Data-driven optimization algorithm for economic operation of microgrid based on dynamic Kriging model and two-stage search strategy

Chaodong Fan
Hunan University of Finance and Economics
Hou Bo
Xiangtan University
Xilong Qu
Hunan University of Finance and Economics
Leyi Xiao (xiaolyttkx@163.com)
Hunan University of Finance and Economics https://orcid.org/0000-0002-7343-2256
Fanyong Cheng
Aihui Polytechnic University

Research Article

Keywords: Kriging model, Two-stage search, Particle swarm algorithm, Microgrid

Posted Date: March 26th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-328600/v1

Data-driven optimization algorithm for economic operation of microgrid based on dynamic Kriging model and two-stage search strategy

Chaodong Fan[a,b,c], Bo Hou[e], Xilong Qu[a], Leyi Xiao[a,*] , Fanyong Cheng[b,d,**]

[a]*Hunan Provincial Key Laboratory of Finance & Economics Big Data Science and Technology, School of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410205, China*

[b]*Key Laboratory of Advanced Perception and Intelligent Control of High-end Equipment of Ministry of Education, Anhui Polytechnic University, Wuhu 241000, China*

[c]*Foshan Green Intelligent Manufacturing Research Institute of Xiangtan University, Guangdong 528311, China*

[d]*School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore*

[e]*School of Automation and Electronic Information, Xiangtan University, Xiangtan 411105, China*

Abstract: The economic operation optimization of microgrid is an important research topic in the power system. Therefore, this paper proposes a surrogate model particle swarm optimization algorithm based on the global-local search mechanism. Firstly, aiming at the problem that the statistical information of Kriging model is difficult to guarantee the prediction accuracy, the dynamic transformation is carried out to enhance the robustness of the model; secondly, the global-local search mechanism is introduced to make the algorithm fully explore the fitness landscape near the Kriging model after quickly locating the current optimal particle position, so as to achieve the balance of convergence quality and convergence efficiency. The proposed method has been tested on numerous benchmark test functions from two test suites, and the results show that the proposed algorithm has more advantages than other comparison algorithms in optimization accuracy. Finally, simulations are carried out in two operating modes of microgrid islanding and grid-connected, which has verified the effectiveness of the proposed method.

Keywords: Kriging model, Two-stage search, Particle swarm algorithm, Microgrid.

1. Introduction

With the aggravation of energy dilemma and environmental deterioration, renewable energy power generation technology has developed rapidly, and the emergence of microgrids provides an effective technical means for the comprehensive utilization of renewable energy [1]. Microgrid not only facilitates the access of distributed energy system, but also realizes demand-side management and maximization of existing energy utilization [2]. With the large-scale construction and investment of smart grid, the economic optimal operation of microgrid has become an important part of its integrated control and energy management research [3].

Many traditional mathematical methods, such as linear programming method [4], stochastic linear programming method [5] and stochastic dynamic programming method [6], etc., are not suitable for solving the economic operation optimization problem of microgrid with multiple micro-sources and numerous constraints due to the need to establish complex mathematical models. Considering the defects of traditional mathematical methods, scholars at home and abroad began to use swarm intelligence optimization method to solve this kind of problems. For example, Wu et al. proposed a hybrid particle swarm optimization algorithm through the hybrid immune mechanism, sub-gradient method and non-linear condition mechanism, and effectively solved the

· Corresponding author.
· · Corresponding author.
E-mail addresses: xiaolyttkx@163.com (L. Xiao), fanyong.cheng@ntu.edu.sg (F. Cheng).

economic operation problem of microgrid under the condition of configuring the rotating reserve power of controllable micro-source [7]. On the basis of considering fuel cells, Sanjaria et al. established a building microgrid optimization model for cogeneration, and applied the hypersphere search algorithm to solve the economic operation problem of the system [8]. Considering the generation cost of distributed generation, Cheng et al. proposed an extreme dynamic programming algorithm, which realizes the control and economic dispatch of microgrid by introducing the prediction module [9]. Li et al. improved the gravitational search algorithm by introducing reverse learning and elite strategy to solve the multi-objective economic operation problem of microgrid [10]. In these swarm intelligence optimization methods, most of them are based on the improved algorithm itself and then optimize the objective function. Their optimization is not only difficult to realize, but also makes the optimization time too long (from minutes to hours) due to the tens of thousands of times of fitness function evaluation. In recent years, in the field of evolutionary algorithms, when faced with the above problems, surrogate models are usually used to assist evolutionary algorithms, and cheaply calculated surrogate models are used to replace expensive actual fitness evaluation [11]. The surrogate model is used to fit the objective function with low computational cost and high prediction accuracy. However, in the current research on the economic optimization of microgrids, the use of surrogate models to deal with such problems has just begun.

Among many surrogate models, Kriging model not only has good fitting effect, but also gives the mean square error estimation of the estimated value at the same time [12]. Therefore, it is widely used in various practical engineering problems [13-15]. In this paper, Kriging model is used as the surrogate model. Based on the improved Kriging model, a particle swarm optimization algorithm (ImKPSO) combining global and local search is proposed. By modifying the correlation matrix of Kriging model, the robustness of the model is enhanced. When the global search quickly locates the current best point, the local search mechanism is activated to speed up the algorithm convergence. In order to verify the performance of the algorithm, we conducted a comprehensive test on the proposed algorithm on CEC2006 and CEC2017, and the comparison shows that the proposed algorithm has good performance in dealing with complex multi-constrained problems. Finally, the proposed algorithm is applied to solve the microgrid optimization model with economic operation as the objective. Two operation modes, island operation and grid-connected operation, are considered. The simulation results verify the effectiveness of the proposed method for solving the economic optimization problem of microgrid.

The rest of this paper is organized as follows. Section 2 briefly introduces the characteristics of each micro-source and analyzes the objective function of the microgrid in two operating modes. The proposed method, ImKPSO, is elaborated in Section 3. Sections 4 present the experimental studies on benchmark test functions. Section 5 applies the proposed algorithm to the economic operation optimization of the microgrid in two operating modes. Finally, Section 6 summarizes the paper.

2. Problem description and mathematical model

2.1 Microgrid system description

A typical microgrid model is shown in Fig. 1. The microgrid is radial, including photovoltaic arrays, wind turbines, micro gas turbines, diesel generators and energy storage batteries. Among them, micro gas turbines, diesel generators and energy storage batteries are controllable loads that participate in the economic dispatch process. Because the output power of photovoltaic arrays and wind turbines changes with climate change, it is difficult to accurately know their actual power at the current moment. They have obvious randomness and

volatility and are uncontrollable loads. The output information needs to be given before dispatch. Therefore, the photovoltaic arrays and wind turbines does not participate in actual scheduling in this article.

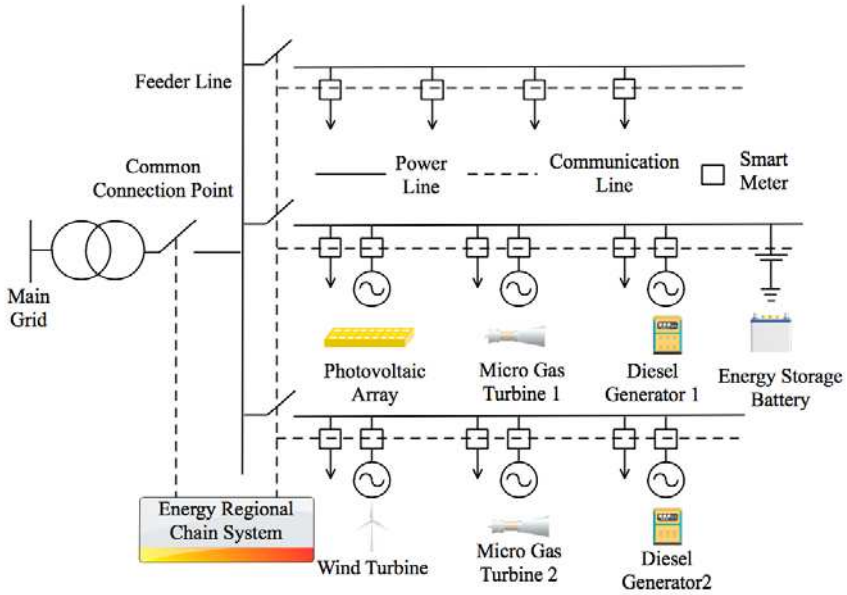

Fig. 1. Typical micro-grid structure diagram.

2.2 Microgrid subsystem model

2.2.1 Micro gas turbine

As a small distributed power generation device, micro gas turbines have flexible, safe, clean and cheap power generation with cogeneration mode, which meet the diversified development needs of micro grid energy supply. The fuel cost is as follows:

$$C_{mf}(t) = c_m \frac{p_m(t)}{\eta_m(t)} \tag{1}$$

where c_m represents the fuel price of the micro gas turbine, which is taken as 0.7￥/m^3 in this paper; $\eta_m(t)$ represents the efficiency of the micro gas turbine power generation system, which is taken as 80% in this paper.

2.2.2 Diesel generator

In a microgrid system, in order to achieve a certain economic effect, diesel generators often exist in the form of multiple relatively small-capacity units. Its operating fuel cost is shown in the following formula:

$$C_{d,f}(t) = a[P_d(t)]^2 + bp_d(t) + c \tag{2}$$

where $p_d(t)$ is the output power value of the diesel generator; the coefficients a, b, and c are the cost parameters, which can be obtained by fitting experimental data or provided by the equipment manufacturer.

2.2.3 Energy storage battery

Energy storage batteries play an important role in maintaining the energy balance of the microgrid, improving power quality, and improving the uninterrupted power supply capability of the microgrid. The degradation cost is shown in the following formula:

$$C(p_b(t)) = C\big(p_{b,c}(t)\big) + C\big(p_{b,d}(t)\big) \tag{3}$$

where $p_{b,c}(t)$, $p_{b,d}(t)$ are the charging and discharging power of the energy storage battery respectively.

2.3 Objective function

The main purpose of microgrid economic optimization problem is to formulate the optimal power supply

strategy of microgrid system to achieve the minimum operation cost. This paper also considers the objective function to be optimized under the two operation modes of microgrid.

1) Island operation mode: when the microgrid is in island operation, it is not connected with the main grid, and its objective function is expressed as follows:

$$\min C_t = C(p_b(t)) + \sum_{m=1}^{M} C(P_k(t)) + \sum_{n=1}^{N} C(P_i(t)) \tag{4}$$

where $C(p_b(t))$ is the total power of energy storage battery charging and discharging, $P_k(t)$ is the actual output value of the k uncontrollable microsource, $P_i(t)$ is the power of the i-th controllable microsource, and C_t is the total cost of the microgrid.

2) Grid-connected operation mode: when the microgrid is in grid-connected operation, it relates to the main grid, and its objective function is expressed as follows:

$$\min C_t = C(p_L(t)) + \sum_{m=1}^{M} C(P_k(t)) + \sum_{n=1}^{N} C(P_i(t)) \tag{5}$$

where p_L is the exchange power between the microgrid and the main grid.

2.4 Constrains

The constraints of the microgrid economic optimization problem include power balance constraint, controllable source power constraint, controllable source ramping constraint, energy storage battery power constraint and exchange power constraint:

1）Power balance constraint:

$$P_D(t) = p_b(t) + p_L(t) + \sum_{m=1}^{M} C(P_k(t)) + \sum_{n=1}^{N} C(P_i(t)) \tag{6}$$

2）Controllable source power constraint:

$$P_i^{min}(t) \le P_i(t) \le P_i^{max}(t) \tag{7}$$

3）Controllable source ramping constraint:

$$r_i^{down}\Delta t \le P_i(t) - P_i(t-1) \le r_i^{up}\Delta t \tag{8}$$

4）Energy storage battery power constraint:

$$-\ p_b^{max}(t) \le\ p_b(t) \le p_b^{max}(t) \tag{9}$$

5）Exchange power constraint:

$$P_L^{min}(t) \le P_L \le P_L^{max}(t) \tag{10}$$

Among them, $P_D(t)$ is the load demand of microgrid, r_i^{down} is the upper limit of slow down power of the i controllable source, and r_i^{up} is the upper limit of climbing power of the i controllable source.

3. Particle swarm optimization algorithm based on improved Kriging model

3.1 Kriging model

The Kriging model, also known as the Gaussian process, was initially applied to the estimation of geological reserves. It was a very popular geostatistical difference method, and was further extended to the design and analysis of deterministic computer experiments [16]. The Kriging model can not only give the estimated value of the unknown function, but also the error estimate of the estimated value. This error estimate can be used to guide new sample points to be added to the sample set to further improve the accuracy of the surrogate model [17]. The Kriging model regards the objective function as a specific realization of a Gaussian static random process, which can be expressed as:

$$y(X) = f(X) + z(X) \tag{11}$$

where $f(X)$ is called global trend model, which can be given as follows:

$$f(X) = [f^1(X), f^2(X),...,f^m(X)] \cdot \beta \tag{12}$$

where $\beta = (\beta^1,\beta^2,...,\beta^m)^T$ is the corresponding coefficient vector.

