = X_{FM}(t) - E \cdot |\mathbf{LF} \cdot X_{FM}(t) - \mathbf{Prey}(t)|. \quad (16)$$

Based on the positions of male and female jackals, the position update formula for the jackal population is

$$X(t+1) = \frac{X_1(t) + X_2(t)}{2}. \quad (17)$$

3.2 MOGJO

The GJO algorithm exhibits excellent search performance for single objective optimization problems, but it cannot be directly employed to solve multi-objective optimization problems. To this end, the GJO is required to be converted by introducing multi-objective optimization mechanisms to accommodate the conflicts between different objectives.

3.2.1 Archive

In order to tackle the multi-objective optimization problem, an archive component is first integrated into the GJO. The archive is mainly utilized to store the non-dominated optimal solutions found so far. The update rules of the archive are as follows: first, if a new solution dominates at least one solution in the archive, the new solution should be stored in the archive instead of the dominated solution; second, if at least one solution in the archive dominates the new solution, the new solution cannot be stored in the archive; third, if the new solution and the solutions in the archive are non-dominated, the new solution should be stored in the archive. With these three rules, the up-to-date non-dominated solutions can be saved efficiently and promptly. Once the number of solutions in the archive reaches the maximum limit, the new solutions that satisfy the access rules are unable to enter the archive. To this end, we make it

possible for new solutions to enter the archive by calculating the crowding degree of each non-dominated solution in the archive and randomly eliminating a solution according to the crowding degree with a roulette wheel method. Taking each non-dominated solution as the center, the number of neighboring solutions within the radius d is calculated as its crowding degree. A larger number of neighboring solutions indicates the more crowded the region where the solution is located. The radius d is calculated as follows.

$$\vec{d} = (\overline{\max} - \overline{\min}) / AS, \tag{18}$$

where AS denotes the size of the archive. $\overline{\max}$ and $\overline{\min}$ refer to the two vectors that store the maximum and minimum values of each objective, respectively. Based on the crowding degree of each solution in the archive, a roulette wheel method is employed to calculate the probability of each solution being eliminated.

$$Prob_i = NS_i / CT, \tag{19}$$

where NS_i is the number of neighboring solutions for the i -th solution in the archive and CT is a constant larger than 1. For a solution, the more crowded it is, the higher the probability of being eliminated. It helps to maintain the diversity of non-dominated solutions.

3.2.2 Leader Selection Mechanism

In the GJO, male jackal is the top hunter, followed by female jackal. Therefore, the selection of the best jackal pair is essential for the evolution of the whole population. In single-objective optimization, the leader individual can be obtained directly by choosing the best fit individual. However, in multi-objective

optimization, the leader individual can not be selected by comparing the fitness values due to the conflict between objectives. Given the guiding role of the leader on the population search, its superiority must be guaranteed. To this end, we propose a leader selection mechanism to choose the best jackal pair in order to enhance the ability of the algorithm to cope with multi-objective optimization problem. For the best male jackal, an elite selection and Euclidean distance index-based approach is used to conduct adaptive selection from the archive. The elite selection method focuses on the selection of the Pareto solution with the minimum crowding degree in the archive as the best male jackal individual solution. It is mandatory for the algorithm to emphasize the diversity of solutions in the optimization process in order to obtain high-quality solutions for decision makers' deliberation. The Pareto solution with the minimum crowding degree also represents the best sparseness, which can contribute to improve the diversity of the population. In addition, the convergence of the algorithm largely depended on the superiority of the individual leader. In theory, the solution in the archive that simultaneously has the minimum value of each objective is optimal, but the conflict between objectives dictates that each objective cannot reach the optimum at the same time. Nevertheless, we assume the existence of such an ideal point and calculate the Euclidean distance ED between the solution in the archive and this ideal point. The smaller the value of ED , it means that the solution is closer to the ideal state, and the solution is exactly the one we attempt to obtain, as shown in Fig. 1.

The ED of each non-dominated solution is calculated as

$$ED_i = \sqrt{(f_1^{\min} - f_{1,i})^2 + (f_2^{\min} - f_{2,i})^2}, \tag{20}$$

where f_1^{\min} and f_1^{\max} denote the minimum values of the 1st and 2nd objectives of the existing solutions in the archive, respectively. $f_{1,i}$ and $f_{2,i}$ represent the two objective values of the i -th solution in the archive, respectively. For the best female jackal, a roulette wheel method is employed to select randomly from the archive. The solution with low crowding degree has a higher probability to be selected, and the probability of selection formula is

$$P'_i = CT / NS_i. \tag{21}$$

3.2.3 Nonlinear Control Parameter

In GJO, the energy descent factor E_1 is a linear parameter, which is likely to lead to insufficient search in the complex search space [19, 20, 22]. In order to solve this drawback, we introduce a nonlinear control parameter based on cosine function instead of the linear one to control the algorithm to fully search in various search areas and find more

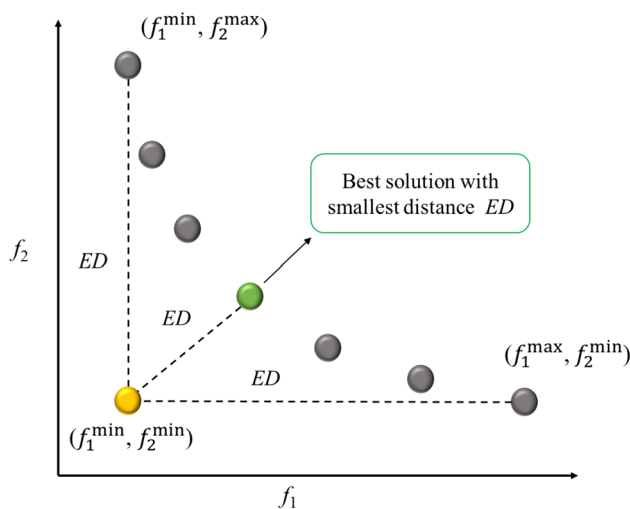


Fig. 1 Best solution selected by Euclidean distance index method

high-quality solutions. The calculation formula of nonlinear parameter is

$$E_1 = s_{\max} - (s_{\max} - s_{\min}) * \cos((1 - t/T) * \pi/2), \quad (22)$$

where s_{\max} and s_{\min} denote the maximum and minimum energy values of the prey, respectively, and in this paper, they are set to 2 and 1. With the introduction of non-linear control parameters, the algorithm finds more high-quality solutions by adequately exploring various search regions. These solutions or their neighbors are highly likely to be stored in the archive as non-dominated optimal solutions, which significantly improve the optimization efficiency of the algorithm.

Indeed, there is a coupling relationship between the leader selection mechanism and the non-linear control parameter. In MOGJO, the non-linear control parameter can adapt to

control the search intensity of the algorithm to find more quality feasible solutions. From these, the algorithm can exploit the local domain to obtain optimal solutions. These found high-quality solutions are most likely to be saved in the archive, since they are largely non-dominated by other found solutions. The leader selection mechanism takes care of improving the convergence and diversity of the non-dominated optimal solution set in the archive, because in the multi-objective optimization algorithm, the outputs must be multiple different non-dominated solutions so that the decision maker can make a variety of choices. Consequently, MOGJO can efficiently tackle complex multi-objective DEED problems with the assistance of both the non-linear control parameter and the leader selection mechanism. The pseudo code of MOGJO is shown in Algorithm 1, and the flowchart of MOGJO solving DEED is shown in Fig. 2.

Algorithm 1 MOGJO

1. Initialize the population and parameters
2. **while** $t < T$
3. Evaluate the fitness of the initial population
4. Find all non-dominated solutions from population
5. Update archive based on access rules
6. Update archive based on Eq. (18) and Eq. (19) if archive is full
7. **if** $rand < 0.5$
8. Update best male jackal position based on elite selection
9. **else**
10. Update best male jackal position based on Eq. (20)
11. **end if**
12. Update best female jackal position based on Eq. (21)
13. **for** $i = 1$ to N
14. Update the energy factor based on Eq. (14) and Eq. (22)
15. **if** $|E| \geq 1$
16. Update the jackal's position based on Eq. (12), Eq. (13) and Eq. (17)
17. **else if** $|E| < 1$
18. Update the jackal's position based on Eq. (14), Eq. (16) and Eq. (17)
19. **end if**
20. **end for**
21. **end while**

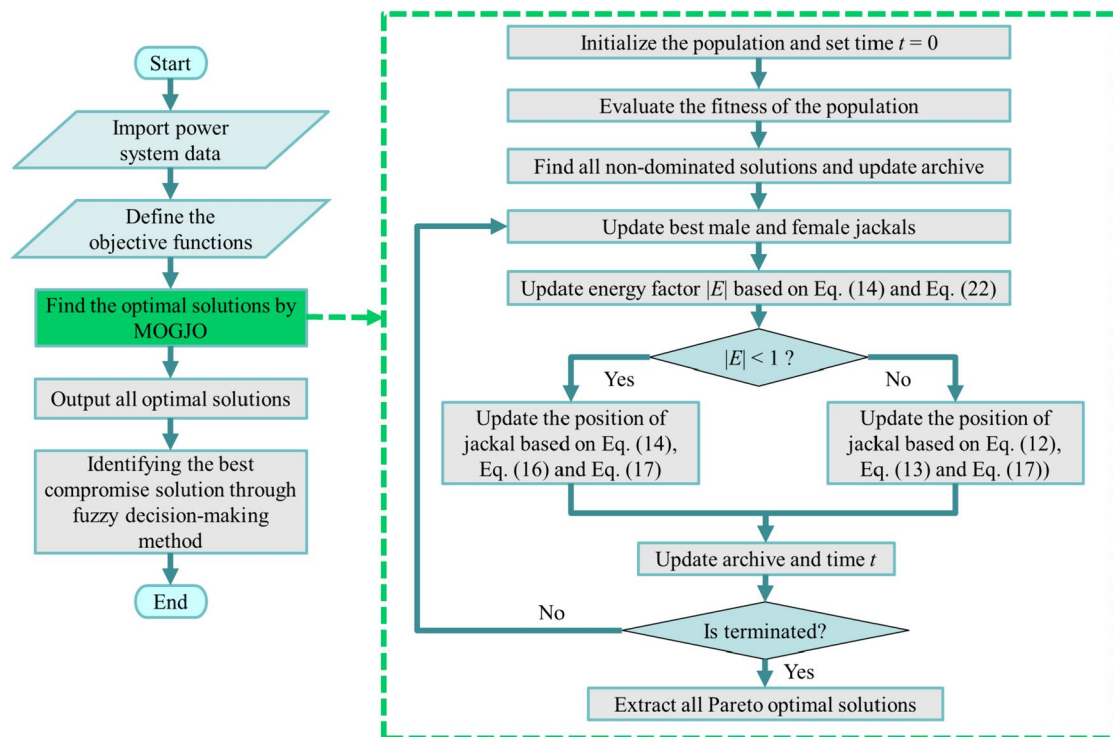


Fig. 2 The flowchart for solving DEED problem using MOGJO

3.2.4 Complexity Analysis

The time complexity of the MOGJO mainly depended on its optimization process and the cost of the archive operation. In the population initialization phase, it takes $O(N * M)$. It takes $O(N * T * fc)$ to evaluate the fitness of individuals, where fc denotes the cost of fitness evaluation. The updating of the archive takes $O(I^2 * T * M)$ in total, where I denotes the number of solutions in the archive. It takes $O(T * N)$ to select the best jackal pair. The position update of jackals takes $O(N * D * T)$ in total. In summary, the time complexity of MOGJO can be expressed as $O(T * (N * fc + I^2 * M + N * D + N))$.

The space complexity of the MOGJO mainly depends on the initialization procedure, so the space complexity of the MOGJO is $O(N * M)$.