$z(X)$ is a random Gaussian process with zero mean. For decision space, there is correlation between sample points, and the covariance is expressed as follows:

$$\mathrm{Cov}[Z(x_i),Z(x_j)] = \sigma^2 R(\theta,x_i,x_j) \tag{13}$$

where σ^2 is the process variance, $R(\theta,x_i,x_j)$ is the correlation function, and θ is the parameter. The two most important parameters β and σ^2 of the Kriging model can be derived using the Lagrangian multiplier method:

$$\widehat{\beta} = (F^T R^{-1} F)^{-1} F^T R^{-1} Y_t \tag{14}$$

$$\widehat{\sigma}^2 = \left(Y_t - F\widehat{\beta}\right)^T R^{-1} (Y_t - F\widehat{\beta})/n \tag{15}$$

Where

$$F = [1\ 1,\ \cdots, 1]^T \in R^n \tag{16}$$

R is the $n \times n$ dimension correlation function matrix:

$$R = \begin{bmatrix} R(x_1,x_1) & \cdots & R(x_1,x_n) \\ \vdots & \ddots & \vdots \\ R(x_n,x_1) & \cdots & R(x_n,x_n) \end{bmatrix} \tag{17}$$

From the above derivation process, we can see that for the correlation function matrix R, the paired covariance of the joint model of all sample data is stored, that is, it contains the statistical information of the sample set. Normally, the correlation function matrix R is a positive definite matrix. However, under certain sample sets (For example: certain two rows or two columns of the matrix are approximately equal), R may appear singular, which makes the condition number of the matrix larger, which leads to a significant decrease in the prediction accuracy of the model or even a prediction failure (As shown in Fig. 2, when the two columns R_m and R_n of the correlation function matrix R are approximately equal, the prediction accuracy of the abnormal prediction curve shown in the circled area in Fig. 3 is greatly reduced). In order to solve this problem, most scholars adopt the method of static transformation. For example, reference [18] adopts the method of directly assigning 1 to the diagonal of the matrix R to overcome the singularity of the correlation function matrix R. However, this method only changes the value of the correlation function matrix and does not improve the prediction accuracy of the model.

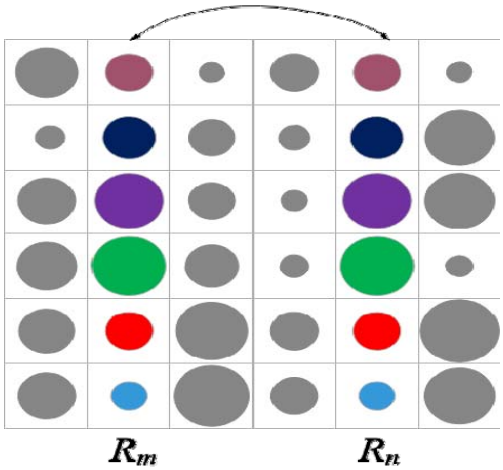

Fig. 2. Correlation function matrix R (R_m and R_n columns are approximately equal).

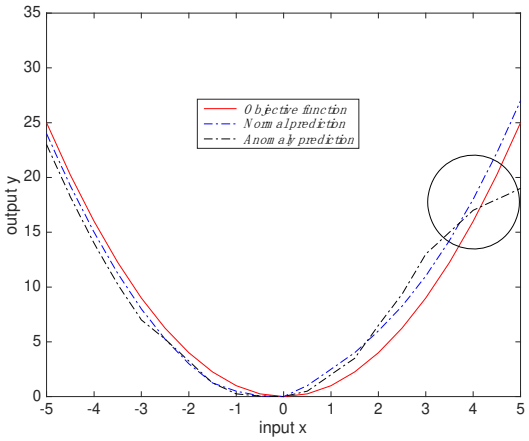

Fig. 3. Normal prediction and anomaly prediction for the test function.

As mentioned above, the modeling process of Kriging model depends on the statistical information of sample points, which is largely related to the initial sample points, and the information contained in the current population is ignored and underutilized. In addition, the construction of the correlation function matrix R cannot guarantee that it is a positive definite matrix, which may lead to the poor prediction accuracy. In view of the above two points, starting from the purpose of enhancing the prediction accuracy of the Kriging model, this

paper proposes a dynamic transformation method for the correlation function matrix R of the Kriging model. The specific steps are as follows:

Step1: Suppose the current population is D_t, and calculate its covariance matrix C.

Step2: Normalize the sample points in the covariance matrix C. The specific method is as follows, and a new matrix D is obtained.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{18}$$

Step3: Calculate $4.44 \times 10^{-13} \cdot D$ to get matrix E (Note: The coefficient must be within a reasonable range. After testing, it is found that too large a coefficient will cause a large error in the linear variance group of the Kriging model, and a too small value will not filter the numerical noise. So, this article takes 4.44×10^{-13}).

Step4: Calculate $F = R + E$, and use matrix F as the new correlation function matrix.

3.2 Particle swarm optimization algorithm

The concept of particle swarm optimization algorithm originated from the study of foraging behavior of birds. In each iteration of the algorithm, the particle updates its position according to two extreme values, which are the global historical optimal X_{gbest} and the individual historical optimal X_{pbest} [19]. The standard particle swarm optimization algorithm is given by the following update formula:

$$V_{t+1} = w_t V_t + c_1 r_1 \left(X_{pbest} - X_t\right) + c_2 r_2 \left(X_{gbest} - X_t\right) \tag{19}$$

$$X_{t+1} = X_t - V_{t+1} \tag{20}$$

where x is the position, V is the speed, w is the inertia weight, c is the learning factor, r_1、r_2 are two random numbers between 0 and 1. In this paper, c_1 and c_2 are taken as 1.49445. The adaptive inertia weight w decreases linearly with the number of iterations from 0.9 to 0.4:

$$w_g = 0.9 - \frac{g}{g_{max}} \tag{21}$$

where $g_{\max}$ is the preset maximum number of iterations.

3.3 Global-local search strategy

For particle swarm optimization (including other heuristic algorithms), the process of searching for solutions is essentially a greedy strategy. Therefore, the better solutions that do not conform to the greedy rules will be ignored by the algorithm and may fall into the local optimum. In order to overcome this shortcoming, researchers often mix other intelligent optimization algorithms or improve the control parameters to change the diversity of the population, so as to avoid the problem that the algorithm is easy to fall into local optimum (As shown in Fig. 4, the current best points x_t^i and x_t^k with the local best point x_t^l are all close to the global best point). However, this kind of "improved algorithm" that makes specific improvements for a certain type of specific problem is often not universal. In response to this problem, combined with the characteristics of the surrogate model, this paper proposes a novel two-stage optimization framework, which can avoid the algorithm falling into the local optimum on the basis of improving the convergence speed.

The global-local search scheme includes two phases: the global search phase and the local search phase. After the initial surrogate model is established, the next step is to select sample points to add to the sample set by “infill criteria”; update the surrogate model according to the new sample set, to cycle until convergence. For Kriging model, many infill criteria such as MSP criterion [20], EI criterion [21], MSE criterion [22], LCB criterion [23] have been developed internationally. Among them, the MSP criterion is widely used because of its simplicity and practicality. Its principle is to find the minimum value of the objective function directly on the surrogate model. The following will describe the MSP criterion and global-local search strategy adopted by the

proposed algorithm.

Global search phase: First, after the improved Kriging model is established, PSO immediately optimizes the global model. Assuming that the current sample set is D_t, the current optimal solution found is x_t^k, that is:

$$x_t^k = arg\,min\hat{f} \tag{22}$$

Then, the actual objective function (the total cost of microgrid in this paper) is used to evaluate x_t^k. Assume that the actual fitness value obtained is y_t^k, then $(x_t^k,\ y_t^k)$ is added as a new sample point to D_t. The updated sample set is D_{t+1}, and then the surrogate model is rebuilt in the new sample set to improve its prediction accuracy.

Local search phase: In order to further accelerate the convergence, a local search mechanism is introduced on the basis of global search. As shown in Fig. 5, if the current optimal solution found by global search is x_t^k, a local improved Kriging model is established based on the K points nearest to x_t^k. in this paper, K takes 20% of the current population size.

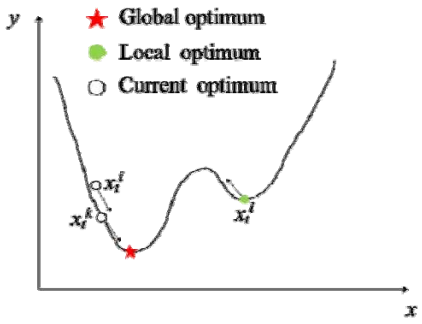

Fig. 4. Speed up the convergence.

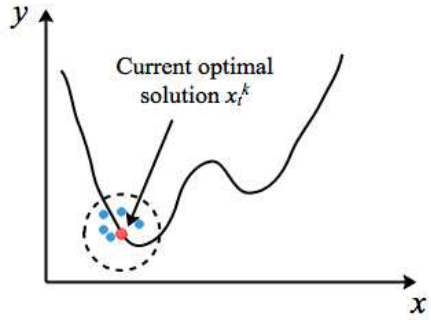

Fig. 5. Schematic diagram of ImKPSO local search mechanism.

After the local surrogate model is established, the process is the same as the global search process. PSO starts to optimize the local model. After the current optimal solution is evaluated by objective function, new sample points are obtained and added to the sample set. The algorithm takes the global search as the starting point. When the accuracy of the solution obtained by the global search cannot be improved any more, it will be converted into local search; when the accuracy of the solution obtained by the local search cannot be improved any more, it will return to the global search, and the cycle will continue until the preset termination conditions are reached. For convenience, we refer to this process as the sub-optimization process in the following text.

3.4 Algorithm steps

The main ideas of the data-driven optimization algorithm proposed in this paper are as follows: the correlation function matrix of Kriging model is dynamically modified to improve the prediction accuracy of Kriging model; the standard particle swarm optimization algorithm is used in the optimization algorithm; according to the characteristics of the surrogate model, the improved Kriging model is combined with the global-local search mechanism to achieve the balance of convergence quality and efficiency. The algorithm flow chart is as Fig. 6:

The detailed steps of the proposed algorithm (ImKPSO) are as follows:

Step1: Use Latin hypercube sampling to initialize particles, $p_t =< X_1^t,..., X_{NP}^t > (t = 1)$. The advantage of choosing Latin hypercube sampling is that it can maximize the information of the initial population with fewer sample points [24].

Step2: Use the objective function to evaluate and obtain $< y_1^t,..., y_{NP}^t > (t = 1)$.

Step3: An improved surrogate model is established by the population:

$$D_t = \{(X_1^t, y_1^t),\ldots,(X_{NP}^t, y_{NP}^t)\}(t = 1,2,\ldots,t_{Termination}).$$

Step4: The standard particle swarm optimization algorithm is used as the search engine, and the sub-optimization process starts to optimize.

Step5: The current optimal solution is evaluated by the objective function, and the new sample points are added to the original database.

Step6: When the algorithm does not reach the preset termination condition, it returns to Step3 to continue iterating. When the preset termination condition is reached, the algorithm terminates and outputs the optimal solution.

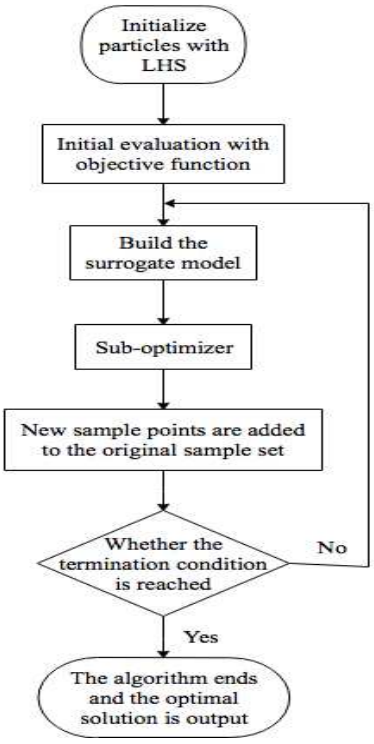

Fig. 6. ImKPSO algorithm flowchart.

4. Benchmark test

In order to verify the various components and overall performance of ImKPSO, 13 test functions in CEC2006 and 9 test functions in CEC2017 (including 10D and 30D) are selected for comprehensive test, and the maximum number of calls ($maxFEs$) of the objective function is taken as the termination condition. Table 1 and Table 2 list the basic characteristics of these test functions. Among them, n is the number of decision variables, ρ is the estimated ratio of the feasible region to the search space, LI is the number of linear inequality constraints, NI is the number of nonlinear inequality constraints, LE is the number of linear equality constraints, NE is the number of nonlinear equality constraints, and $f(x^*)$ is the currently known optimal solution of each test function. It is worth noting that the optimal solutions of some test functions in the list are extremely difficult to obtain (for example: g02 is a high-dimensional nonlinear function with multiple local optima). Based on the above considerations, in this section, we set $maxFEs$=3000.

4.1 Performance analysis of improved correlation function matrix

In order to improve the prediction accuracy of the model, this paper first improves the correlation function matrix of Kriging model. In order to verify its effectiveness, a variant of ImKPSO, ImKPSO-or is used as a

comparison algorithm. ImKPSO-or uses Kriging model without any improvement as the surrogate model, and the others remain unchanged. Since the currently known optimal values $f(x^*)$ of these 13 test functions are all

Table 1. 13 test functions in CEC2006

Function	n	Type	ρ	LI	NI	LE	NE	$f(x^*)$
g01	13	quadratic	0.0111%	9	0	0	0	-15.0000000000
g02	20	nonlinear	99.9971%	0	2	0	0	-0.8036191042
g04	5	quadratic	52.1230%	0	6	0	0	-30665.5386717834
g06	2	cubic	0.0066%	0	2	0	0	-6961.8138755802
g07	10	quadratic	0.0003%	3	5	0	0	24.3062090681
g08	2	nonlinear	0.8560%	0	2	0	0	-0.0958250415
g09	7	polynomial	0.5121%	0	4	0	0	680.6300573745
g10	8	linear	0.0010%	3	3	0	0	7049.2480205286
g12	3	quadratic	4.7713%	0	1	0	0	-1.0000000000
g16	5	nonlinear	0.0204%	4	34	0	0	-1.9051552586
g18	9	quadratic	0.0000%	0	13	0	0	-0.8660254038
g19	15	nonlinear	33.4761%	0	5	0	0	32.6555929502
g24	2	linear	79.6556%	0	2	0	0	-5.5080132716

Table 2. 9 test functions in CEC2017

Function	Type	LI	NI	Search space
*f*01	Non Separable	0	1	$[-100, 100]^D$
*f*02	Non Separable, Rotatad	0	1	$[-100, 100]^D$
*f*04	Separable	0	2	$[-10, 10]^D$
*f*05	Non Separable	0	2	$[-10, 10]^D$
*f*13	Non Separable	2	1	$[-100, 100]^D$
*f*19	Separable	0	2	$[-50, 50]^D$
*f*20	Non Separable	0	2	$[-100, 100]^D$
*f*22	Rotatad	2	1	$[-100, 100]^D$
*f*28	Rotatad	0	2	$[-50, 50]^D$

known, this section uses the function error value $(f(x_{best}) - f(x^*))$ as an observation value. Each group of experiments was run 25 times independently, and the mean and standard deviation of 25 times were taken as the results. In addition, Wilcoxon's rank sum test with significance level of 0.05 was performed between ImKPSO-or and ImKPSO. The statistical results of the two algorithms are shown in Table 3.

It can be seen from Table 3 that ImKPSO is superior to ImKPSO-or in 8 test functions of the test set in the table, and on the other 5 test functions, ImKPSO has at least the same performance as ImKPSO-or. In addition, the performance of ImKPSO is significantly better than ImKPSO-or on the functions of g10, g12, g16, g18, and g19. The above results are because the statistical information contained in the current population has been fully utilized in the stage of establishing the surrogate model of ImKPSO. Compared with ImKPSO-or, the improved Kriging model of ImKPSO can better fit the objective function. The above analysis shows that the improvement of the correlation function matrix in the proposed algorithm can effectively improve the prediction accuracy of the Kriging model.

Table 3. Test results of ImKPSO-or and ImKPSO on 13 functions of CEC2006

Function	ImKPSO-or	ImKPSO
	（Ave± Std）	（Ave± Std）
g01	2.59E-06±1.17E-06	2.66E-06±3.31E-06
g02	4.85E-01±2.20E-01	**3.91E-01±2.17E-01**
g04	2.05E-07±6.52E-08	2.17E-07±7.70E-08
g06	3.39E-04±2.51E-04	3.24E-04±5.26E-05
g07	1.19E-06±3.50E-07	1.03E-06±2.77E-07
g08	7.22E-08±6.37E-08	**5.84E-08±4.81E-09**
g09	5.27E-07±2.68E-07	**3.46E-07±2.11E-07**
g10	7.72E-05±5.27E-06	**2.30E-06±3.71E-06**
g12	2.93E-03±2.27E-03	**5.30E-04±2.51E-04**
g16	6.41E-07±4.88E-07	**2.92E-08±1.03E-08**
g18	5.82E-04±3.60E-04	**6.67E-05±3.35E-05**
g19	2.44E-02±6.30E-03	**3.73E-03±2.21E-03**
g24	4.97E-05±2.55E-05	4.76E-05±3.34E-06

Table 4. The statistical results of the optimal solutions obtained by ImKPSO-Glo, ImKPSO-Loc, ImKPSO @ CEC2006.

Function	ImKPSO-Glo	ImKPSO-Loc	ImKPSO
g01	2.63E-02 (3.92E-01)	7.20E-01 (3.55E-01)	**9.97E-05 (4.33E-06)**
g02	5.74E-01 (4.07E-02)	1.12E-01 (3.40E+00)	**2.85E-03 (4.28E-04)**
g04	4.81E+00 (2.66E+02)	7.35E+01 (4.66E+04)	**3.61E-07 (7.20E-05)**
g06	8.49E-01 (2.56E-01)	2.97E+03 (4.58E+04)	**5.54E-04 (8.31E-04)**
g07	7.33E+01 (5.12E+03)	3.43E+03 (8.20E+03)	**4.88E-09 (6.37E-07)**
g08	4.08E-03 (3.35E-01)	5.79E-02 (2.66E+01)	**3.35E-11 (8.54E-09)**
g09	6.19E+03 (2.07E+04)	3.13E+02 (4.02E+03)	**1.08E-05 (3.34E-06)**
g10	3.14E-02 (6.70E-02)	2.07E-02 (1.72E-01)	**5.99E-07 (3.14E-05)**
g12	2.33E-03 (8.35E-01)	1.20E-03 (5.58E-01)	**7.82E-08 (2.66E-06)**
g16	6.46E-02 (7.29E+00)	7.16E-01 (6.55E+04)	**8.11E-06 (5.19E-07)**
g18	4.03E+01 (2.90E+02)	3.59E+02 (268E+02)	**7.04E-10 (3.49E-08)**
g19	8.85E-01 (4.31E+00)	4.16E-01 (5.83E+01)	**2.82E-06 (5.60E-06)**
g24	2.39E-03 (5.21E-01)	6.77E-05 (4.30E-02)	**6.01E-07 (5.05E-09)**

4.2 Performance analysis of global-local search mechanism

The search process of ImKPSO is based on a global-local combination of search. In order to verify its performance, two variants of ImKPSO are used as comparison algorithms, namely ImKPSO-Glo and ImKPSO-Loc. The former retains the global search and eliminates the local search; the latter retains the local search and eliminates the global search. In order to verify the influence of these two parts on the performance of the algorithm, this section selects 13 test functions in CEC2006 for testing. The experimental results are taken in the same way as in section 4.1. Table 4 records the statistical results (mean and standard deviation) of the three algorithms on 13 test functions. Fig. 7 shows the convergence curves of the three algorithms on the functions g04, g08 and g12. It can be seen from Table 4 that whether it is ImKPSO-Glo or ImKPSO-Loc, the overall

performance is far inferior to ImKPSO, and the statistical results often differ by several orders of magnitude. In addition, it can be seen from Fig. 7 that the convergence trends of the three algorithms on the three test functions are basically similar. Whether it is ImKPSO-Glo or ImKPSO-Loc, the performance is not as good as ImKPSO. In the case of no local search, the global search optimization process is slow. This is because the global search searches in the entire feasible region, while the local search is responsible for exploring the fitness landscape near the current optimal particle. The above analysis shows that the introduction of local search can effectively improve the optimization efficiency of the algorithm, especially in the later stage of the algorithm. However, if the global part is removed and only the local search is retained, the algorithm can easily fall into the local optimum.

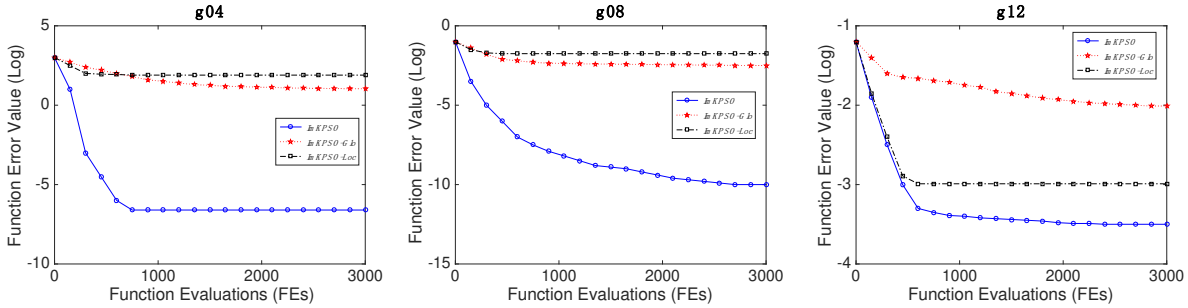

Fig. 7. Convergence graphs of three algorithms on three functions of CEC2006.

4.3 Comparison with Four Methods in CEC2006

In order to verify the performance of ImKPSO in dealing with complex optimization problems, this section uses the 13 test functions in Table 1 to test ImKPSO, and makes horizontal comparison with DE [25], PSO [26], SWT-PSO [27], iDEaSm [28]. SWT-PSO is a particle swarm optimization algorithm based on random weight balance proposed by Mohan et al. In consideration of distributed power generation, Mohan et al. used SWT-PSO to solve the economic optimization operation of the microgrid, and verified that the algorithm has strong practicability in solving this problem. iDEaSm is a surrogate-assisted optimization algorithm based on an improved difference algorithm. The algorithm adapts the scaling factor F and combines the improved difference algorithm with the Kriging model. Comparing with algorithms such as JADE[29], ESMDE[30], L-SHADE[31], etc., it is shown that iDEaSm has strong performance when dealing with constrained optimization problems. For the fairness of comparison, the selected algorithms all set the maximum number of calling the objective function as the algorithm termination condition. In this section, set maxFES=3000, and each group of experiments run 25 times independently. Table 5 shows the mean and variance of each algorithm. In the table, "NF" means that no feasible solution was found in 25 independent runs.

It can be seen from Table 5 that ImKPSO outperforms the other 4 algorithms on 12 of the test functions, and only performs worse than iDEaSm on g24. This is because one of the ImKPSO falls into a local optimum (experimental results are retained). On the 8 test functions of g01, g06, g07, g08, g09, g10, g18 and g19, the average function error of ImKPSO is several orders of magnitude lower than that of the other 4 algorithms. At the same time, comparing the variances, it is found that the variance is also the smallest on the 12 test functions that ImKPSO obtains the optimal average error. The above results show that ImKPSO has a better performance than the comparison algorithm in terms of accuracy and stability. It can also be seen from Table 5 that although iDEaSm is inferior to ImKPSO in most test functions, it has more advantages compared to SWT-PSO. Only in the five test functions of g08, g09, g10, g12 and g18, the performance is weaker than SWT-PSO, which indicates that the surrogate-assisted optimization algorithm has strong competitiveness in solving such constrained optimization problems.

Based on the above comparative analysis results, ImKPSO has significant advantages in optimization accuracy and stability whether it is compared with the same type of surrogate-assisted optimization algorithm iDEaSm in the horizontal direction or compared with the improved evolutionary algorithm SWT-PSO in the vertical direction.

Table 5. The statistical results of the optimal solutions obtained by different algorithms @ CEC2006.

Function		DE	PSO	SWT-PSO	iDEaSm	ImKPSO
g01	Ave	5.53E+01	2.02E+01	8.77E+00	5.32E-01	**3.39E-07**
	Std	4.09E+01	5.43E+01	1.06E+00	8.66E-01	**4.26E-07**
g02	Ave	9.97E-01	1.06E+00	6.90E-01	**4.11E-01**	**3.88E-01**
	Std	2.64E-01	3.03E-01	4.88E-02	**5.92E-02**	**2.01E-02**
g04	Ave	2.88E+04	2.23E+03	8.82E+02	2.84E-06	**3.21E-07**
	Std	5.26E+03	7.72E+03	5.60E+01	3.55E-06	**1.19E-07**
g06	Ave	7.08E+01	7.99E+02	7.53E+02	7.26E-03	**7.76E-06**
	Std	8.61E+01	5.15E+02	3.01E+02	2.08E-02	**2.21E-05**
g07	Ave	2.09E-01	6.81E+01	6.18E+01	7.07E-01	**9.12E-08**
	Std	8.10E-02	7.09E+01	4.55E+01	9.23E-01	**4.21E-08**
g08	Ave	3.79E-02	6.44E-03	2.41E-05	6.52E-02	**7.56E-11**
	Std	2.18E-02	8.11E-03	9.27E-05	3.57E-02	**2.28E-11**
g09	Ave	4.18E+01	5.56E+02	5.31E+01	8.11E+02	**5.81E-07**
	Std	7.20E+01	3.34E+02	3.88E+01	1.93E+02	**6.17E-07**
g10	Ave	3.37E+02	7.11E+02	9.60E+01	4.22E+02	**3.70E-06**
	Std	6.02E+02	5.59E+01	2.88E+01	8.66E+01	**4.62E-06**
g12	Ave	7.18E-03	6.61E-03	5.71E-04	1.90E-03	**5.77E-04**
	Std	2.09E-03	1.39E-04	6.65E-04	5.51E-03	**2.54E-04**
g16	Ave	1.15E-01	2.00E-01	2.44E-01	3.66E-07	**7.11E-08**
	Std	1.18E-01	7.55E-02	7.26E-02	2.90E-07	**7.64E-09**
g18	Ave	9.65E-01	9.77E-01	6.55E-01	7.03E-01	**5.77E-05**
	Std	4.88E-01	4.33E-01	3.48E-01	5.77E-02	**4.57E-04**
g19	Ave	NF	2.09E+02	9.28E+02	7.99E+00	**2.77E-05**
	Std	NF	3.48E+02	8.27E+01	4.25E+00	**2.21E-05**
g24	Ave	2.85E-02	6.99E-02	7.22E-03	**8.67E-11**	6.19E-08
	Std	5.11E-02	2.05E-03	6.34E-03	**3.76E-11**	5.38E-09

4.4 Comparison with Two State-of-the-art Methods in CEC2017

In order to further explore the ability of ImKPSO to explore complex problems, in this section, we test ImKPSO on 9 functions in CEC2017 (including 10D and 30D), and select two state-of-the-art algorithms DE-AOPS [32] and GLoSADE [33] to compare with ImKPSO. The following is the brief information of these two comparison algorithms:

DE-AOPS: An improved differential evolution algorithm that can adaptively select the scaling factor F and the cross-factor CR within a given continuous range. The test results show that DE-AOPS can effectively reduce the number of evaluations of the fitness function while ensuring the accuracy of convergence (the original results point out: Compared with the comparison algorithm used in the article, DE-AOPS was able to reduce the average FEs by 13.29% and 23.06 % for the 10D and 30D problems, respectively).