4 Validation of MOGJO with Benchmark Test Functions

Before tackling the DEED problem, we validate the optimization performance of MOGJO by means of twelve benchmark test functions. Table 1 presents the fundamental information of the benchmark test functions, selected from the CEC2020 multi-objective test suite [25]. To better measure the optimization performance of the algorithm, we adopted two performance indicators to quantitatively analyze the Pareto optimal solutions,

namely, Pareto Sets Proximity (PSP) [26] and Inverted Generational Distance in decision space (IGDX) [27]. The former mainly measures the coverage of the Pareto solutions obtained by the algorithm on the real Pareto solutions, and a higher PSP value means a better quality of the obtained Pareto solutions. The latter is used to measure the distance between the Pareto solutions obtained by the algorithm and the true Pareto solutions, and a small distance indicates that the solution is close to the global optimum.

Table 1 Details of the benchmark functions

Function	Pareto front geometry	Pareto set geometry	Number of objectives
MMF1	Convex	Nonlinear	2
MMF2	Convex	Nonlinear	2
MMF4	Concave	Nonlinear	2
MMF5	Convex	Nonlinear	2
MMF7	Convex	Nonlinear	2
MMF8	Concave	Nonlinear	2
MMF10	Convex	Linear	2
MMF11	Convex	Linear	2
MMF12	Convex	Linear	2
MMF13	Convex	Nonlinear	2
MMF14	Concave	Linear	3
MMF15	Concave	Linear	3

The experimental parameters of MOGJO were set as follows: the maximum number of function evaluations was 30,000, the number of iterations was about 300, and the number of jackal populations was 100. The experiments were conducted on a computer with processor Intel (R) Core (TM) i5-5200U CPU @ 2.20 GHz, Windows 10 operating system, and RAM-12 GB. The algorithm performs 20 unbiased trials on each function and the mean and standard deviation of the obtained performance indicators are recorded.

The results of the PSP obtained by MOGJO and other algorithms such as Multi-Objective Slime Mould Algorithm (MOSMA) [28], Non-Dominated Ranking Genetic Algorithm II (NSGA-II) [28], Strength Pareto Evolutionary Algorithm 2 (SPEA 2) [28], Multi-Objective Particle Swarm Algorithm (MOPSO) [28], Multi-Objective Grey Wolf Optimizer (MOGWO) [28], Multi-Objective Salp Swarm Algorithm (MSSA) [28] and Multi-Objective Whale Optimization Algorithm (MOWOA) [28] in twelve functions are presented in

Table 2. From Table 2, the mean values of PSP obtained by MOGJO on nine functions outperform MOSMA, NSGA-II, SPEA 2, MOPSO, MOGWO, MSSA and MOWOA, which indicates that the Pareto solutions obtained by MOGJO covered on the real Pareto solutions well. The results of the IGD_X indicator are presented in Table 3. From Table 3, it can be observed that the values of IGD_X obtained by MOGJO are slightly inferior to SPEA 2 in MMF5 and MMF7, but MOGJO exhibits excellent optimization performance in the remaining 10 functions. It indicates that the Pareto solutions obtained by MOGJO have great convergence and distribution. Figure 3 depicts the Pareto optimal solutions and Pareto optimal fronts obtained by MOGJO in MMF1, MMF13 and MMF14. It can be observed from Fig. 3 that MOGJO can provide well-distributed Pareto optimal solutions and Pareto optimal front in both decision space and objective space, which fully demonstrates the superior performance of MOGJO in effectively handling multi-objective optimization problems.

Table 2 Comparison of mean and standard deviation values of PSP (mean \pm std.)

Functions	MOSMA [28]	NSGA-II [28]	SPEA 2 [28]	MOPSO [28]	MOGWO [28]	MSSA [28]	MOWOA [28]	MOGJO
MMF1	6.96 \pm 0.78	5.78 \pm 0.86	624 \pm 1.93	1.86 \pm 0.68	0.60 \pm 0.46	1.78 \pm 0.60	3.26 \pm 0.47	11.12 \pm 1.45
MMF2	22.02 \pm 7.33	6.55 \pm 3.39	14.15 \pm 15.77	15.26 \pm 19.35	0.82 \pm 1.00	6.80 \pm 322	3.68 \pm 0.82	37.09 \pm 7.25
MMF4	3.04 \pm 1.09	3.09 \pm 0.52	3.13 \pm 0.82	0.98 \pm 0.22	4.49 \pm 3.63	0.89 \pm 0.28	2.62 \pm 1.16	9.29 \pm 3.79
MMF5	4.42 \pm 0.61	3.45 \pm 0.47	4.63 \pm 0.83	0.60 \pm 0.20	0.00 \pm 0.00	0.57 \pm 0.27	1.87 \pm 0.78	4.82 \pm 2.05
MMF7	6.39 \pm 2.87	7.54 \pm 1.28	9.37 \pm 2.22	0.17 \pm 0.08	0.86 \pm 0.23	0.66 \pm 0.16	4.36 \pm 0.76	4.58 \pm 2.68
MMF8	0.82 \pm 0.38	0.33 \pm 0.18	0.14 \pm 0.07	0.40 \pm 1.01	0.06 \pm 0.02	0.18 \pm 0.10	0.14 \pm 0.12	2.02 \pm 1.26
MMF10	157.8 \pm 16.3	47.4 \pm 56.0	91.6 \pm 18.9	73.7 \pm 18.8	47.6 \pm 35.7	23.2 \pm 23.9	109.9 \pm 9.7	137.13 \pm 16.01
MMF11	123.1 \pm 7.89	93.28 \pm 11.2	84.34 \pm 12.0	30.31 \pm 10.4	75.43 \pm 11.2	17.84 \pm 18.0	109.2 \pm 13.4	125.9 \pm 5.85
MMF12	399.1 \pm 48.9	303.7 \pm 36.8	224.3 \pm 35.0	167.6 \pm 49.2	135.8 \pm 37.2	42.17 \pm 40.3	238.4 \pm 36.4	315.45 \pm 31.5
MMF13	9.08 \pm 3.39	5.55 \pm 1.26	4.68 \pm 2.98	0.77 \pm 0.37	0.96 \pm 0.30	1.85 \pm 1.30	1.86 \pm 1.44	21.59 \pm 2.85
MMF14	8.61 \pm 0.87	5.08 \pm 0.65	3.18 \pm 0.69	3.61 \pm 0.05	0.46 \pm 0.08	1.26 \pm 0.54	3.57 \pm 0.02	9.07 \pm 2.57
MMF15	13.1 \pm 1.78	9.45 \pm 1.28	6.96 \pm 1.19	16.2 \pm 0.83	0.77 \pm 0.18	1.18 \pm 0.32	14.2 \pm 1.30	16.24 \pm 0.96

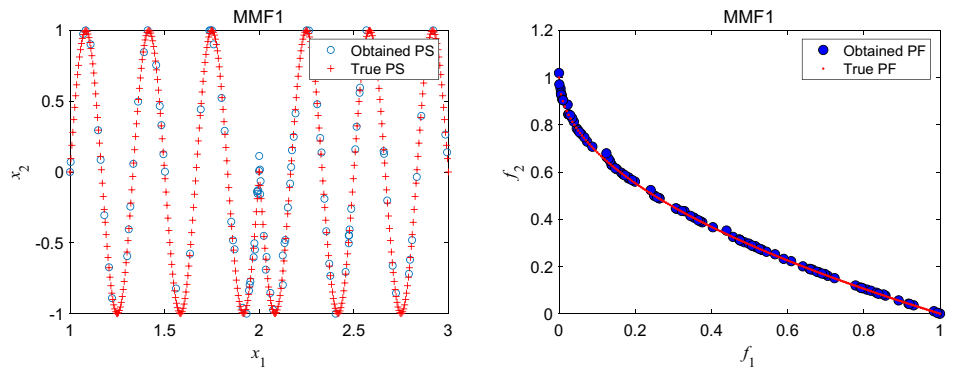
Best results from all comparison methods are bolded

Table 3 Comparison of mean and standard deviation values of IGD_X (mean \pm std.)

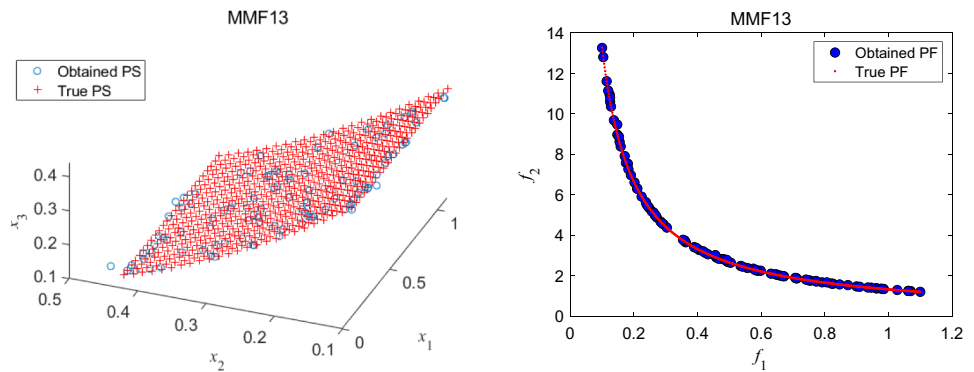
Functions	MOSMA [28]	NSGA-II [28]	SPEA 2 [28]	MOPSO [28]	MOGWO [28]	MSSA [28]	MOWOA [28]	MOGJO
MMF1	0.14 \pm 0.01	0.17 \pm 0.02	0.16 \pm 0.04	0.41 \pm 0.08	0.91 \pm 0.23	0.46 \pm 0.09	0.30 \pm 0.04	0.09 \pm 0.01
MMF2	0.05 \pm 0.02	0.17 \pm 0.09	0.19 \pm 0.17	0.17 \pm 0.11	0.73 \pm 0.32	0.17 \pm 0.13	0.28 \pm 0.05	0.03 \pm 0.01
MMF4	0.33 \pm 0.07	0.31 \pm 0.05	0.26 \pm 0.05	0.58 \pm 0.08	0.30 \pm 0.21	0.57 \pm 0.09	0.40 \pm 0.12	0.12 \pm 0.04
MMF5	0.22 \pm 0.02	0.28 \pm 0.03	0.21 \pm 0.04	0.83 \pm 0.13	2.10 \pm 0.02	0.84 \pm 0.13	0.50 \pm 0.13	0.25 \pm 0.12
MMF7	0.15 \pm 0.04	0.12 \pm 0.02	0.10 \pm 0.03	0.37 \pm 0.01	0.71 \pm 0.26	0.37 \pm 0.03	0.18 \pm 0.02	0.12 \pm 0.04
MMF8	1.28 \pm 0.41	2.00 \pm 0.47	2.55 \pm 0.63	3.89 \pm 1.75	3.09 \pm 0.19	2.02 \pm 0.57	2.75 \pm 0.64	0.67 \pm 0.52
MMF10	0.01 \pm 0.00	0.21 \pm 0.19	0.01 \pm 0.00	0.01 \pm 0.00	0.06 \pm 0.06	0.14 \pm 0.18	0.01 \pm 0.00	0.01 \pm 0.00
MMF11	0.01 \pm 0.00	0.01 \pm 0.00	0.01 \pm 0.00	0.03 \pm 0.01	0.01 \pm 0.00	0.10 \pm 0.10	0.01 \pm 0.00	0.01 \pm 0.00
MMF12	0.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	0.01 \pm 0.00	0.01 \pm 0.00	0.05 \pm 0.04	0.00 \pm 0.00	0.00 \pm 0.00
MMF13	0.11 \pm 0.04	0.13 \pm 0.02	0.16 \pm 0.04	0.19 \pm 0.02	0.21 \pm 0.02	0.18 \pm 0.05	0.16 \pm 0.04	0.05 \pm 0.01
MMF14	0.12 \pm 0.01	0.20 \pm 0.03	0.30 \pm 0.04	0.28 \pm 0.00	0.51 \pm 0.01	0.44 \pm 0.07	0.28 \pm 0.00	0.12 \pm 0.05
MMF15	0.08 \pm 0.01	0.11 \pm 0.02	0.13 \pm 0.02	0.06 \pm 0.00	0.36 \pm 0.02	0.35 \pm 0.02	0.07 \pm 0.01	0.06 \pm 0.00

Best results from all comparison methods are bolded

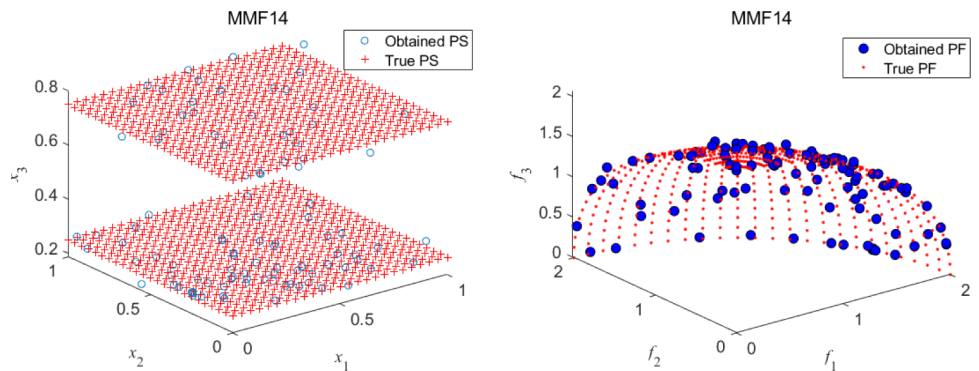
Fig. 3 Pareto optimal solutions and Pareto optimal fronts obtained by MOGJO



(a) Pareto optimal solutions and Pareto optimal front obtained by MOGJO in MMF1



(b) Pareto optimal solutions and Pareto optimal front obtained by MOGJO in MMF13



(c) Pareto optimal solutions and Pareto optimal front obtained by MOGJO in MMF14

5 Simulation Experiments of MOGJO to Solve DEED Problem

In the previous section, we have demonstrated the effectiveness of MOGJO by means of benchmark test functions. In this section, MOGJO is applied to solve the DEED optimization problem with two objectives. Four cases are employed to further investigate the optimization performance of MOGJO. The DEED problems are studied on 5-unit, 10-unit, and IEEE 30 bus 6-generator, and 30-unit systems for these four cases. The population size is set to 100, the number of iterations is 100, and the maximum size of the

archive is 100. The MOGJO is compared with the state-of-the-art DEED solutions in each case. The best compromise solution is selected by the Fuzzy decision-making approach.

In order to better evaluate the performance of the algorithm in a dynamic environment, this paper adopts the Mean Hypervolume (MHV) metric [29] to evaluate the quality of the solutions obtained by the algorithm during the time period. The MHV metric is calculated as follows: first, compute the Hypervolume (HV) [30] metric of the Pareto optimal solutions obtained by the algorithm at each time, and then calculate the mean HV metric of the solutions obtained by the algorithm for all the moments. The larger value of the

MHV metric indicates the better convergence and diversity of the solutions set.

5.1 Case 1: 5-unit Test System

The fuel cost and emission coefficients of the 5-units test system are available from Ref. [11]. The power load demands of the system during 24 h are [410, 435, 475, 530, 558, 608, 626, 654, 690, 704, 720, 740, 704, 690, 654, 580, 558, 608, 654, 704, 680, 605, 527, 463] MW. One hour represents a

Table 4 Best compromise solution for 5-units system obtained by MOGJO and other algorithms

Algorithm	Fuel cost (\$)	Emission (lb)
MOGJO	47,117	18,592
MOMVO [18]	47,659	18,077
MOPPO [13]	50,389	20,818
PSO [31]	50,893	20,163
EP [32]	48,628	21,154
SA [33]	48,621	21,188
PS [33]	47,911	18,927
MONNDE [11]	49,884	18,647
PPSO [34]	48,369	23,685
I-NSGA-III [15]	48,160	18,116

dispatch period. The transmission loss and valve point effect are considered in this system. In this system, the up and down ramp limits are set to 30 MW/h for P_1 and P_2 generators, 40 MW/h for P_3 , and 50 MW/h for P_4 and P_5 , respectively. MOGJO is compared with MOMVO [18], Multi-Objective Proximal Policy Optimization (MOPPO) [13], PSO [31], Evolutionary Programming (EP) [32], Simulated Annealing (SA) [33], Pattern Search (PS) [33], MONNDE [11], Phasor PSO (PPSO) [34] and I-NSGA-III [15].

The best compromise fuel costs and emissions obtained by MOGJO and other algorithms are presented in Table 4. The total fuel cost obtained by MOGJO in 24 h is 47,117 \$ and the corresponding total pollutant gas emission is 18,592 lb. Compared with the MOMVO, MOPPO, PSO, EP, SA, PS, MONNDE and PPSO algorithms, MOGJO significantly has the best value in terms of fuel cost. In addition, MOGJO provides less emission than most algorithms. To further compare the superiority of the best compromise dispatching solutions obtained by MOGJO and latest algorithms, a comparison based on the Pareto dominance relation and the Euclidean distance metric is depicted in Fig. 4. In Fig. 4, the black dots represent the dominated solutions and the non-black dots stand for the non-dominated optimal solutions. The Euclidean distance is calculated according to Eq. (20). In the comparison of the Pareto dominance relation between MOGJO and the other nine algorithms, only the solutions provided by MOGJO and MOMVO are

Fig. 4 Comparison of the best compromise solutions in Case 1

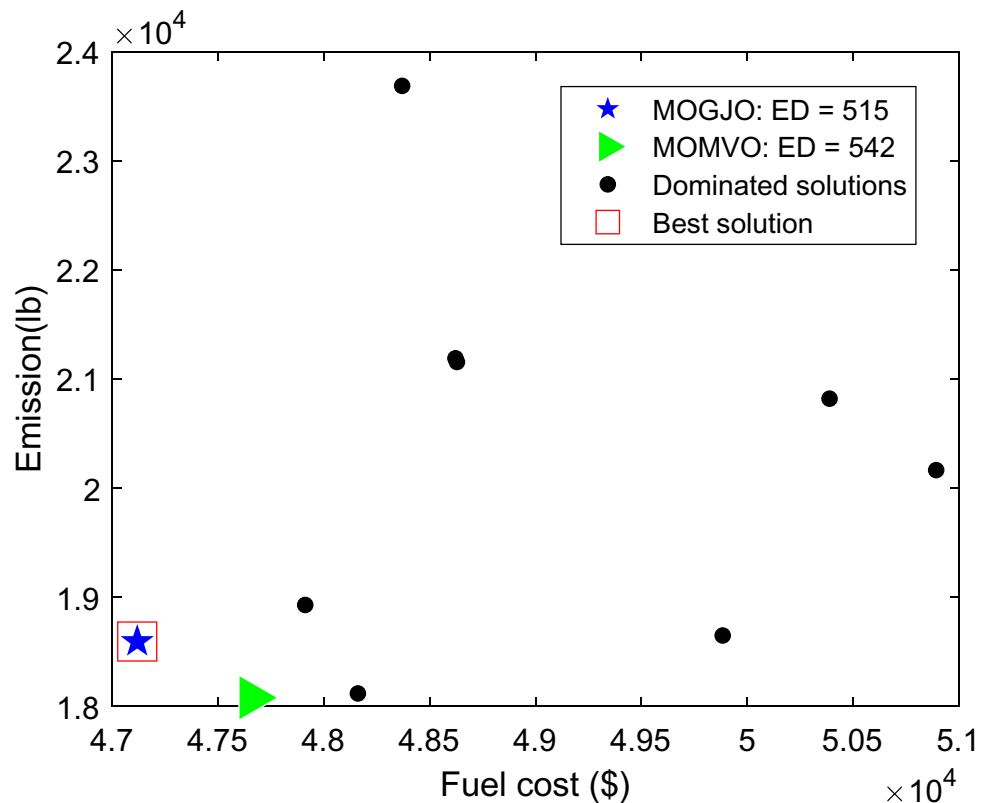


Table 5 Results of performance indicator obtained by MOGJO and MOMVO in Case 1

Algorithms	MOGJO	MOMVO
MHV	0.057902 (3.58E-03)	0.051562 (1.70E-03)
ED	515	542
CPU time (seconds)	83.1405	86.7642

Best results from all comparison methods are bolded

non-dominated, while the solutions provided by the other algorithms are dominated solutions. Moreover, in the comparison of Euclidean distance index, the value of *ED* obtained by MOGJO is 515, while the MOMVO is 542. Therefore, the compromise solution provided by MOGJO has the most desirable fuel cost and pollution emission.

Table 5 records the comparison results of the MHV metrics and CPU running time obtained by MOGJO and the latest DEED method, MOMVO. From the table, MOGJO provides a better MHV value than MOMVO, and at the same time, MOGJO requires less running time than MOMVO. In the comparison of ED, MOGJO has the closest distance to the ideal point, which also indicates that its convergence is better than other algorithms. Thus, it is evident that MOGJO outperforms the state-of-the-art algorithms for the DEED problem with five generators.

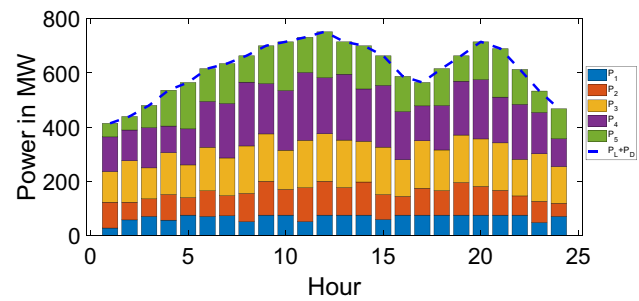


Fig. 5 Compromise dispatching schedule obtained by MOGJO for 5-unit system

Table 6 presents the optimal dispatching scheme and transmission line losses for each generator provided by MOGJO. The scheme includes the dispatching assignments for five generators in different periods. It can be observed from Table 6 that the scheduling of each generator can be kept within the ramp limits and satisfy the power balance constraint and output power constraint. The optimal dispatching solution suggested by MOGJO is depicted in Fig. 5, where the blue dashed line denotes the sum of the power demand and line losses of the system over 24 periods. It can be seen from Fig. 5 that the optimal dispatching solution recommended by MOGJO can

Table 6 The compromise solution obtained by the MOGJO for the 5-units system

Hour	P_1	P_2	P_3	P_4	P_5	P_L
1	27.35	95.33	112.76	128.11	50.00	3.62
2	57.35	65.33	152.76	113.45	50.00	3.97
3	70.40	64.61	114.19	148.22	82.17	4.68
4	56.49	94.60	154.19	98.22	132.17	5.80
5	75.00	64.60	120.49	133.04	171.17	6.43
6	70.24	94.60	160.49	168.97	121.17	7.65
7	72.47	74.21	139.02	200.86	147.40	8.15
8	50.85	104.21	175.00	235.41	97.40	9.14
9	75.00	124.96	174.48	185.41	139.79	9.93
10	75.00	94.96	144.58	219.84	179.69	10.37
11	51.08	124.96	175.00	249.99	129.69	11.07
12	75.00	125.00	175.00	206.67	169.43	11.43
13	75.00	101.75	175.00	242.96	119.43	10.47
14	75.00	122.10	150.25	192.96	159.34	9.96
15	58.71	92.10	175.00	227.61	109.34	9.02
16	75.00	69.14	135.00	177.61	130.04	6.96
17	75.00	99.14	175.00	128.80	86.40	6.49
18	75.00	90.30	149.93	163.83	136.40	7.62
19	75.00	120.30	175.00	198.75	93.72	9.03
20	75.00	106.01	175.00	218.82	139.23	10.35
21	75.00	91.45	175.00	168.82	179.00	9.52
22	75.00	70.81	135.00	202.64	129.00	7.64
23	47.68	78.35	175.00	152.64	79.00	5.80
24	70.41	48.35	135.00	102.64	110.90	4.38