GLoSADE: A surrogate-assisted evolutionary algorithm based on surrogate ensembles (GRNN and RBF). GLoSADE uses the best fitness value and the maximum uncertainty as the infill criteria, and has achieved good convergence accuracy. The test results show that on most of the benchmark functions of the three test suites CEC2006, CEC2010 and CEC2017, the performance of GLoSADE is better than the comparison algorithm used in the original paper (for example: mWiE [34], FROFI [35], etc.).

Two Comparison algorithms use the same parameter settings as their source reports. In addition, this section sets the maximum number of calls to the objective function as the algorithm termination condition, and set maxFEs=3000. Each group of experiments was run 25 times independently. The statistical results of the final optimal values (mean and standard deviation) of all algorithms are listed in Table 6. In order to judge whether there is a statistical significance between ImKPSO and other two algorithms, we did the Wilcoxon's rank sum test at a 0.05 significance level for results. In the table, "NF" means that no feasible solution was found in 25 independent runs.

Table 6. The statistical results of the optimal solutions obtained by DE-AOPS, GLoSADE and ImKPSO @ CEC2017.

Function（10D）	DE-AOPS	GLoSADE	ImKPSO
f01	3.77E+01 (1.14E+00)	4.96E+02 (2.87E+01)	**3.41E+00 (5.40E+01)**
f02	2.21E+03 (3.55E+02)	2.66E+01 (1.08E+01)	**2.20E+01 (2.62E+01)**
f04	7.35E+01 (7.45E+02)	7.71E+00 (6.29E+03)	**6.55E+00 (2.07E+00)**
f05	6.60E+04 (6.30E+01)	4.32E+03 (5.40E+01)	**3.05E+02 (3.32E+01)**
f13	**1.19E+02 (5.72E+00)**	5.45E+03 (2.11E+02)	4.11E+02 (8.13E+02)
f19	NF	3.99E+03 (8.75E+01)	**3.22E+01 (9.04E+02)**
f20	6.50E+03 (4.28E+02)	6.82E+02 (2.02E+01)	**2.60E+02 (3.92E+01)**
f22	**2.44E+02 (4.56E+04)**	3.37E+02 (5.51E+04)	7.90E+02 (5.77E+03)
f28	NF	NF	NF
Function（30D）	DE-AOPS	GLoSADE	ImKPSO
f01	3.77E+03 (6.65E+03)	**1.00E+02 (4.09E+04)**	3.43E+02 (5.36E+04)
f02	6.20E+01 (8.93E+01)	7.72E+01 (2.27E+02)	**5.36E+01 (5.21E+02)**
f04	3.16E+02 (6.22E+03)	6.96E+00 (5.36E+02)	**1.27E+00 (2.74E+02)**
f05	4.94E+03 (3.34E+01)	7.55E+03 (4.88E+05)	**6.99E+00 (1.52E+03)**
f13	NF	NF	NF
f19	NF	NF	NF
f20	4.04E+00 (6.53E+01)	**3.20E+00 (2.01E+04)**	6.24E+00 (4.60E+00)
f22	5.74E+01 (1.77E+02)	2.91E+01 (7.92E+01)	**8.20E+00 (3.32E+01)**
f28	NF	NF	NF

It can be seen from Table 6 that ImKPSO shows excellent performance compared with the other two comparison algorithms on 10D and 30D problems. On 10D, ImKPSO is weaker than DE-AOPS only on f13 and f22. Except for f28, ImKPSO has better accuracy than DE-AOPS and GLoSADE in the remaining six functions. f28 is a rotated function, its search space is relatively large, and it is not easy to find its optimal solution. We can see from the table that when maxFEs=3000, none of the three algorithms can find the optimal solution. On 30D, there are two points worth noting: First, the performance of the two surrogate-assisted evolutionary algorithms GLoSADE and ImKPSO are better than DE-AOPS. As the dimensionality increases, the objective function becomes more and more complex, and it is quite difficult for heuristic algorithms to achieve satisfactory

optimization results within a limited number of objective function calls. The satisfactory optimization effect of DE-AOPS depends on many objective function calls. In its original paper, when solving the 30D problem, the objective function calls set by DE-AOPS are as high as 600,000. In addition, we found that GLoSADE outperforms ImKPSO only on $f01$ and $f20$. GLoSADE selects GRNN and RBF to build surrogate ensembles. Although GRNN has a smooth fitness landscape, it does not fit the objective function well; while RBF has excellent performance on high-dimensional problems. Its algorithm complexity is relatively high ($O(DN^3)$), therefore, in order to achieve a satisfactory convergence effect, GLoSADE adopts both the best fitness value and the maximum uncertainty as the infill criteria. However, the method of assigning the number of objective function calls to these two criteria is random. When the algorithm searches at the maximum uncertainty, it cannot fully explore the fitness landscape at the best fitness value, which makes the performance of the algorithm still limited when solving complex optimization problems. It can be seen that ImKPSO has a strong advantage when dealing with complex optimization problems.

5. Instance simulation

This section uses the typical microgrid example system shown in Fig. 1. In the microgrid, the micro power supply is composed of photovoltaic arrays, micro gas turbines, diesel generators and wind turbines, and the energy storage unit is composed of energy storage batteries. The power load demand and power market price in the microgrid system are shown in Fig. 8-9. The predicted power curves of the two uncontrollable micro-source photovoltaic arrays and wind turbines are shown in Fig. 10-11. The main parameters of each micro-source are shown in Table 7. The parameters a, b, and c in Table 7 are the quadratic, primary and constant coefficients between the operating fuel cost and output power of diesel generators.

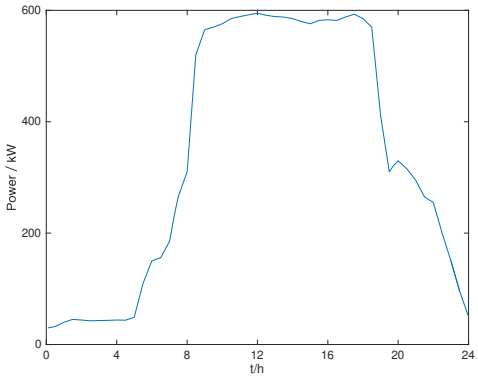

Fig. 8. Power load demand curve

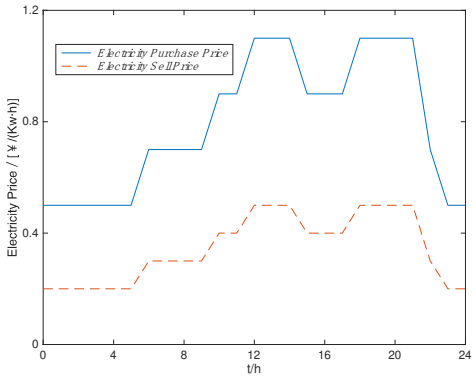

Fig. 9. Electricity market price curve

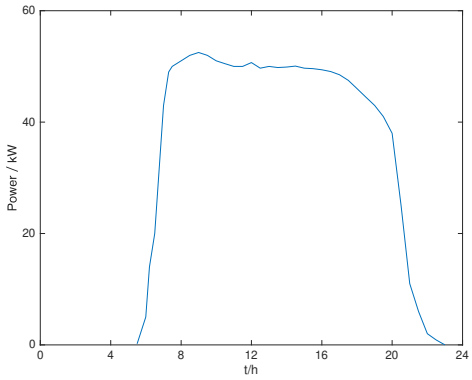

Fig. 10. Photovoltaic array power curve

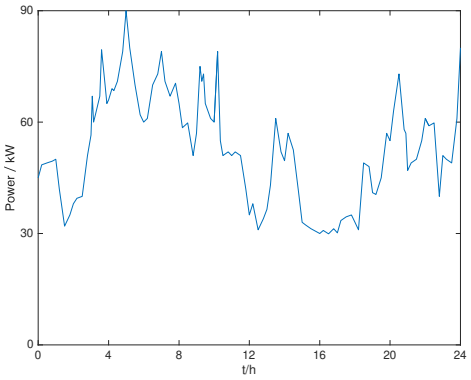

Fig. 11. Wind turbine power curve

Table 7. Operating cost data for each power resources

Type	Upper power limit /kW	Lower power limit /kW	Upper limit of climbing /kW	Upper limit of ramp down /kW	Generation cost /kW	Operating cost /kW	Parameter *a*	Parameter *b*	Parameter *c*
Photovoltaic array	0	100	–	–	1.50	0.08	–	–	–
Micro gas turbine 1	5	65	3	-5	–	0.11	–	–	–
Micro gas turbine 2	5	65	3	-5	–	0.11	–	–	–
Diesel generator 1	0	200	5	-5	–	0.08	0.00084	0.0192	5.62
Diesel generator 2	0	250	8	-8	–	0.10	0.00085	0.0212	6.87
Wind Turbine	0	130	–	–	0.60	0.15	–	–	–
Energy storage battery	-150	150	–	–	1.70	0.02	–	–	–

5.1 Economic operation optimization under island operation mode of microgrid

This section first considers that the microgrid works in island operation mode, and compares the proposed algorithm with DE, PSO, SWT-PSO and iDEaSm in island operation mode. All algorithms were run 25 times independently under the same experimental conditions, and the experimental results were averaged. Each algorithm takes the maximum number of calls to the objective function (maxFEs) as the termination condition. Set maxFEs=3000 in this section. Table 8 lists the optimal average values of each algorithm. It can be seen from Table 8 that, compared with the other four comparison algorithms, ImKPSO obtains the lowest average operating cost in island operation mode.

Fig. 12 shows the comparison of the convergence curves of various algorithms in island operation mode. It can be seen from Fig. 12 that, comparing the other four algorithms, ImKPSO not only has the best convergence accuracy, but also has the fastest convergence speed. Although SWT-PSO uses a random weight balance mechanism to balance the global search and local exploration capabilities of particles, it can be seen from the figure that both the convergence accuracy and the convergence speed are not as good as the two surrogate-assisted optimization algorithms iDEaSm and ImKPSO. Compared with iDEaSm, which also uses the Kriging model, ImKPSO's improvements to the correlation function matrix of the Kriging model make ImKPSO's overall performance even better.