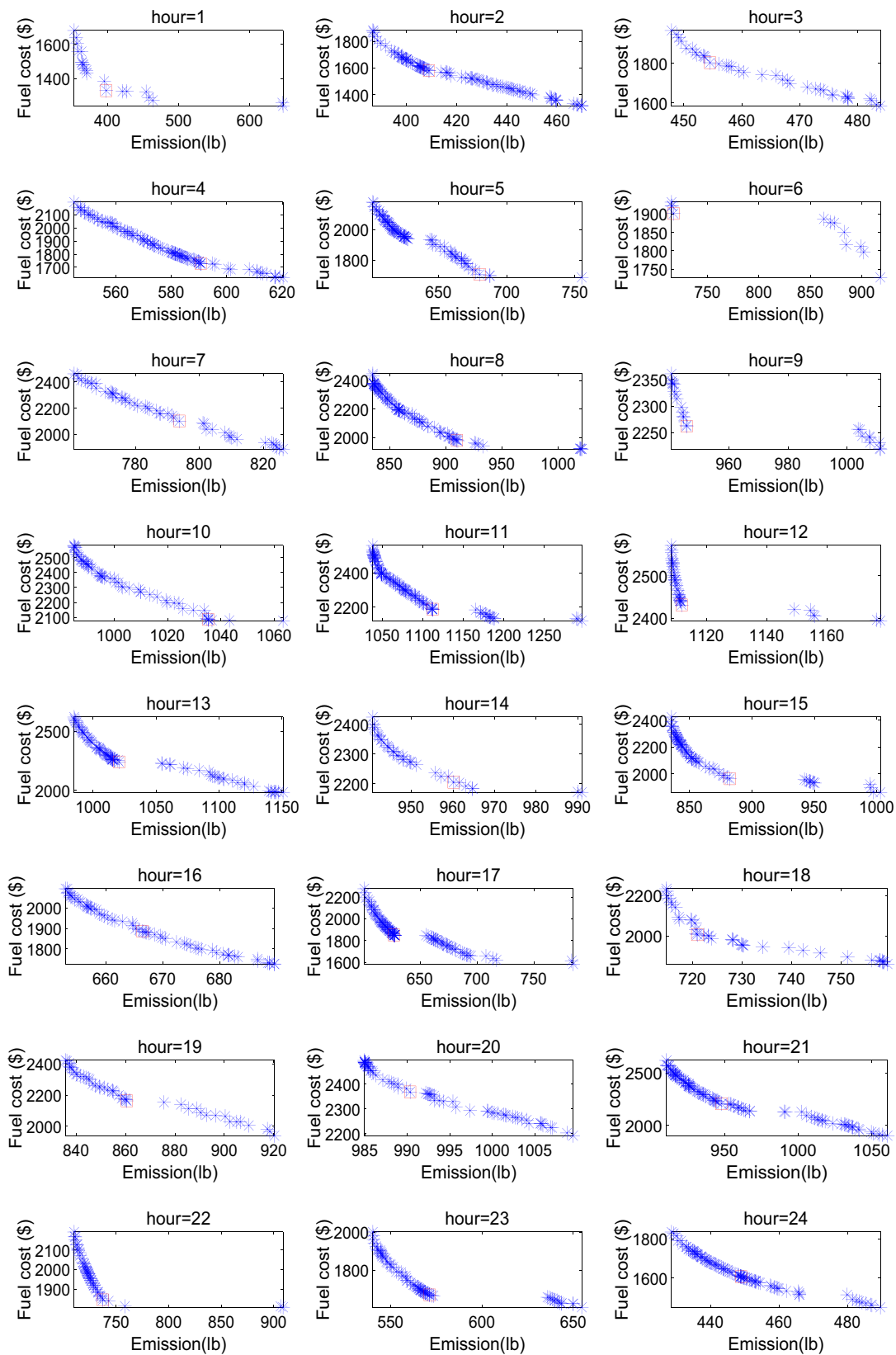


Fig. 6 Pareto optimal front obtained by MOGJO in 24 h for 5-unit system

satisfy the power balance constraint in each dispatching period. The Pareto optimal fronts obtained by MOGJO for each hour are depicted in Fig. 6, where the blue asterisks represent the Pareto optimal solutions and the red boxes denote the optimal compromise solutions. It can be easily observed from Fig. 6 that the Pareto optimal fronts obtained by MOGJO in most periods are well distributed, which is contributed by the leader selection mechanism proposed in this paper. This mechanism enables the selection of jackal pair that can most improve the current population, which has a significant contribution to accelerate the convergence of the algorithm and improve the quality of the population.

5.2 Case 2: 10-unit Test System

In this case, the test system contains 10 generators. The data of fuel, pollutant emission coefficients and B-loss coefficients in the system are obtained from [18]. The power demands of the system in 24 h are [1036, 1110, 1258, 1406, 1480, 1628, 1702, 1776, 1924, 2022, 2106, 2150, 2072, 1924, 1776, 1554, 1480, 1628, 1776, 1972, 1924, 1628, 1332, 1184] MW. In this system, the up and down ramp limits are set to 80 MW/h for generators P_1 - P_3 , 50 MW/h for generators P_4 - P_6 , and 30 MW/h for generators P_7 - P_{10} . MOGJO is compared with the MOMVO [18], MOPPO [13], SPSO [9], PSOAWL [9], NSGA-II [35], Improved Bacterial Foraging Algorithm (IBFA) [36], Real-Coded Genetic Algorithm (RCGA) [35], Modified Adaptive Multi-Objective Differential Evolution (MAMODE) [37], Chemical Reaction

Optimization (CRO) [17], Hybrid Differential Evolution-Based CRO (HCRO) [17], Multi-Objective Differential Evolution (MODE) [38], Multi-Objective Hybrid Differential Evolution With SA (MOHDE-SAT) [38], New Enhanced Harmony Search (NEHS) [39], MONNDE [11], PPSO [34] and PSO with Convergence Speed Controller (PSO-CSC) [40] algorithms.

The optimal fuel costs and pollutant emissions obtained by MOGJO and other algorithms are presented in Table 7. From Table 7, the optimal fuel cost provided by MOGJO is 2.4929×10^6 \$, and the corresponding optimal emission is 3.0020×10^5 lb. MOGJO provides better fuel cost and pollutant gas emissions than 7 of the 17 existing methods for both objectives. In terms of fuel cost, MOGJO also outperforms MOMVO, PSOAWL, IBFA, HCRO, MODE, MOHDE-SAT, NEHS and MONNDE. MOGJO also provides competitive results in terms of pollutant gas emission. A visual comparison diagram of the results presented in Table 7 is illustrated in Fig. 7. In the Pareto domination comparison of the results provided by MOGJO and the other algorithms, except for the solutions provided by MOGJO, MOMVO, PSOAWL and NEHS, which are non-dominated, the others are all dominated solutions. In terms of the Euclidean distance metric, the value of *ED* obtained by MOGJO is 5655, while

Table 7 Best compromise solutions for 10-unit system obtained by MOGJO and other algorithms

Algorithm	Fuel cost $\times 10^6$ (\$)	Emission $\times 10^5$ (lb)
MOGJO	2.4929	3.0020
MOMVO [18]	2.5038	2.9688
MOPPO [13]	2.6795	3.1482
SPSO [9]	2.6044	3.1075
PSOAWL [9]	2.5463	2.9455
NSGA-II [35]	2.5226	3.0994
IBFA [36]	2.5171	2.9904
RCGA [35]	2.5251	3.1246
MAMODE [37]	2.5141	3.0274
CRO [17]	2.5178	3.0194
HCRO [17]	2.5171	2.9907
MODE [38]	2.5276	2.9805
MOHDE-SAT [38]	2.5280	2.9776
NEHS [39]	2.5332	2.9512
MONNDE [11]	2.5600	2.9782
PPSO [34]	2.5358	3.0375
PSO-CSC [40]	2.5247	3.0524
I-NSGA-III [15]	2.5108	3.0562

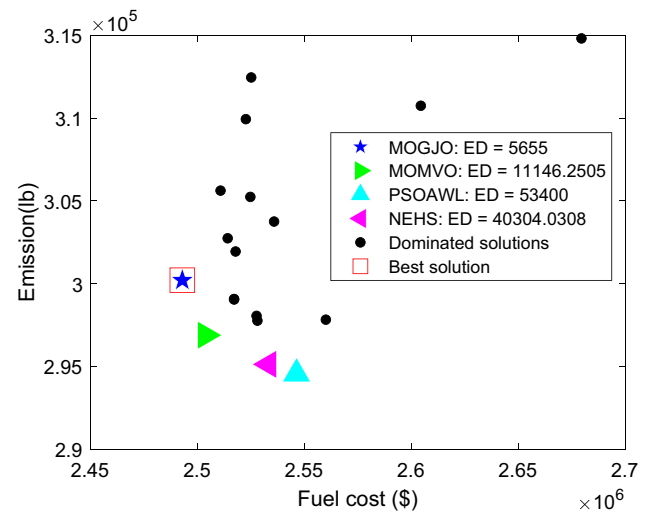


Fig. 7 Comparison of the best compromise solutions in Case 2

Table 8 Results of performance indicator obtained by MOGJO and MOMVO in Case 2

Algorithms	MOGJO	MOMVO
MHV	0.024146 (1.90E-04)	0.02331 (4.99E-04)
ED	5655	11,146.2505
CPU time (seconds)	57.4851	71.1712

Best results from all comparison methods are bolded

MOMVO, PSOAWL and NEHS are 11,146.2505, 53,400 and 40,304.0308, respectively. Therefore, the performance of MOGJO in 10-unit system outperforms the state-of-the-art DEED methods.

Table 8 shows the results of MHV metrics and CPU time obtained by MOGJO in Case 2. It can be easily observed from the table that the MHV metrics of MOGJO are better than the latest DEED method, MOMVO, which proves that MOGJO can effectively balance the convergence and diversity of the Pareto optimal solution set, thus providing the best solution to the decision-maker. The ED value provided by MOGJO is significantly better than that of MOMVO, which also proves that MOGJO has better convergence. In addition, in the comparison of CPU time, the running cost of MOGJO is significantly less than MOMVO, which also proves that MOGJO is an efficient DEED solver.

The optimal power scheduling provided by MOGJO in Case 2 is presented in Table 9. From Table 9, the output power of each generator in the adjacent periods can be well controlled within the ramp limits, and the generators satisfy the power output constraint and power balance constraint in each period. It can be also verified from the load allocation scheme of each generator in each period illustrated in Fig. 8. From Fig. 8, the real power and

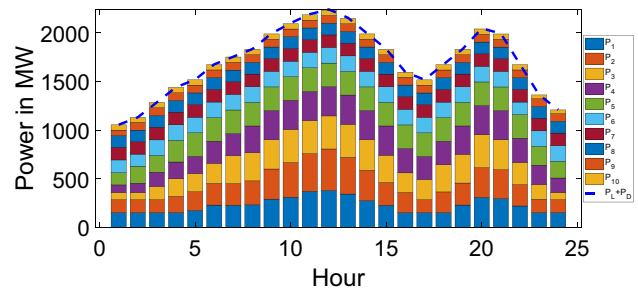


Fig. 8 Compromise dispatching schedule obtained by MOGJO for 10-unit system

$P_D + P_L$ can be well matched. The Pareto optimal fronts obtained by MOGJO in 24 time periods are depicted in Fig. 9. In terms of the distribution of the solutions set, the Pareto optimal solutions denoted by the blue asterisks have better coverage. Meanwhile, the optimal trade-off solutions denoted by the red boxes can better handle the conflict between the two objectives of fuel cost and pollution emission to the optimal trade-off degree. It also fully demonstrates that the performance of MOGJO is more prominent than other algorithms in locating different Pareto-optimal solutions in the search space of the DEED problem.