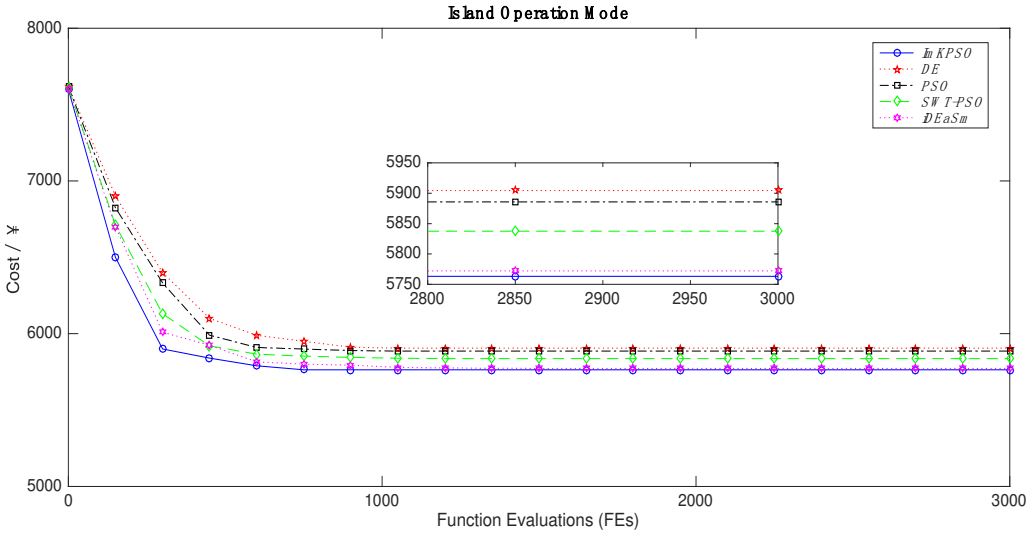

Fig. 12. Convergence curve of each algorithm in island operation mode

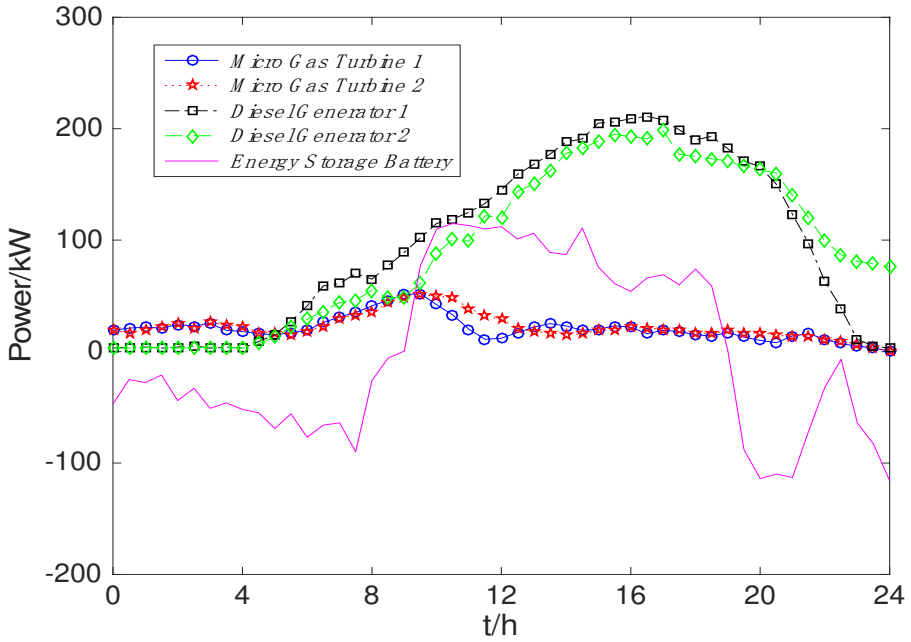

Fig. 13. System output optimization results in island mode

Fig. 13 shows the system output optimization results of the microgrid in island operation mode (dispatching in a 30-minute cycle). As can be seen from the figure, due to the low power supply cost of micro gas turbines and diesel generators, the load demand is mainly supplied by micro gas turbines and diesel generators. From 1 to 9 o'clock and 19 to 24 o'clock, the generation power of renewable energy such as photovoltaic arrays and wind turbines increases, which exceeds the load demand. Therefore, the energy storage battery is charged to store the excess energy. At 9-19, when the load demand is greater than the total power of distributed generation, the energy storage battery starts to release electrical energy.

5.2 Economic operation optimization under the grid-connected operation mode of microgrid

This section considers that the microgrid works in grid-connected operation mode. In the grid-connected operation mode, the microgrid adopts the time-of-use price strategy. Compare the proposed algorithm with DE, PSO, SWT-PSO and iDEaSm in grid-connected operation mode. The setting is the same as the island operation mode. All algorithms are independently run 25 times under the same experimental conditions, and the experimental results are averaged. Each algorithm regards the maximum number of calling the objective function (maxFEs) as the termination condition, and set maxFEs=3000. Table 9 lists the optimal average values of the five algorithms in the grid-connected operation mode. From Table 9, it can be seen that ImKPSO still achieves the lowest average operating cost compared to the other four comparison algorithms.

Table 8. Average operation cost of each algorithm in island operation mode

Algorithms	Average operating cost（￥）
DE	5904.56
PSO	5885.61
SWT-PSO	5837.25
iDEaSm	5772.37
ImKPSO	5763.13

Table 9. Average operation cost of each algorithm in grid connected operation mode

Algorithms	Average operating cost（￥）
DE	3519.62
PSO	3517.48
SWT-PSO	3496.14
iDEaSm	3475.83
ImKPSO	3440.62

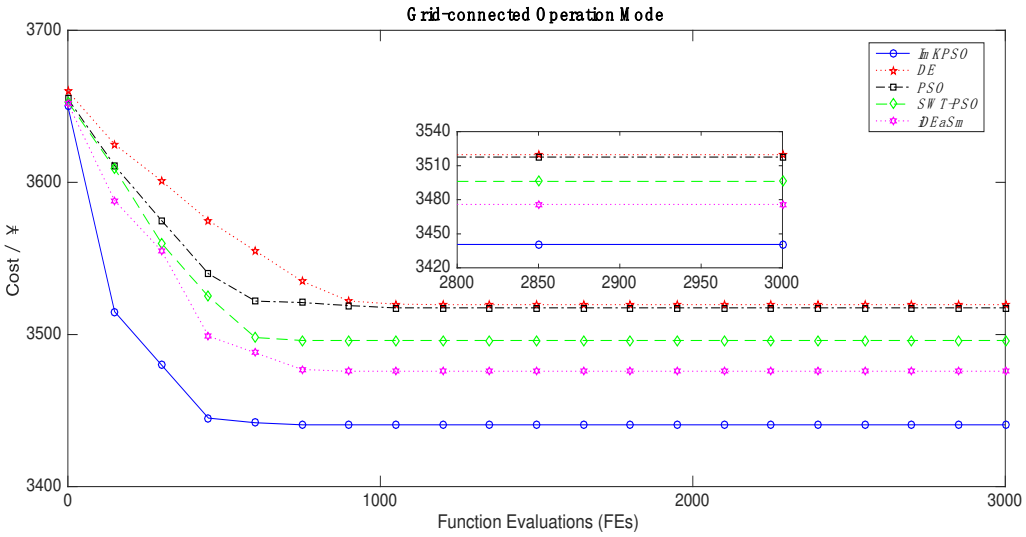

Fig. 14. Convergence curve of each algorithm in grid connected operation mode

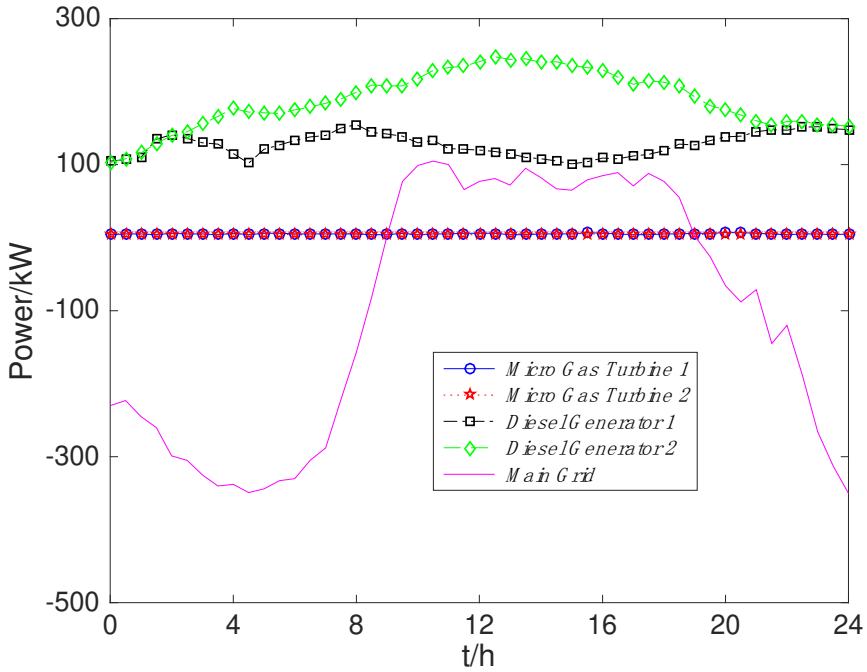

Fig. 15. System output optimization results in grid connected operation mode

Fig. 14 shows the comparison of convergence curves of various algorithms under grid-connected operation mode. It can be seen from Fig. 14 that, compared with the other four algorithms, ImKPSO has the least number of calls to the objective function to find the optimal solution and the fastest convergence speed. By comparing the convergence curves of the algorithms in Fig. 10 under island operation mode, it can be seen that the optimization performance of ImKPSO is more obvious than the other four algorithms in grid-connected operation mode.

Fig. 15 shows the system output optimization results of the microgrid in the grid-connected operation mode. Due to the connection with the main grid, the power relationship in the microgrid system tends to be relatively balanced. From 1 to 9 o'clock and 19 to 24 o'clock, the generation power of renewable energy sources such as photovoltaic arrays and wind turbines increases, which exceeds the load demand. Therefore, the microgrid transmits excess power to the main grid. From 9 to 19 hours, the load demand is greater than the total generation

power of distributed generation, and the microgrid absorbs electrical energy from the main grid.

The above results show that the proposed algorithm has excellent performance in solving the economic optimization operation of microgrid.

6. Conclusion

In this paper, a data-driven optimization algorithm is proposed and applied to the economic optimal operation of microgrid. The algorithm first improves the Kriging model, and introduces a global-local search mechanism on this basis; by combining the inexpensive surrogate model with the particle swarm optimization algorithm, it not only guarantees the quality of the solution, but also improves the convergence speed. In order to verify the effectiveness of the proposed algorithm, 13 test functions in CEC2006 and 9 test functions in CEC2017 were tested, At the same time, the economic optimization operation of microgrid is simulated and verified, the results showed that the proposed algorithm has strong advantages when solving such computationally complex optimization problems.

Acknowledgments

The authors are very grateful to the anonymous reviewers for their valuable comments on improving this article. Additionally, this work is supported by the Hunan Provincial Natural Science Foundation of China (2020JJ4587), Guangdong Basic and Applied Basic Research Foundation (2019A1515110423), Degree & Postgraduate Education Reform Project of Hunan Province (2019JGYB115).

Conflict of Interest

The authors declare that they have no conflict of interest.

References

[1] E. Harmon, U. Ozgur, M. H. Cintuglu, R. de Azevedo, K. Akkaya, and O. A. Mohammed, “The internet of microgrids: A cloud-based framework for wide area networked microgrids,” IEEE Transactions on Industrial Informatics, vol. 14, no. 3, pp. 1262–1274, March 2018.

[2] C. Stevanoni, Z. De Gre`ve, F. Valle e, and O. Deblecker, “Long-term planning of connected industrial microgrids: A game theoretical ap- proach including daily peer-to-microgrid exchanges,” IEEE Transactions on Smart Grid, vol. 10, no. 2, pp. 2245–2256, March 2019.

[3] B. Fan, S. Guo, J. Peng, Q. Yang, W. Liu, and L. Liu, “A consensus- based algorithm for power sharing and voltage regulation in dc micro- grids,” IEEE Transactions on Industrial Informatics, vol. 16, no. 6, pp. 3987–3996, June 2020.

[4] D. Livengood and R. Larson, “The energy box: Locally automated optimal control of residential electricity usage,” Service Science, vol. 1, pp. 1–16, 03 2009.

[5] A. Chaouachi, R. M. Kamel, R. Andoulsi, and K. Nagasaka, “Multi- objective intelligent energy management for a microgrid,” IEEE Trans- actions on Industrial Electronics, vol. 60, no. 4, pp. 1688–1699, April 2013.

[6] Luu Ngoc An and Tran Quoc-Tuan, “Optimal energy management for grid connected microgrid by using dynamic programming method,” in 2015 IEEE Power Energy Society General Meeting, July 2015, pp. 1–5.

[7] D.-H. Wu, C. Gao, and Z.-C. Ji, “Economic optimization operation of the microgrid using the hybrid particle swarm optimization algorithm,” Kongzhi Lilun Yu Yingyong/Control Theory and Applications, vol. 35, pp. 457–467, 04 2018.

[8] M. Sanjari, A. H. Yatim, and G. B. Gharehpetian, “Applicationofhyper- spherical search algorithm for optimal energy resources dispatch in residential microgrids,” Applied Soft Computing Journal, vol. 37, pp. 15–23, 12 2015.

[9] J. Chen, T. Yu, L. Yin, J. Jian, and J. Tang, “Integrated dispatch and control of microgrid based onextreme dynamic programming algorithm,” Control Theory and Applications, vol. 36, no. 10, pp. 1698–1706, 10 2019.

[10] P. Li, W. Xu, Z. Zhou, and R. Li, “Optimized operation of microgrid based on gravitational search algorithm,” vol. 34, no. 34, 10 2013, pp. 338–342.

[11] A. T. W. Min, Y. Ong, A. Gupta, and C. Goh, “Multiproblem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems,” IEEE Transactions on Evolutionary Computation, vol. 23, no. 1, pp. 15–28, Feb 2019.

[12] Z.-H. Han, “Kriging surrogate model and its application to design optimization: A review of recent progress,” Acta Aeronautica ET As- tronautica Sinica, vol. 37, pp. 3197–3225, 11 2016.

[13] Y. Jin, H. Wang, T. Chugh, D. Guo, and K. Miettinen, “Data-driven evolutionary optimization: An overview and case studies,” IEEE Trans- actions on Evolutionary Computation, vol. 23, no. 3, pp. 442–458, June 2019.

[14] S. Ahmed, C. Grabher, H. Kim, and T. Koseki, “Multifidelity surrogate assisted rapid design of transverse-flux permanent magnet linear syn- chronous motor,” IEEE Transactions on Industrial Electronics, vol. 67, no. 9, pp. 7280–7289, Sep. 2020.

[15] A. Habib, H. K. Singh, T. Chugh, T. Ray, and K. Miettinen, “A multiple surrogate assisted decomposition-based evolutionary algorithm for expensive multi/many-objective optimization,” IEEE Transactions on Evolutionary Computation, vol. 23, no. 6, pp. 1000–1014, Dec 2019.

[16] C. Juang and T. B. Bui, “Reinforcement neural fuzzy surrogate-assisted multiobjective evolutionary fuzzy systems with robot learning control application,” IEEE Transactions on Fuzzy Systems, vol. 28, no. 3, pp. 434–446, March 2020.

[17] S. Deng, R. El Bechari, S. Brisset, and S. Cle net, “Iterative kriging- based methods for expensive black-box models,” IEEE Transactions on Magnetics, vol. 54, no. 3, pp. 1–4, March 2018.

[18] J. Yin, S. Ng, and K. Ng, “Kriging metamodel with modified nugget- effect: The heteroscedastic variance case,” Computers and Industrial Engineering, vol. 61, pp. 760–777, 10 2011.

[19] H. Wang, Y. Jin, and J. Doherty, “Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems,” IEEE Transactions on Cybernetics, vol. 47, no. 9, pp. 2664–2677, Sep. 2017.

[20] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya, “A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization,” IEEE Trans- actions on Evolutionary Computation, vol. 22, no. 1, pp. 129–142, Feb 2018.

[21] M. T. M. Emmerich, K. C. Giannakoglou, and B. Naujoks, “Single- and multiobjective evolutionary optimization assisted by gaussian random field metamodels,” IEEE Transactions on Evolutionary Computation, vol. 10, no. 4, pp. 421–439, Aug 2006.

[22] C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng, “Surrogate-assisted co- operative swarm optimization of high-dimensional expensive problems,” IEEE Transactions on Evolutionary Computation, vol. 21, no. 4, pp. 644–660, Aug 2017.

[23] B. Liu, Q. Zhang, and G. G. E. Gielen, “A gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive opti- mization problems,” IEEE Transactions on Evolutionary Computation, vol. 18, no. 2, pp. 180–192, April 2014.

[24] H. Yu, C. Y. Chung, K. P. Wong, H. W. Lee, and J. H. Zhang, “Probabilistic load flow evaluation with hybrid latin hypercube sampling and cholesky decomposition,” IEEE Transactions on Power Systems, vol. 24, no. 2, pp. 661–667, May 2009.

[25] R. Storn and K. Price, “De-a simple and efficient heuristic for global optimization over continuous space,” Joumal of Global Optimization, vol. 114, pp. 341–359, 01 1997.

[26] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proceed- ings of ICNN’95 - International Conference on Neural Networks, vol. 4, Nov 1995, pp. 1942–1948 vol.4.

[27] V. Mohan, J. G. Singh, and W. Ongsakul, “An efficient two stage stochastic optimal energy and reserve management in a microgrid,” Applied Energy, vol. 160, pp. 28–38, 12 2015.

[28] N. Awad, M. Ali, R. Mallipeddi, and P. Suganthan, “An improved differential evolution algorithm using efficient adapted surrogate model for numerical optimization,” Information Sciences, vol. 451, 04 2018.

[29] J. Zhang and A. C. Sanderson, “Jade: Adaptive differential evolution with optional external archive,” IEEE Transactions on Evolutionary Computation, vol. 13, no. 5, pp. 945–958, Oct 2009.

[30] R. Mallipeddi and M. Lee, “An evolving surrogate model-based differen- tial evolution algorithm,” Applied Soft Computing, vol. 34, pp. 770–787, 09 2015.

[31] R. Tanabe and A. S. Fukunaga, “Improving the search performance of shade using linear population size reduction,” in 2014 IEEE Congress on Evolutionary Computation (CEC), July 2014, pp. 1658–1665.

[32] S. Elsayed, R. Sarker, C. Coello, and T. Ray, “Adaptation of oper- ators and continuous control parameters in differential evolution for constrained optimization,” Soft Computing, 07 2017.

[33] Y. Wang, D. Yin, S. Yang, and G. Sun, “Global and local surrogate- assisted differential evolution for expensive constrained optimization problems with inequality constraints,” IEEE Transactions on Cybernet- ics, vol. 49, no. 5, pp. 1642–1656, May 2019. [34] A. Maesani, G. Iacca, and D. Floreano, “Memeticviabilityevolutionfor constrained optimization,” IEEE Transactions on Evolutionary Compu- tation, vol. 20, no. 1, pp. 125–144, Feb 2016.

[35] Y. Wang, B. Wang, H. Li, and G. G. Yen, “Incorporating objective function information into the feasibility rule for constrained evolutionary optimization,” IEEE Transactions on Cybernetics, vol. 46, no. 12, pp. 2938–2952, Dec 2016.

Figures

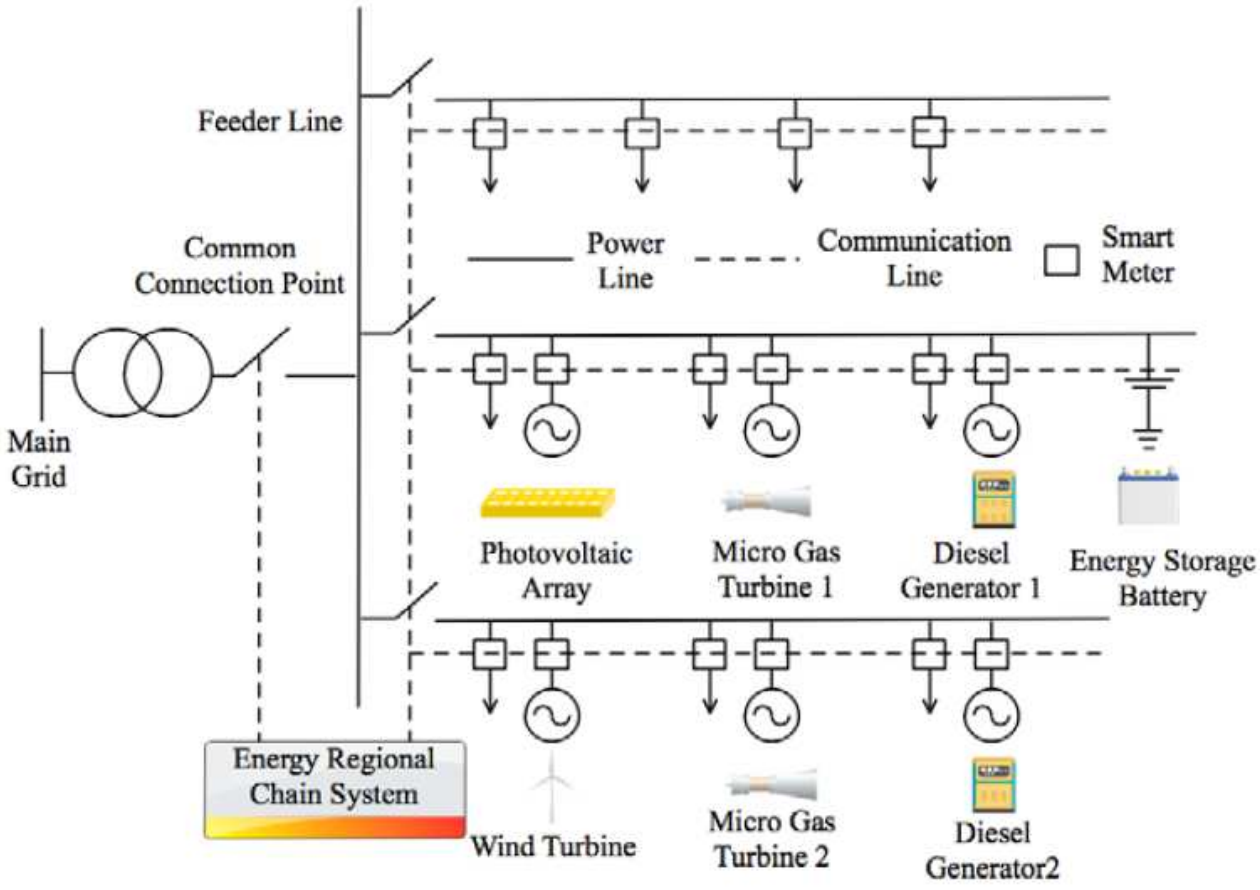

Figure 1

Typical micro-grid structure diagram.

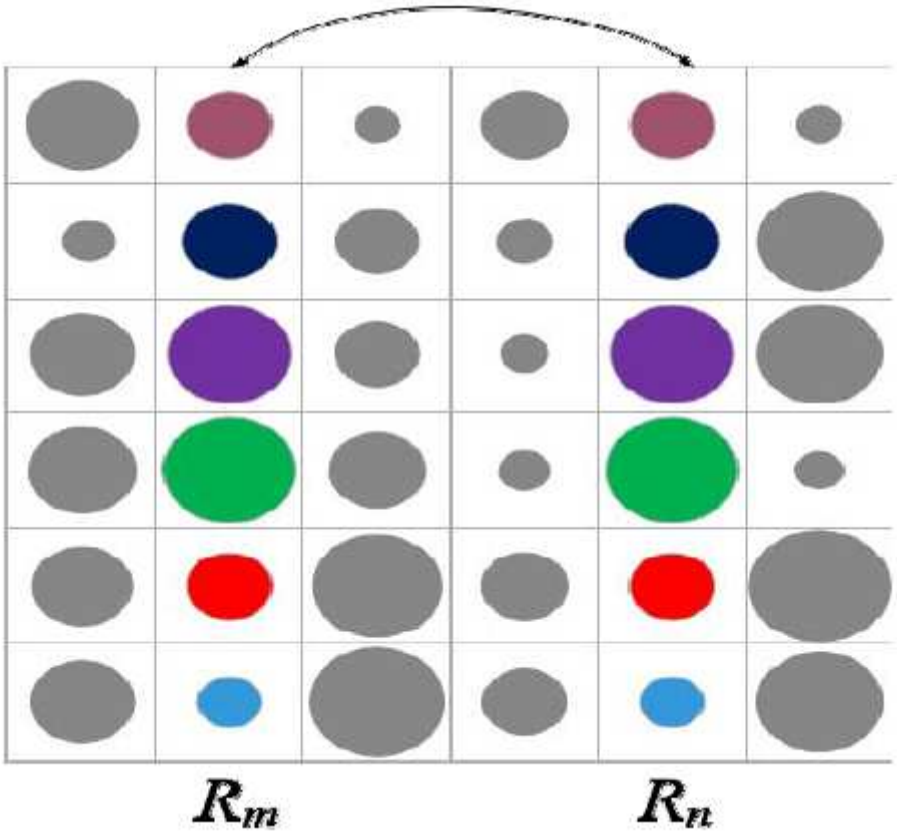

Figure 2

Correlation function matrix R (Rm and Rn columns are approximately equal).

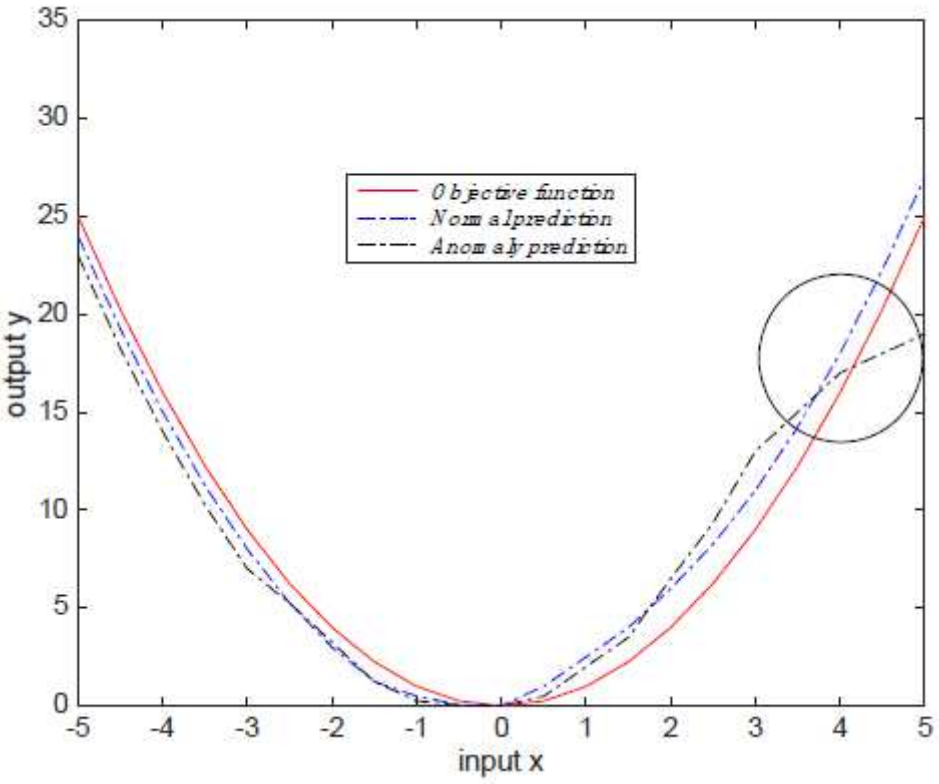

Figure 3

Normal prediction and anomaly prediction for the test function.