Table 9 The compromise solution obtained by the MOGJO for the 10-unit system

Hour	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_L
1	150.00	135.00	73.00	79.15	124.82	131.90	130.00	120.00	56.28	55.00	19.75
2	150.00	135.00	73.00	96.64	174.82	124.51	126.21	120.00	80.00	51.51	22.57
3	150.00	135.00	136.07	146.64	172.57	160.00	130.00	120.00	80.00	55.00	28.43
4	150.00	168.95	182.13	171.49	222.57	160.00	130.00	120.00	80.00	55.00	35.67
5	173.41	193.78	186.26	176.58	243.00	160.00	130.00	120.00	80.00	55.00	39.96
6	229.61	222.82	207.35	226.58	243.00	160.00	130.00	120.00	80.00	55.00	49.16
7	228.05	222.74	287.35	226.24	243.00	160.00	130.00	120.00	80.00	55.00	53.73
8	231.90	246.72	289.83	274.90	243.00	160.00	130.00	120.00	80.00	55.00	58.77
9	290.97	309.58	301.26	300.00	243.00	160.00	130.00	120.00	80.00	55.00	70.60
10	310.27	356.62	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	78.96
11	370.69	387.43	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	87.15
12	379.24	427.33	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	91.68
13	343.17	377.90	340.00	300.00	243.00	160.00	130.00	120.00	80.00	55.00	83.75
14	273.35	313.77	314.76	300.00	243.00	160.00	130.00	120.00	80.00	55.00	70.44
15	227.50	233.77	281.94	300.00	243.00	160.00	130.00	120.00	80.00	55.00	58.59
16	150.00	209.06	201.94	250.00	243.00	160.00	130.00	120.00	80.00	51.64	43.90
17	150.00	135.00	207.00	237.39	243.00	160.00	130.00	120.00	80.00	55.00	39.26
18	150.00	215.00	279.14	241.57	243.00	160.00	130.00	120.00	80.00	55.00	48.32
19	230.00	226.32	295.40	291.57	243.00	160.00	130.00	120.00	80.00	55.00	58.56
20	308.73	306.32	337.72	300.00	243.00	160.00	130.00	120.00	80.00	55.00	74.40
21	294.04	299.37	307.90	300.00	243.00	160.00	130.00	120.00	80.00	55.00	70.49
22	220.29	219.37	227.90	250.00	243.00	128.61	130.00	120.00	80.00	55.00	49.20
23	150.00	139.37	150.56	200.00	193.00	160.00	130.00	120.00	80.00	39.54	31.76
24	150.00	135.00	73.00	150.00	171.89	160.00	130.00	120.00	80.00	38.44	25.34

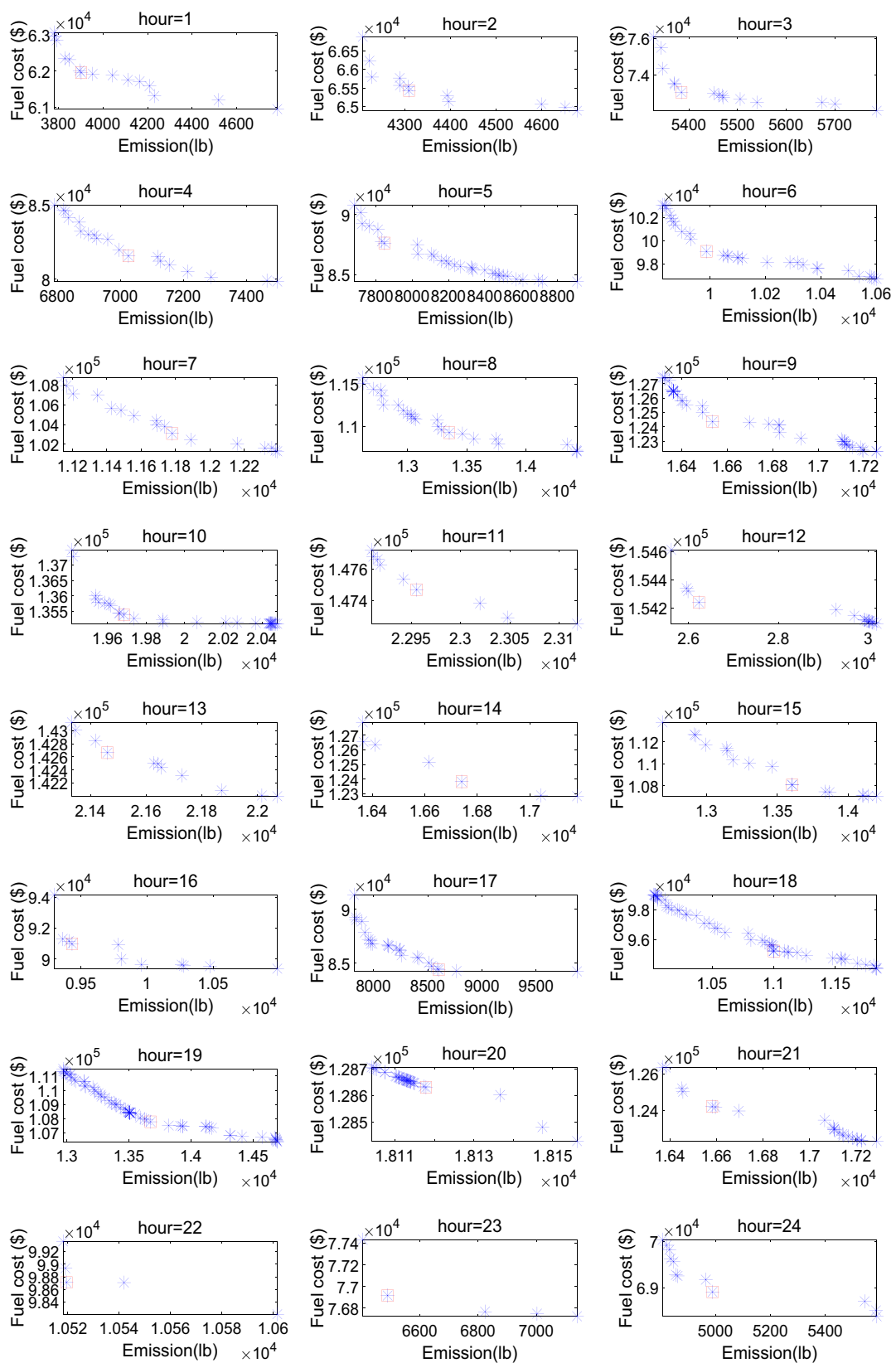


Fig. 9 Pareto optimal fronts obtained by MOGJO in 24 h for 10-unit system

Table 10 Best compromise solutions obtained by MOGJO and other algorithms in Case 3

Algorithm	Fuel cost (\$)	Emission (lb)
MOGJO	25,699.01	6.17904
MSSA [41]	25,727.57	5.94564
MALO [41]	26,233.87	5.72757
MOGOA [41]	26,337.08	5.74666
MAMODE [37]	25,912.89	5.97955
GSOMP [7]	25,924.46	6.00415
NEHS [39]	26,294.59	5.72785
MHS [39]	26,624.35	6.00505
HS-NPSA [39]	26,796.48	6.20471
DHS [39]	26,552.55	5.94119
HIS [39]	26,426.88	5.85317

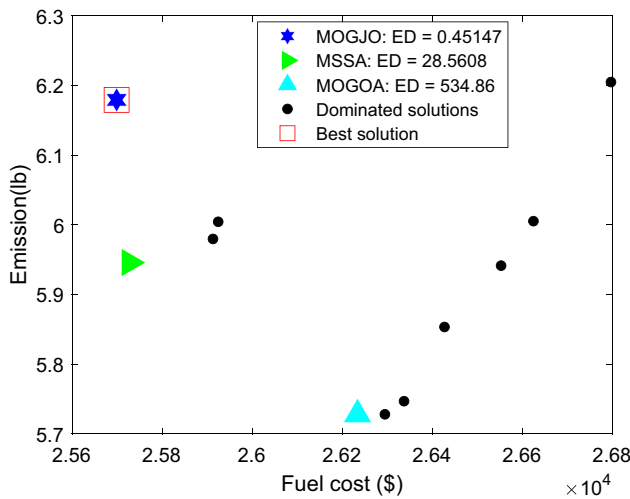


Fig. 10 Comparison of the best compromise solutions in Case 3

5.3 Case 3: IEEE 30 Bus System

In this case, the standard IEEE 30 bus system with six thermal units is used to verify the performance of MOGJO, where the data on fuel, emission coefficients and network loss factors are taken from [7]. The power demands of the system in 24 h are [3.25, 3.90, 3.50, 3.00, 3.35, 4.00, 4.75, 5.05, 5.45, 5.20, 5.50, 5.75, 5.25, 5.15, 4.75, 5.30, 5.15, 5.75, 5.25, 5.25, 4.55, 4.25, 4.25, 4.00]

MW. In this system, the up and down ramp limits of all six generators are 0.5. In Case 3, MOGJO is compared with MSSA [41], Multi-Objective Ant Lion Optimizer (MALO) [41], Multi-Objective Grasshopper Optimization Algorithm (MOGOA) [41], MAMODE [37], Group Search Optimizer with Multiple Producers (GSOMP) [7], NEHS [39], Modified Harmony Search (MHS) [39], Harmony Search with A Novel Parameter Setting Approach (HS-NPSA) [39], Differential Harmony Search (DHS) [39] and Improved Harmony Search (HIS) [39] algorithms.

The optimal fuel cost and emission obtained by MOGJO are presented in Table 10. From Table 10, the optimal fuel cost provided by MOGJO is 25,699.01 \$ and the corresponding emission is 6.17904 lb. The fuel cost provided by MOGJO is the best among all algorithms. As can be seen from the Pareto dominance relationship of the best compromise solution shown in Fig. 10, only MOGJO, MSSA and MOGOA are non-dominated, while the others are all dominated. In addition, in the comparison of the Euclidean distance metric, the value of ED obtained by MOGJO is 0.45147, which is significantly better than the 28.5608 and 534.86 obtained by MSSA and MOGOA. Thus, it can be demonstrated that MOGJO is able to reduce the fuel cost and pollution emission to the vicinity of the ideal solution that minimizes the two objectives, which demonstrates a superior performance than the latest published algorithms.

Table 11 shows the performance metrics results of MOGJO with MSSA, MOGOA and MALO algorithms in Case 3. From the table, MOGJO provides the best MHV and ED values, which proves that MOGJO outperforms the latest DEED methods. Although in the comparison of CPU time, MOGJO is not the fastest run, it is still competitive. The load scheduling of the six generators provided by MOGJO in different periods is presented in Table 12. All generators satisfy the up and down ramp constraint, the power balance constraint, and the output power constraint. For more visual comparison of the degree of constraint satisfaction, the real power and the sum of the demand power and transmission line losses are depicted in Fig. 11. The optimal dispatching solution provided by MOGJO satisfies the power balance constraint. Figure 12 shows the Pareto optimal fronts obtained by MOGJO for each period. From Fig. 12, MOGJO can provide multiple high-quality Pareto optimal solutions in different periods. In the comparison of optimal fuel cost and

Table 11 Results of performance indicator obtained by MOGJO and other methods in Case 3

Algorithms	MOGJO	MSSA	MOGOA	MALO
MHV	0.03565 (1.16E-03)	0.033649(6.4E-05)	0.033927(3.79E-04)	0.030815 (2.39E-04)
ED	0.45147	28.56083	638.0700	534.8600
CPU time (seconds)	73.6263	42.2105	449.8568	64.4731

Best results from all comparison methods are bolded

Table 12 The compromise solution obtained by the MOGJO for the IEEE 30 bus system

Hour	P_1	P_2	P_3	P_4	P_5	P_6	P_L
1	0.30	0.41	0.70	0.73	0.75	0.39	0.03
2	0.36	0.48	0.80	1.06	0.72	0.53	0.05
3	0.35	0.40	0.63	0.91	0.75	0.50	0.04
4	0.28	0.32	0.65	0.72	0.59	0.46	0.03
5	0.36	0.44	0.67	0.82	0.64	0.46	0.04
6	0.27	0.49	0.77	0.97	0.88	0.67	0.05
7	0.40	0.51	1.03	1.10	0.98	0.79	0.07
8	0.56	0.59	1.11	1.14	1.04	0.69	0.08
9	0.61	0.71	1.03	1.28	1.14	0.77	0.10
10	0.59	0.54	1.11	1.23	1.04	0.78	0.09
11	0.43	0.56	1.14	1.43	1.12	0.90	0.09
12	0.54	0.72	1.13	1.50	1.14	0.82	0.11
13	0.47	0.58	1.11	1.42	1.04	0.71	0.08
14	0.37	0.58	1.07	1.38	1.07	0.76	0.08
15	0.50	0.51	1.04	1.10	0.95	0.72	0.07
16	0.51	0.59	1.12	1.46	1.05	0.65	0.09
17	0.52	0.64	1.10	1.32	1.10	0.55	0.08
18	0.49	0.63	1.26	1.50	1.18	0.79	0.10
19	0.55	0.70	1.04	1.10	1.10	0.83	0.09
20	0.46	0.71	1.04	1.41	1.03	0.69	0.09
21	0.36	0.53	1.03	1.12	0.98	0.59	0.06
22	0.41	0.47	0.94	1.08	0.85	0.55	0.05
23	0.42	0.59	0.85	1.05	0.84	0.55	0.06
24	0.36	0.41	0.83	0.95	0.94	0.56	0.05

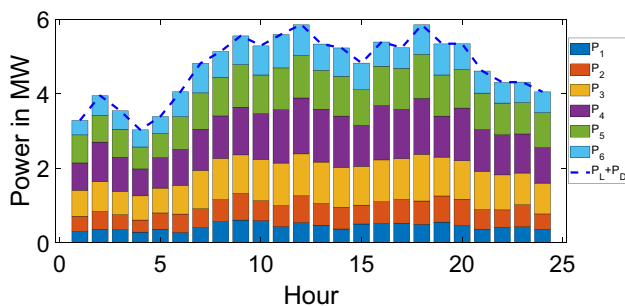


Fig. 11 Compromise dispatching schedule obtained by MOGJO for IEEE 30 bus system

emissions, MOGJO is superior to other algorithms. It also demonstrates that the MOGJO algorithm proposed in this paper can provide highly competitive results in addressing complex dynamic multi-objective optimization problems.

5.4 Case 4: 30-unit System

In Case 4, a large-scale power system containing 30 generators is used to further verify the effectiveness of MOGJO. Transmission losses are not considered in this system. System data such as fuel, emission coefficients and unit ramp

limits are from [17]. The 24-h power demand of the system is [3108, 3330, 3774, 4218, 4440, 4884, 5106, 5328, 5772, 6216, 6438, 6660, 6216, 5772, 5328, 4662, 4440, 4884, 5328, 6216, 5772, 4884, 3996, 3552] MW. In Case 4, MOGJO is compared with MOMVO, MSSA, CRO [17] and HCRO [17] algorithms.

Table 13 shows the best compromise solution provided by MOGJO and other algorithms. It can be observed from the table that the solution provided by MOGJO has the lowest emissions. Figure 13 shows the comparison results of the Pareto dominance relationship and ED indicator of the solutions provided by MOGJO and other algorithms. As can be seen from the figure, compared with other algorithms, the ED value of the solution provided by MOGJO is 55,484.0557, which is closest to the ideal solution. This also proves that MOGJO has good convergence.

Table 14 shows the comparison results of performance indicators and CPU time between MOGJO and other algorithms. As can be seen from the table, although MOGJO does not provide the best MHV value, it is better than other algorithms in ED indicator. It should be noted that when MOGJO uses ED to select the best leader to guide the algorithm search, its selection source is the non-dominated optimal solution set. Therefore, the leader determined by the ED value is always the individual with the best convergence in

Fig. 12 Pareto optimal fronts obtained by MOGJO in 24 h for IEEE 30 bus system

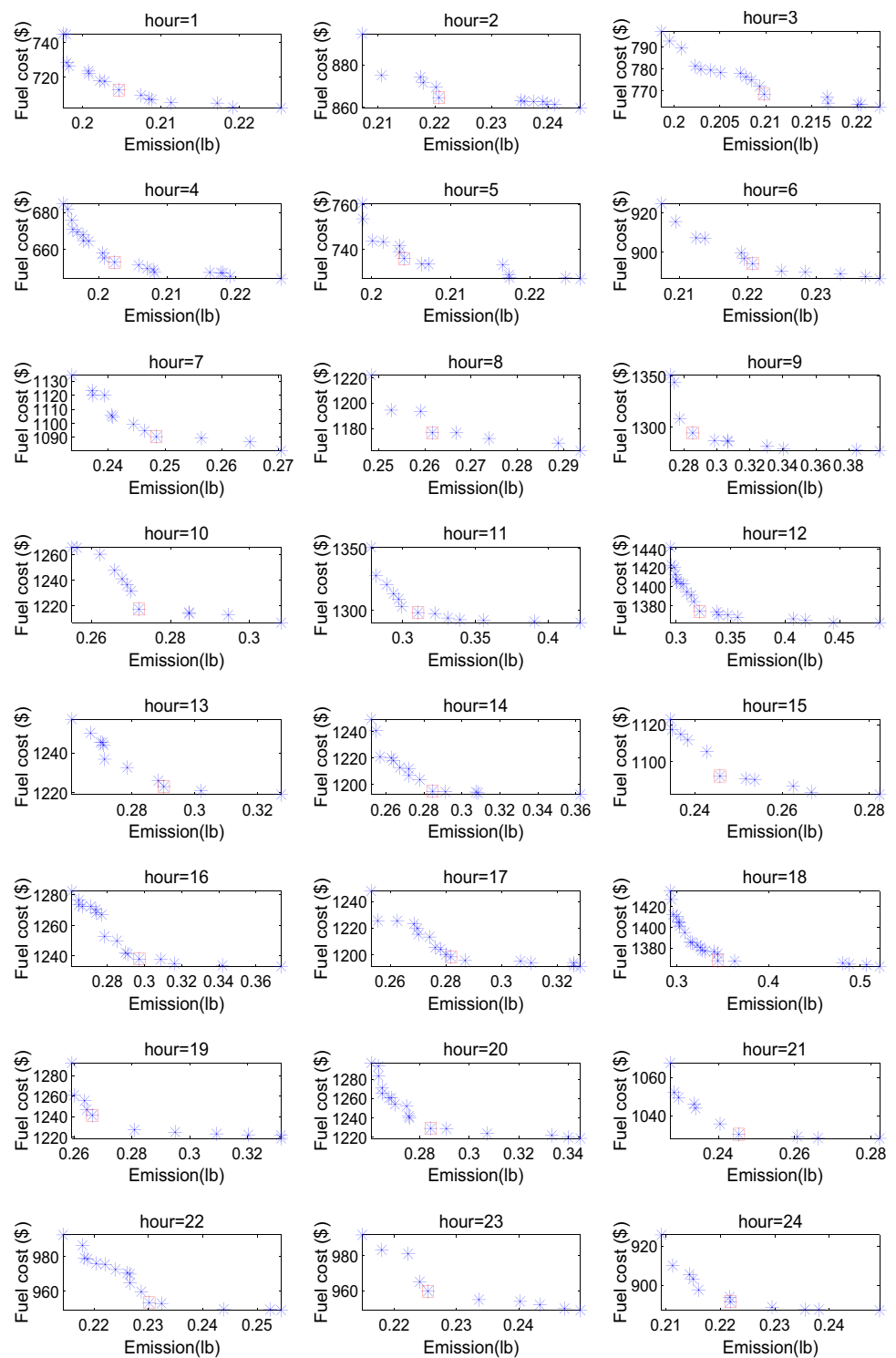


Table 13 Best compromise solutions for 30-unit system obtained by MOGJO and other algorithms

Algorithm	Fuel cost (\$)	Emission (lb)
MOGJO	3,154,426.06	876,701.60
MOMVO	3,149,561.51	931,881.71
MSSA	3,167,232.68	885,827.15
CRO [17]	3,113,266	985,484
HCRO [17]	3,098,942	984,932

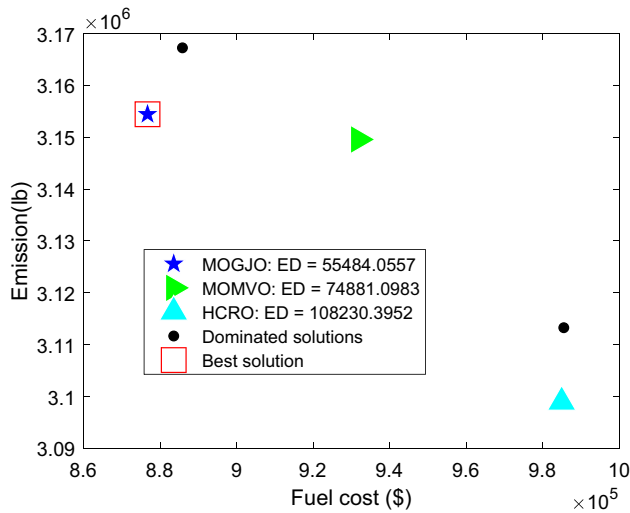


Fig. 13 Comparison of the best compromise solutions in Case 4

the current population, which can speed up the convergence of the algorithm in the subsequent search process.

Table 15 lists the power dispatching schemes for the 30 generators provided by MOGJO. It can be observed from the table that the power provided by MOGJO in each hour is equal to the load demand for the corresponding period. This result can also be verified by the power dispatching and load demand distribution diagram of each generator at

various periods shown in Fig. 14. This further proves that MOGJO can still provide optimal scheduling solutions in larger power systems, showing highly competitive optimization performance. Figure 15 shows the Pareto optimal front obtained by MOGJO in every hour. It can be observed from the figure that the optimal front provided by MOGJO has a good distribution.