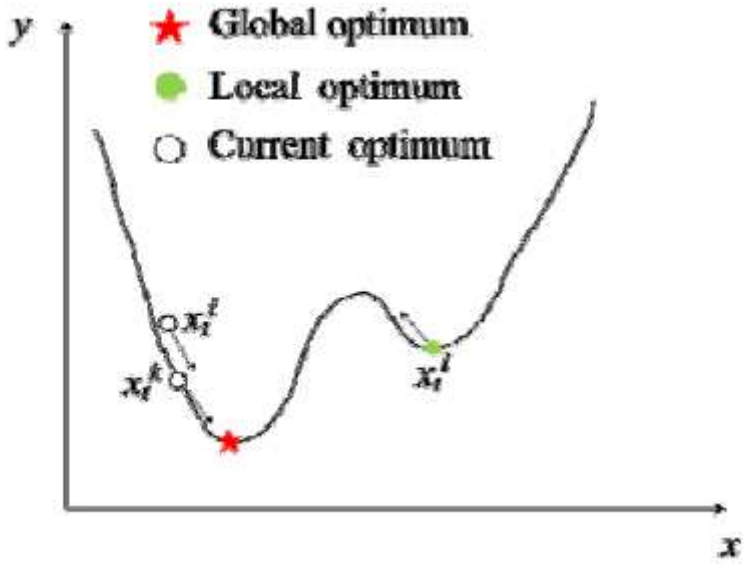

Figure 4

Speed up the convergence.

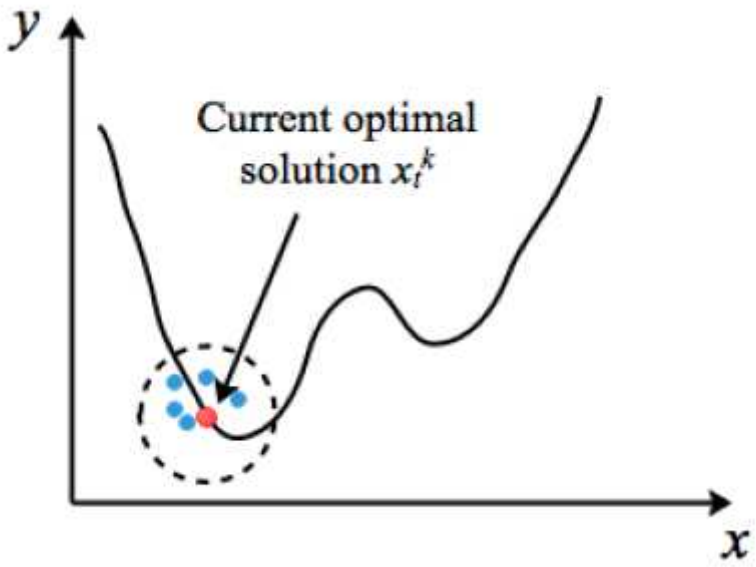

Figure 5

Schematic diagram of ImKPSO local search mechanism.

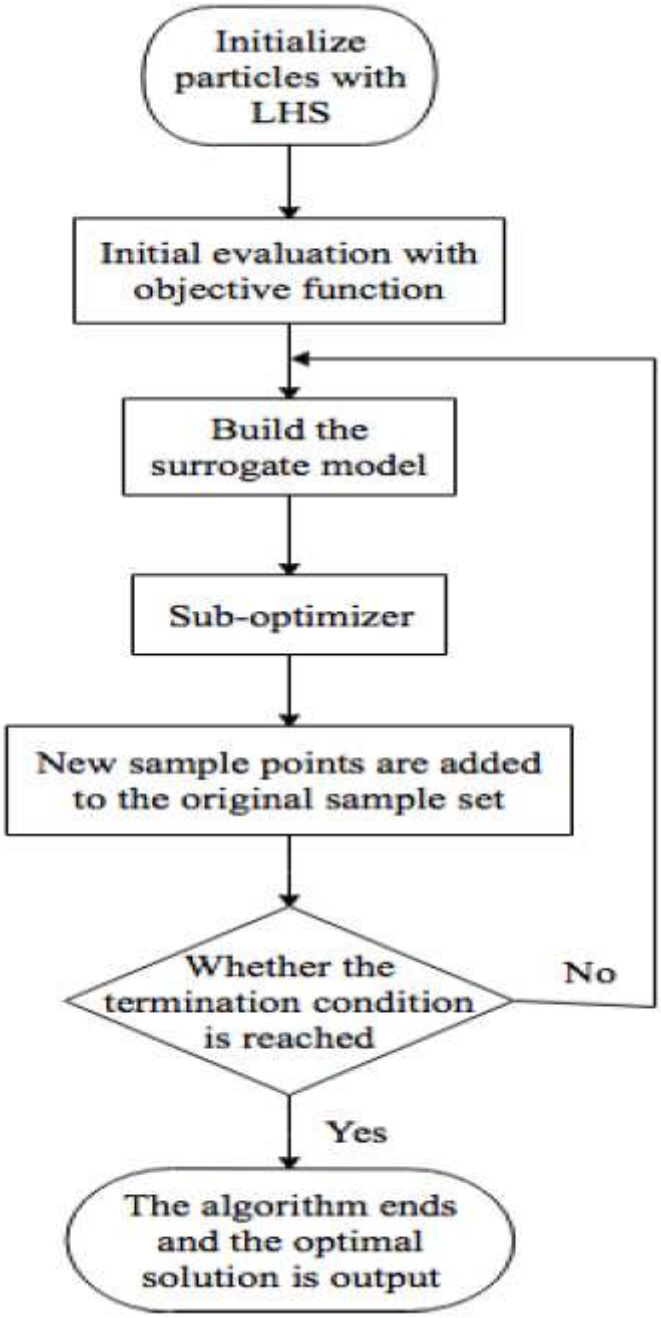

Figure 6

ImKPSO algorithm flowchart.

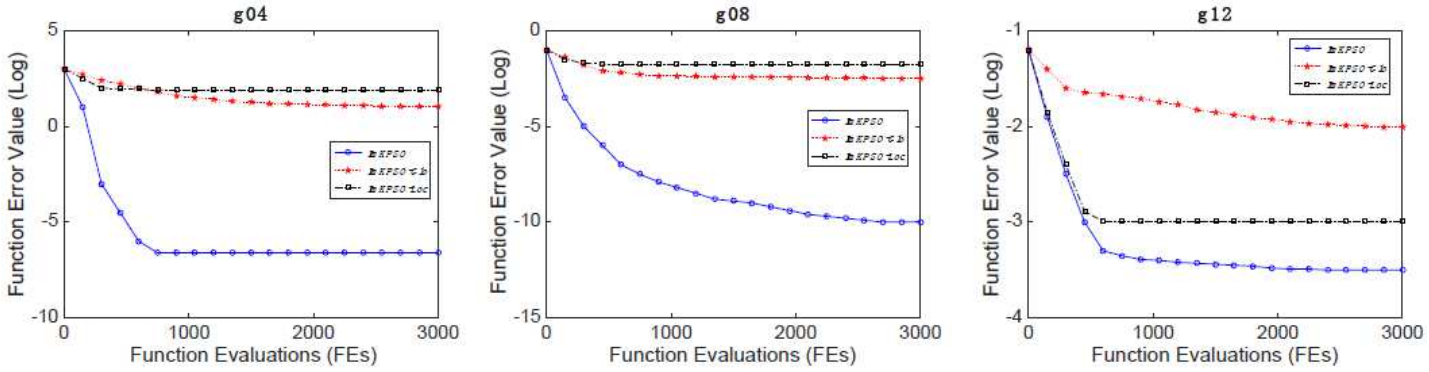

Figure 7

Convergence graphs of three algorithms on three functions of CEC2006.