6 Conclusion

In this paper, a new multi-objective optimization algorithm, denoted as MOGJO, is developed to solve the dynamic multi-objective economic emission dispatch problems. First, MOGJO is validated for its performance using CEC2020 multi-objective benchmark test functions. The test results show that the Pareto optimal solutions and Pareto optimal fronts obtained by MOGJO have better distribution compared to the state-of-the-art multi-objective algorithms, indicating that MOGJO outperforms other algorithms in optimization performance. Moreover, MOGJO is applied to complex multi-objective DEED problems, where fuel cost and pollution emission are two conflicting optimization objectives. Compared with other latest algorithms, the DEED optimal scheduling scheme provided by MOGJO has the best trade-off between the two objectives of fuel cost and pollution emissions, and satisfies various complex constraints. In the 5-unit, 10-unit, and IEEE 30-bus systems, the fuel costs provided by MOGJO are 47,117 \$, 2.4929×10^6 \$ and 25,699.01 \$ respectively. In the 30-unit system, MOGJO provided the lowest emissions of 876,701.60 lb compared to other algorithms. Besides, the optimal compromise solutions recommended by MOGJO are demonstrated to outperform the state-of-the-art DEED methods for both fuel cost and pollution emission objectives in a comparison of Pareto dominance relationship and Euclidean distance metric with the latest published results. In future work, it is considered to solve the DEED problem in more complex power systems, such as combined heat and power

Table 14 Results of performance indicator obtained by MOGJO and other methods in Case 4

Algorithms	MOGJO	MSSA	MOMVO
MHV	0.024701 (1.94E-03)	0.02569 (2.72E-03)	0.017887 (7.19E-04)
ED	55,484.0557	68,897.6999	74,881.0983
CPU time (seconds)	98.7037	101.2685	93.6805

Best results from all comparison methods are bolded

Table 15 The compromise solution obtained by the MOGJO for the 30-unit system

Hour	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}
1	150.00	213.81	73.00	121.93	230.24	160.00	130.00	86.20	20.00	50.00	150.00	218.44	73.00
2	153.82	135.00	96.21	109.62	217.89	132.85	130.00	91.64	50.00	50.00	150.48	138.44	74.28
3	156.61	215.00	165.54	92.18	172.02	154.47	129.88	120.00	51.29	53.77	203.04	215.78	89.39
4	224.75	135.66	196.70	129.60	219.89	159.80	129.89	119.52	56.60	55.31	280.79	137.14	158.68
5	212.13	214.87	176.97	175.70	214.36	147.56	130.00	111.24	63.54	50.00	225.59	211.30	181.91
6	236.65	282.68	185.27	182.34	229.75	160.00	130.00	120.00	79.70	50.78	296.46	221.45	203.97
7	316.65	310.87	195.47	170.22	233.95	160.00	130.00	117.80	52.78	50.26	267.90	301.45	216.91
8	244.10	308.18	275.47	183.12	221.70	148.10	129.63	116.96	54.50	50.36	315.57	314.65	296.90
9	317.71	338.66	297.63	211.90	221.27	160.00	130.00	100.55	80.00	50.00	302.96	380.15	321.20
10	385.38	392.51	340.00	256.49	231.16	160.00	130.00	118.21	79.35	53.46	382.96	308.65	299.77
11	417.21	312.51	321.05	298.51	240.84	157.53	129.10	120.00	54.53	52.24	379.95	388.65	331.99
12	378.99	392.51	340.00	300.00	242.11	160.00	130.00	120.00	79.55	56.00	455.50	309.49	334.28
13	298.99	312.51	319.45	294.98	231.00	160.00	130.00	118.07	64.57	52.36	382.93	389.49	299.91
14	307.13	304.32	299.79	287.73	215.74	159.55	126.79	120.00	54.42	54.95	302.93	309.49	287.80
15	227.13	308.83	290.71	246.09	232.15	143.81	122.22	113.37	59.41	52.12	289.81	229.49	267.93
16	235.50	307.84	210.71	196.09	214.71	150.98	130.00	101.59	36.19	52.28	234.68	227.27	192.20
17	155.50	227.84	199.55	148.82	184.89	160.00	130.00	120.00	29.52	50.72	296.46	224.20	173.59
18	227.57	295.22	255.46	167.68	220.05	148.05	130.00	120.00	57.28	50.00	223.50	304.20	186.76
19	301.47	304.09	282.66	181.05	221.45	156.76	130.00	119.97	50.19	50.00	303.00	224.99	264.42
20	379.39	384.09	315.02	231.05	243.00	160.00	130.00	120.00	80.00	54.30	380.76	304.99	335.62
21	302.80	304.20	328.10	277.65	242.71	160.00	129.83	119.68	75.16	50.59	302.77	310.52	318.70
22	226.41	224.20	248.10	230.82	222.80	145.64	129.69	110.65	64.17	55.11	226.30	230.79	240.33
23	152.75	146.16	175.30	180.87	177.66	153.32	111.15	119.93	52.46	52.31	230.10	150.81	184.27
24	150.00	218.39	147.10	130.94	136.73	160.00	130.00	89.93	54.22	55.17	151.04	136.28	136.86
Hour	P_{14}	P_{15}	P_{16}	P_{17}	P_{18}	P_{19}	P_{20}	P_{21}	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}
1	60.00	73.00	57.00	130.00	47.00	20.00	50.00	150.00	135.00	73.00	60.00	144.52	160.00
2	110.00	121.96	107.00	130.00	74.51	50.00	56.00	150.00	215.00	132.72	68.96	171.27	160.00
3	71.42	164.06	138.42	130.00	87.35	79.47	51.51	171.83	138.40	162.51	118.47	179.59	150.05
4	120.72	211.87	158.46	129.55	92.71	51.78	55.85	226.29	215.75	180.45	114.97	187.99	159.46
5	139.20	208.09	160.00	130.00	118.13	69.99	51.20	227.89	211.87	175.09	124.21	217.29	160.00
6	183.58	229.44	156.16	128.96	120.00	76.40	52.94	228.59	223.39	187.19	173.31	224.51	158.98
7	182.78	201.03	125.00	130.00	115.82	51.11	50.84	289.09	303.39	201.48	172.86	231.03	144.31
8	188.03	226.64	160.00	124.31	118.50	72.64	50.18	306.66	223.39	281.48	175.02	232.32	152.96
9	225.93	243.00	159.84	130.00	120.00	80.00	53.31	302.70	303.39	297.04	194.86	223.31	150.63
10	268.18	243.00	152.21	130.00	117.29	80.00	50.00	382.70	383.39	292.93	244.86	238.70	137.27
11	300.00	237.55	159.82	129.56	120.00	79.69	52.57	376.69	392.02	330.45	292.75	227.89	158.90
12	300.00	242.25	160.00	130.00	120.00	80.00	50.48	455.65	396.57	339.06	300.00	242.05	159.80
13	300.00	238.59	156.23	130.00	114.13	80.00	56.00	379.09	393.13	335.26	252.18	226.22	145.35
14	275.46	226.59	160.00	129.87	119.25	59.68	53.73	302.86	313.57	298.82	253.61	229.81	159.17
15	225.46	224.25	148.64	130.00	106.40	48.49	51.39	284.40	307.48	275.11	240.12	189.52	160.00
16	175.46	174.25	132.28	128.95	91.41	20.00	50.00	300.12	227.48	195.11	195.26	165.85	160.00
17	125.46	138.57	155.14	130.00	120.00	49.35	54.81	220.12	307.48	199.13	149.71	176.93	157.93
18	175.46	183.73	158.33	130.00	120.00	52.06	50.00	300.12	231.54	211.03	151.10	219.94	135.37
19	185.09	233.73	155.05	129.85	120.00	60.67	50.58	290.40	311.54	278.81	180.53	222.08	156.38
20	233.05	242.29	160.00	130.00	120.00	79.90	51.46	370.40	391.54	299.61	230.53	243.00	160.00
21	194.97	242.80	160.00	129.61	119.84	79.46	55.65	304.95	315.39	287.74	188.30	243.00	160.00
22	169.53	222.07	154.97	129.55	115.68	54.97	52.23	226.62	280.08	211.94	177.05	223.92	154.78
23	120.54	172.07	132.90	130.00	86.75	68.28	50.03	152.53	221.39	132.33	132.48	211.74	142.73
24	110.79	182.70	150.17	130.00	88.23	61.54	50.15	150.00	142.53	73.75	100.47	166.04	144.30

Table 15 (continued)

Hour	P_{27}	P_{28}	P_{29}	P_{30}	P_D
1	126.24	75.63	20.00	50.00	3108
2	128.23	47.95	26.15	50.00	3330
3	130.00	77.85	52.86	51.24	3774
4	126.41	49.28	80.00	52.16	4218
5	127.98	79.28	71.78	52.83	4440
6	130.00	100.89	80.00	50.60	4884
7	130.00	120.00	80.00	52.97	5106
8	130.00	120.00	50.60	56.00	5328
9	130.00	116.95	79.01	50.00	5772
10	128.30	120.00	55.95	53.27	6216
11	127.81	120.00	77.18	51.01	6438
12	129.86	120.00	79.84	56.00	6660
13	128.37	120.00	55.14	52.07	6216
14	128.20	116.85	63.90	50.00	5772
15	130.00	118.84	51.87	52.95	5328
16	112.29	117.83	74.79	50.85	4662
17	127.13	93.68	79.98	53.48	4440
18	129.93	118.47	77.58	53.55	4884
19	128.75	120.00	59.52	54.96	5328
20	130.00	120.00	80.00	56.00	6216
21	128.95	120.00	66.62	52.02	5772
22	129.28	120.00	55.59	50.75	4884
23	127.78	98.69	78.67	50.00	3996
24	123.50	79.04	52.12	50.00	3552

Best results from all comparison methods are bolded

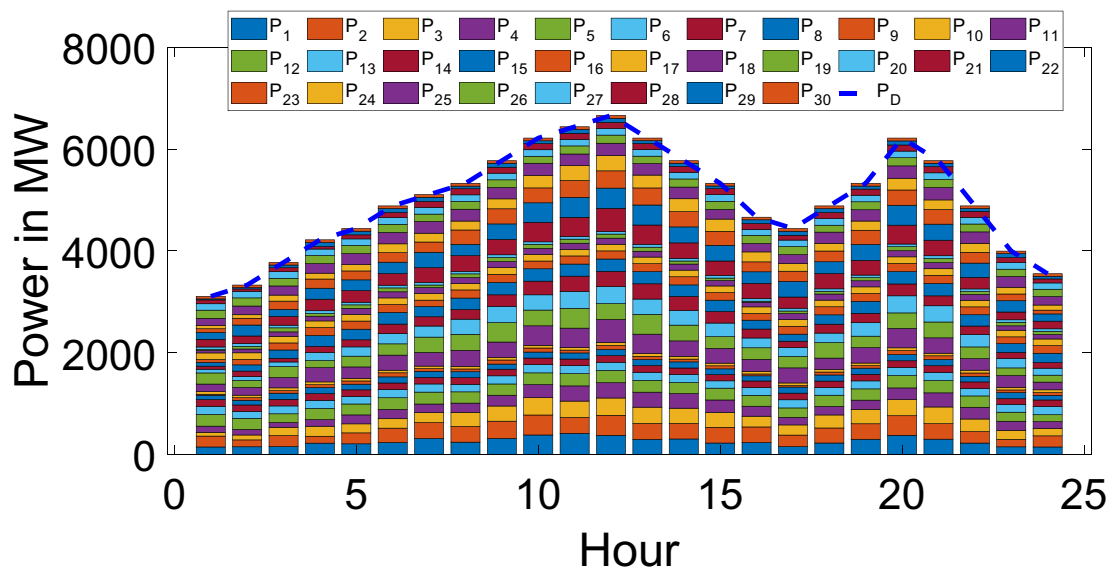


Fig. 14 Compromise dispatching schedule obtained by MOGJO for 30-unit system

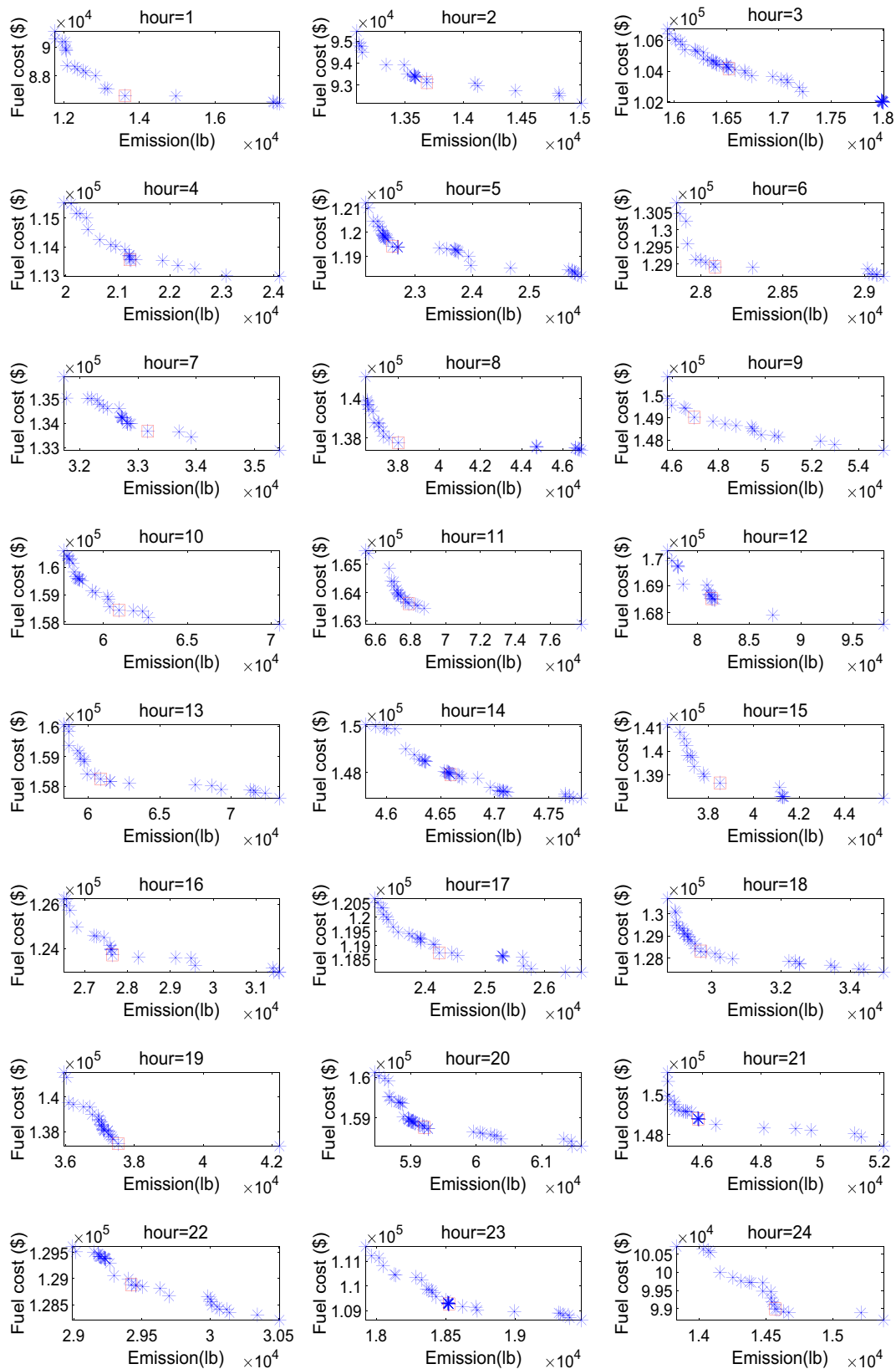


Fig. 15 Pareto optimal fronts obtained by MOGJO in 24 h for 30-unit system

system [42], to further validate the optimization performance of MOGJO. Furthermore, it is an interesting idea to address the DEED problem for power systems incorporating renewable energy sources [43] using MOGJO.

Acknowledgements This work is supported by the National Natural Science Foundation of China under Grant No. 61802328, 61972333, and 61771415.

Data Availability The data involved in this study are all public data, which can be downloaded through public channels.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

- Dong, R., Sun, L., Ma, L., Heidari, A. A., Zhou, X., & Chen, H. (2023). Boosting kernel search optimizer with slime mould foraging behavior for combined economic emission dispatch problems. *Journal of Bionic Engineering*, 20, 2863–2895.
- Lai, W., Zheng, X., Song, Q., Hu, F., Tao, Q., & Chen, H. (2022). Multi-objective membrane search algorithm: A new solution for economic emission dispatch. *Applied Energy*, 326, 119969.
- Sheng, W., Li, R., Yan, T., Tseng, M. L., Lou, J., & Li, L. (2023). A hybrid dynamic economics emissions dispatch model: Distributed renewable power systems based on improved COOT optimization algorithm. *Renewable Energy*, 204, 493–506.
- McLarty, D., Panossian, N., Jabbari, F., & Traverso, A. (2019). Dynamic economic dispatch using complementary quadratic programming. *Energy*, 166, 755–764.
- Somuah, C. B., & Khunaizi, N. (1990). Application of linear programming redispatch technique to dynamic generation allocation. *IEEE Transactions on Power Systems*, 5(1), 20–26.
- Travers, D. L., & Kaye, R. J. (1998). Dynamic dispatch by constructive dynamic programming. *IEEE Transactions on Power Systems*, 13(1), 72–78.
- Guo, C. X., Zhan, J. P., & Wu, Q. H. (2012). Dynamic economic emission dispatch based on group search optimizer with multiple producers. *Electric Power Systems Research*, 86, 8–16.
- Arul, R., Velusami, S., & Ravi, G. (2015). A new algorithm for combined dynamic economic emission dispatch with security constraints. *Energy*, 79, 496–511.
- Mason, K., Duggan, J., & Howley, E. (2017). Multi-objective dynamic economic emission dispatch using particle swarm optimisation variants. *Neurocomputing*, 270, 188–197.
- Li, L. L., Liu, Z. F., Tseng, M. L., Zheng, S. J., & Lim, M. K. (2021). Improved tunicate swarm algorithm: Solving the dynamic economic emission dispatch problems. *Applied Soft Computing*, 108, 107504.
- Mason, K., Duggan, J., & Howley, E. (2018). A multi-objective neural network trained with differential evolution for dynamic economic emission dispatch. *International Journal of Electrical Power & Energy Systems*, 100, 201–221.
- Zhu, Y., Qiao, B., Dong, Y., Qu, B., & Wu, D. (2019). Multiobjective dynamic economic emission dispatch using evolutionary algorithm based on decomposition. *IEEE Transactions on Electrical and Electronic Engineering*, 14(9), 1323–1333.
- Shao, Z., Si, F., Wu, H., & Tong, X. (2021). An agile and intelligent dynamic economic emission dispatcher based on multi-objective proximal policy optimization. *Applied Soft Computing*, 102, 107047.
- Nourianfar, H., & Abdi, H. (2019). Solving the multi-objective economic emission dispatch problems using fast non-dominated sorting TVAC-PSO combined with EMA. *Applied Soft Computing*, 85, 105770.
- Wu, P., Zou, D., Yu, N., Zhang, G., & Kong, L. (2022). An improved NSGA-III for the dynamic economic emission dispatch considering reliability. *Energy Reports*, 8, 14304–14317.
- Qiao, B., Liu, J., & Hao, X. (2021). A multi-objective differential evolution algorithm and a constraint handling mechanism based on variables proportion for dynamic economic emission dispatch problems. *Applied Soft Computing*, 108, 107419.
- Roy, P. K., & Bhui, S. (2016). A multi-objective hybrid evolutionary algorithm for dynamic economic emission load dispatch. *International Transactions on Electrical Energy Systems*, 26(1), 49–78.
- Sundaram, A. (2022). Multiobjective multi-verse optimization algorithm to solve dynamic economic emission dispatch problem with transmission loss prediction by an artificial neural network. *Applied Soft Computing*, 124, 109021.
- Chopra, N., & Ansari, M. M. (2022). Golden jackal optimization: A novel nature-inspired optimizer for engineering applications. *Expert Systems with Applications*, 198, 116924.
- Houssein, E. H., Abdelkareem, D. A., Emam, M. M., Hameed, M. A., & Younan, M. (2022). An efficient image segmentation method for skin cancer imaging using improved golden jackal optimization algorithm. *Computers in Biology and Medicine*, 149, 106075.
- Rezaie, M., Akbari, E., Ghadimi, N., Razmjoo, N., & Ghadamyari, M. (2022). Model parameters estimation of the proton exchange membrane fuel cell by a modified golden jackal optimization. *Sustainable Energy Technologies and Assessments*, 53, 102657.
- Mahdy, A., Hasanien, H. M., Turky, R. A., & Aleem, S. H. A. (2023). Modeling and optimal operation of hybrid wave energy and PV system feeding supercharging stations based on golden jackal optimal control strategy. *Energy*, 263, 125932.
- Zhang, J., Zhang, G., Kong, M., & Zhang, T. (2023). Adaptive infinite impulse response system identification using an enhanced golden jackal optimization. *The Journal of Supercomputing*, 78, 10823–10848.
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), 67–82.
- Yue, C., Qu, B., Yu, K., Liang, J., & Li, X. (2019). A novel scalable test problem suite for multimodal multiobjective optimization. *Swarm and Evolutionary Computation*, 48, 62–71.
- Yue, C., Qu, B., & Liang, J. (2017). A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems. *IEEE Transactions on Evolutionary Computation*, 22(5), 805–817.
- Zhou, A., Zhang, Q., & Jin, Y. (2009). Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation*, 13(5), 1167–1189.
- Houssein, E. H., Mahdy, M. A., Shebl, D., Manzoor, A., Sarkar, R., & Mohamed, W. M. (2022). An efficient slime mould algorithm for solving multi-objective optimization problems. *Expert Systems with Applications*, 187, 115870.
- Yan, L., Qi, W., Liang, J., Qu, B., Yu, K., Yue, C., & Chai, X. (2023). Inter-individual correlation and dimension based dual learning for dynamic multi-objective optimization. *IEEE Transactions on Evolutionary Computation*, 27(6), 1780–1793.

30. Guerreiro, A. P., Fonseca, C. M., & Paquete, L. (2021). The hyper-volume indicator: Computational problems and algorithms. *ACM Computing Surveys (CSUR)*, 54(6), 1–42.
31. Basu, M. (2006). Particle swarm optimization based goal-attainment method for dynamic economic emission dispatch. *Electric Power Components and Systems*, 34(9), 1015–1025.
32. Basu, M. (2007). Dynamic economic emission dispatch using evolutionary programming and fuzzy satisfying method. *International Journal of Emerging Electric Power Systems*, 8(4), Article 1.
33. Alsumait, J. S., Qasem, M., Sykulski, J. K., & Al-Othman, A. K. (2010). An improved pattern search based algorithm to solve the dynamic economic dispatch problem with valve-point effect. *Energy Conversion and Management*, 51(10), 2062–2067.
34. Ghasemi, M., Akbari, E., Rahimnejad, A., Razavi, S. E., Ghavidel, S., & Li, L. (2019). Phasor particle swarm optimization: A simple and efficient variant of PSO. *Soft Computing*, 23, 9701–9718.
35. Basu, M. (2008). Dynamic economic emission dispatch using non-dominated sorting genetic algorithm-II. *International Journal of Electrical Power and Energy Systems*, 30(2), 140–149.
36. Pandit, N., Tripathi, A., Tapaswi, S., & Pandit, M. (2012). An improved bacterial foraging algorithm for combined static/dynamic environmental economic dispatch. *Applied Soft Computing*, 12(11), 3500–3513.
37. Jiang, X., Zhou, J., Wang, H., & Zhang, Y. (2013). Dynamic environmental economic dispatch using multiobjective differential evolution algorithm with expanded double selection and adaptive random restart. *International Journal of Electrical Power and Energy Systems*, 49, 399–407.
38. Zhang, H., Yue, D., Xie, X., Hu, S., & Weng, S. (2015). Multi-elite guide hybrid differential evolution with simulated annealing technique for dynamic economic emission dispatch. *Applied Soft Computing*, 34, 312–323.
39. Li, Z., Zou, D., & Kong, Z. (2019). A harmony search variant and a useful constraint handling method for the dynamic economic emission dispatch problems considering transmission loss. *Engineering Applications of Artificial Intelligence*, 84, 18–40.
40. Huang, H., Lv, L., Ye, S., & Hao, Z. (2019). Particle swarm optimization with convergence speed controller for large-scale numerical optimization. *Soft Computing*, 23, 4421–4437.
41. Hassan, M. H., Kamel, S., Domínguez-García, J. L., & El-Naggar, M. F. (2022). MSSA-DEED: A multi-objective salp swarm algorithm for solving dynamic economic emission dispatch problems. *Sustainability*, 14(15), 9785.
42. Xiong, G., Shuai, M., & Hu, X. (2022). Combined heat and power economic emission dispatch using improved bare-bone multi-objective particle swarm optimization. *Energy*, 244, 123108.
43. Liu, Z. F., Li, L. L., Liu, Y. W., Liu, J. Q., Li, H. Y., & Shen, Q. (2021). Dynamic economic emission dispatch considering renewable energy generation: A novel multi-objective optimization approach. *Energy*, 235, 121407.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.