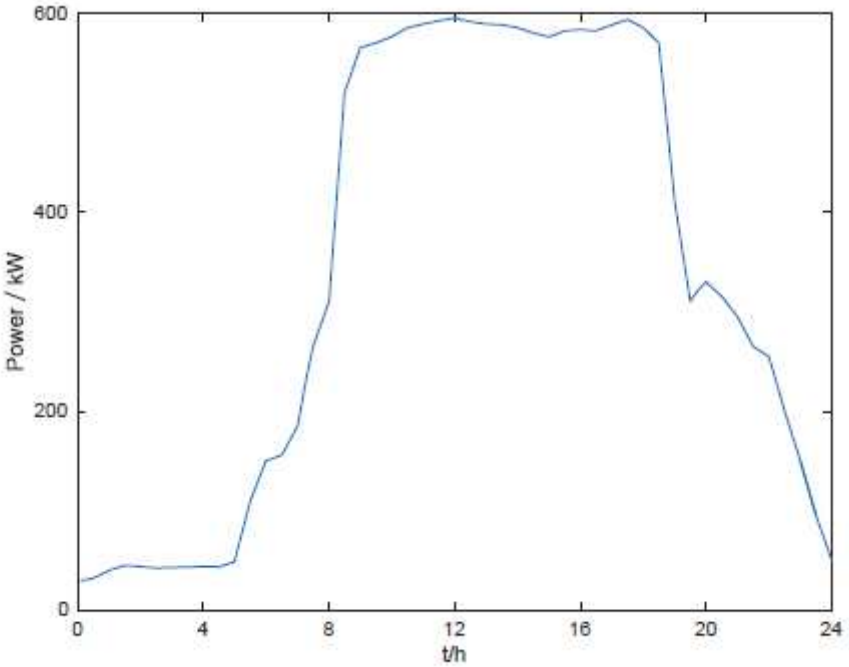

Figure 8

Power load demand curve

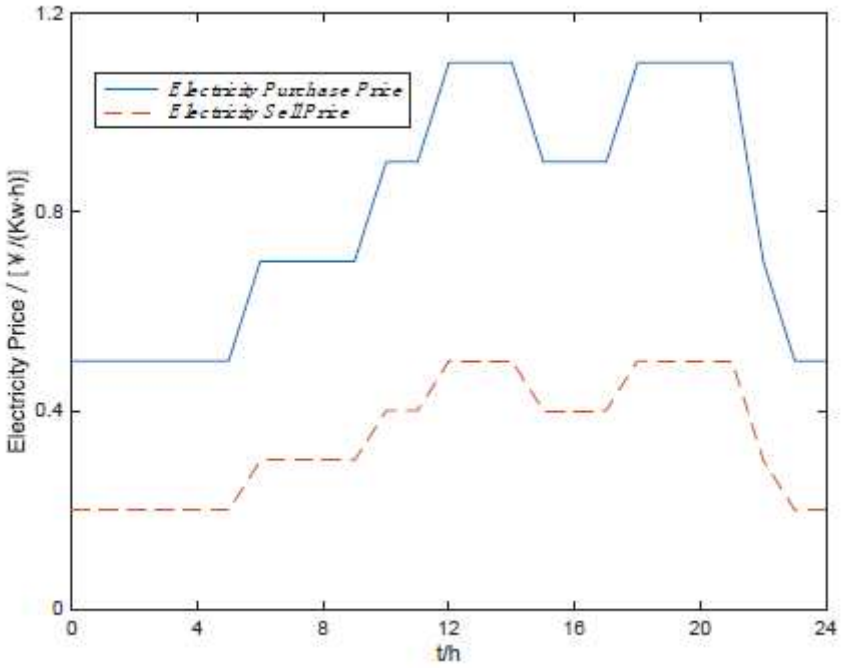

Figure 9

Electricity market price curve

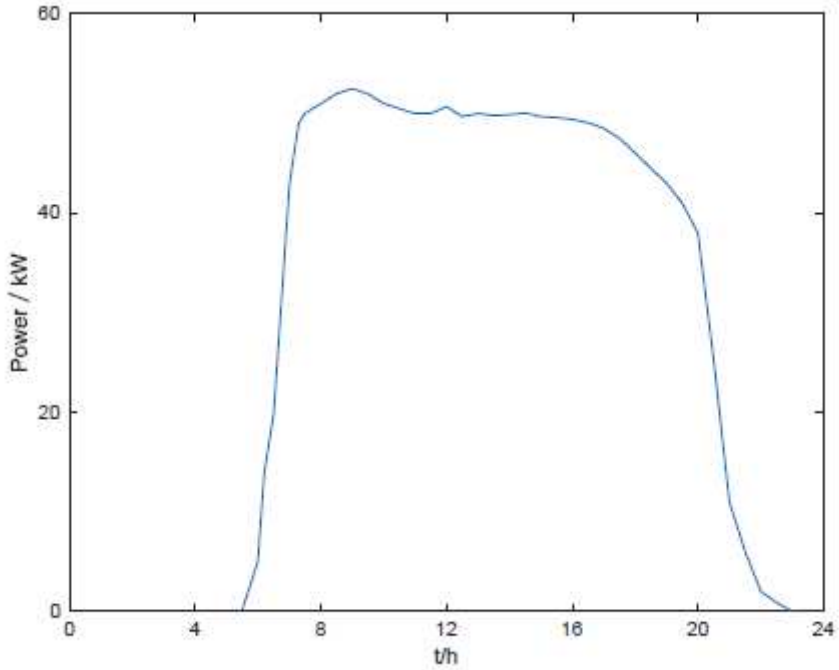

Figure 10

Photovoltaic array power curve

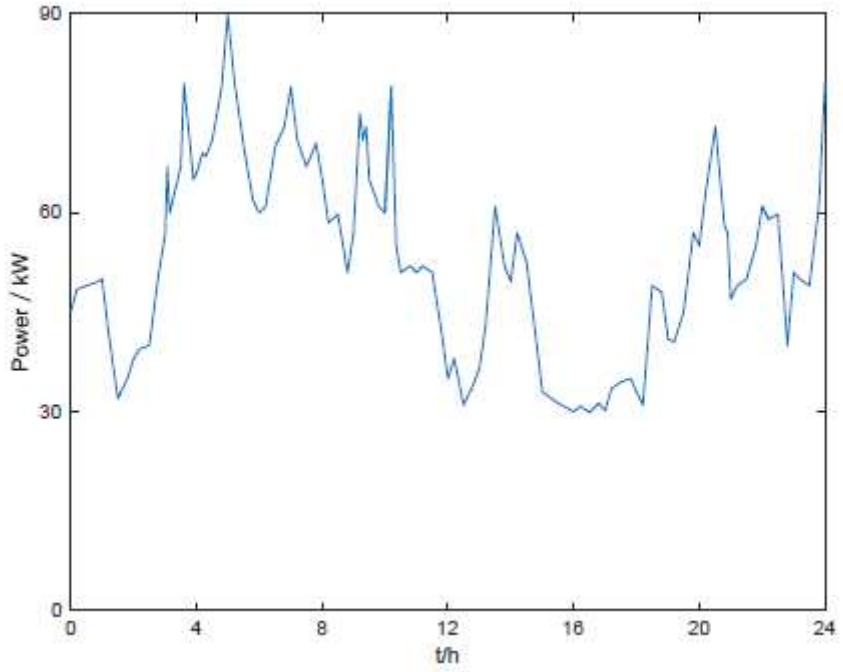

Figure 11

Wind turbine power curve

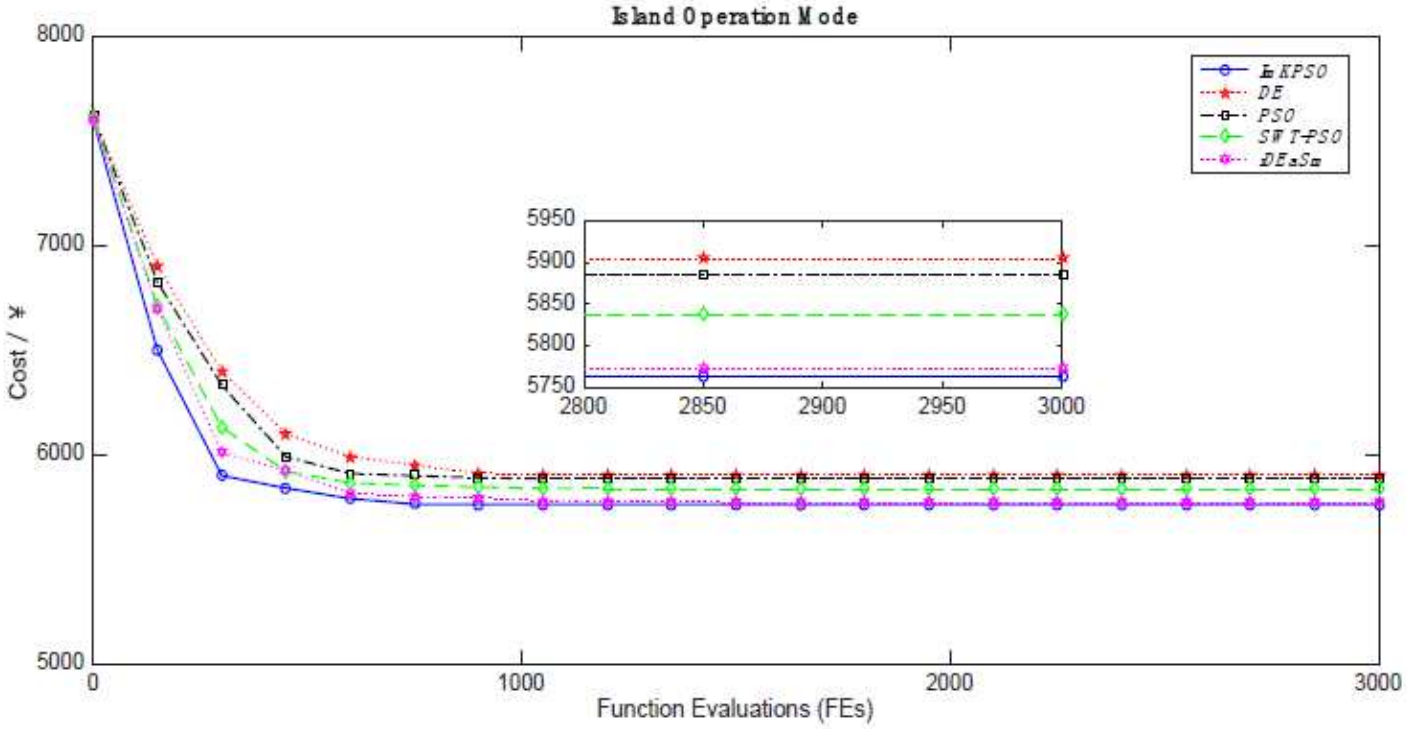

Figure 12

Convergence curve of each algorithm in island operation mode

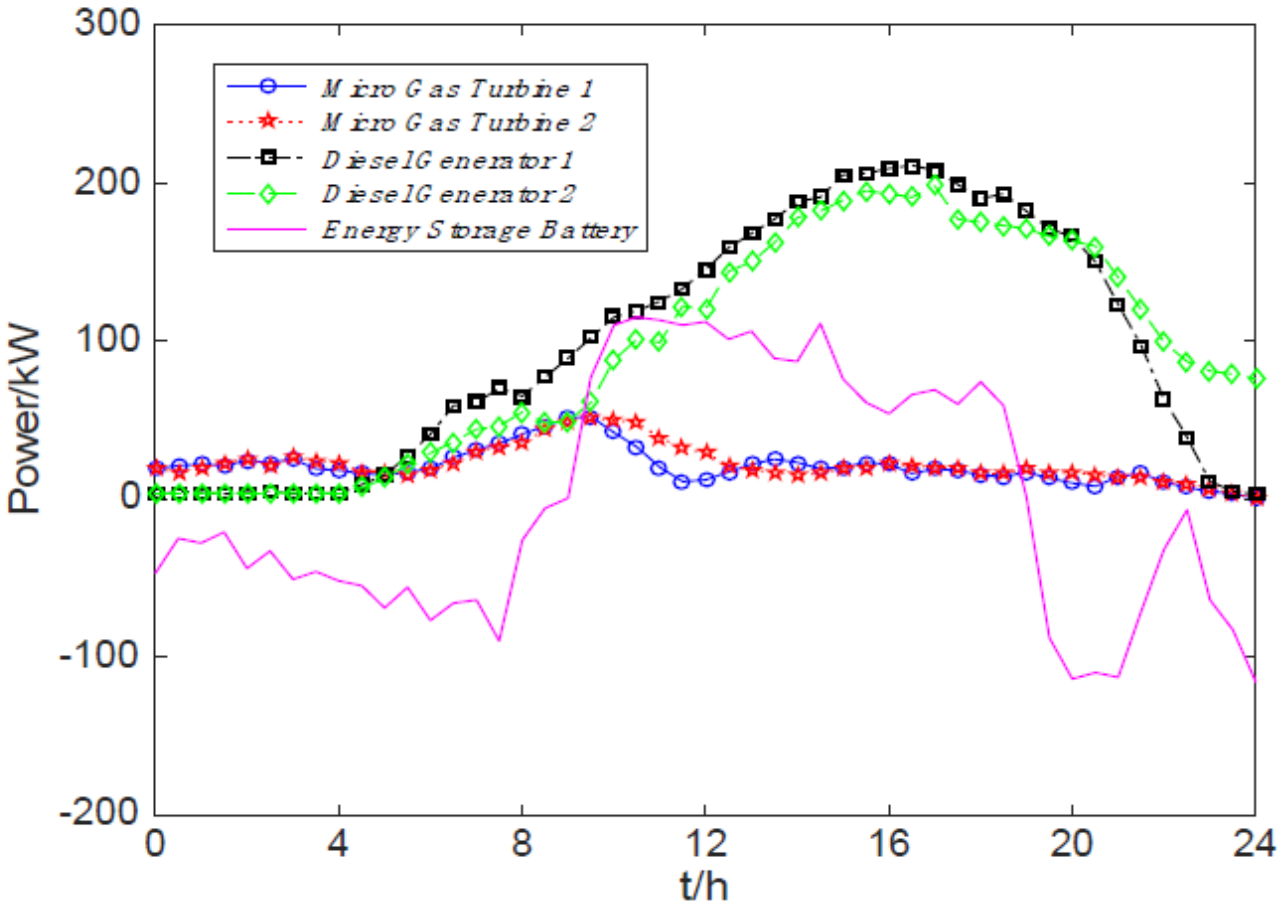

Figure 13

System output optimization results in island mode

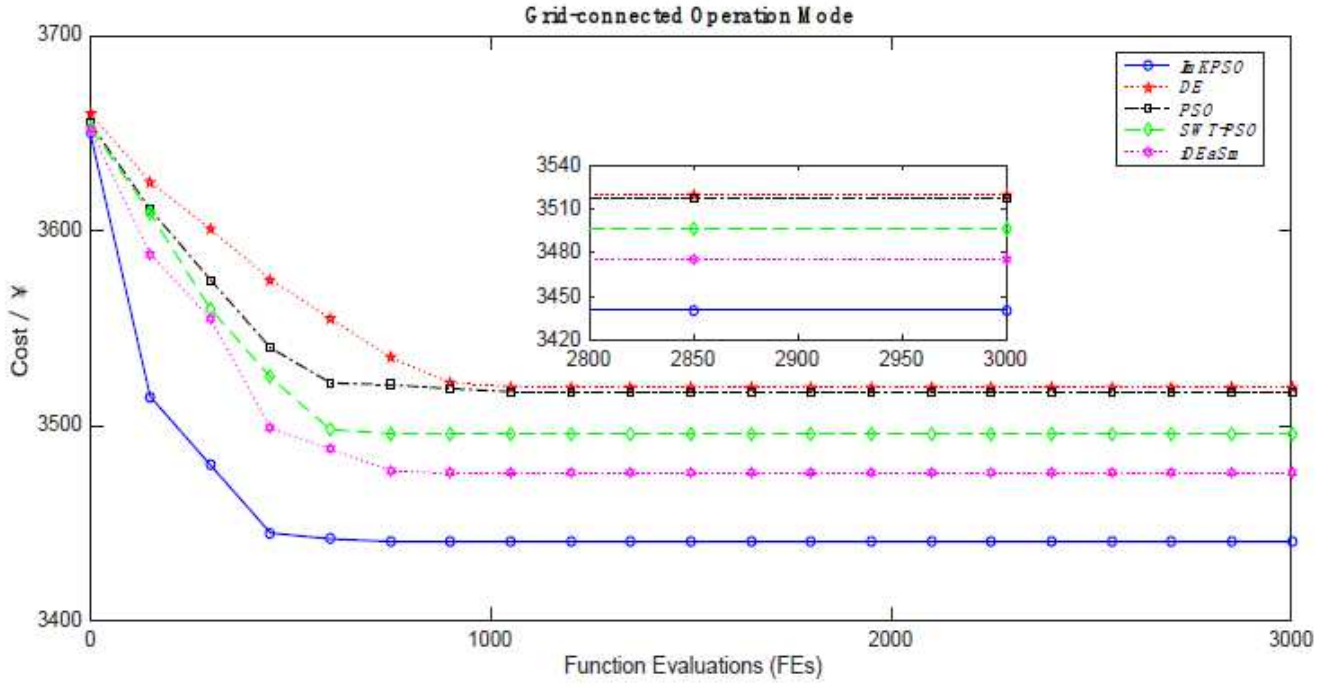

Figure 14

Convergence curve of each algorithm in grid connected operation mode

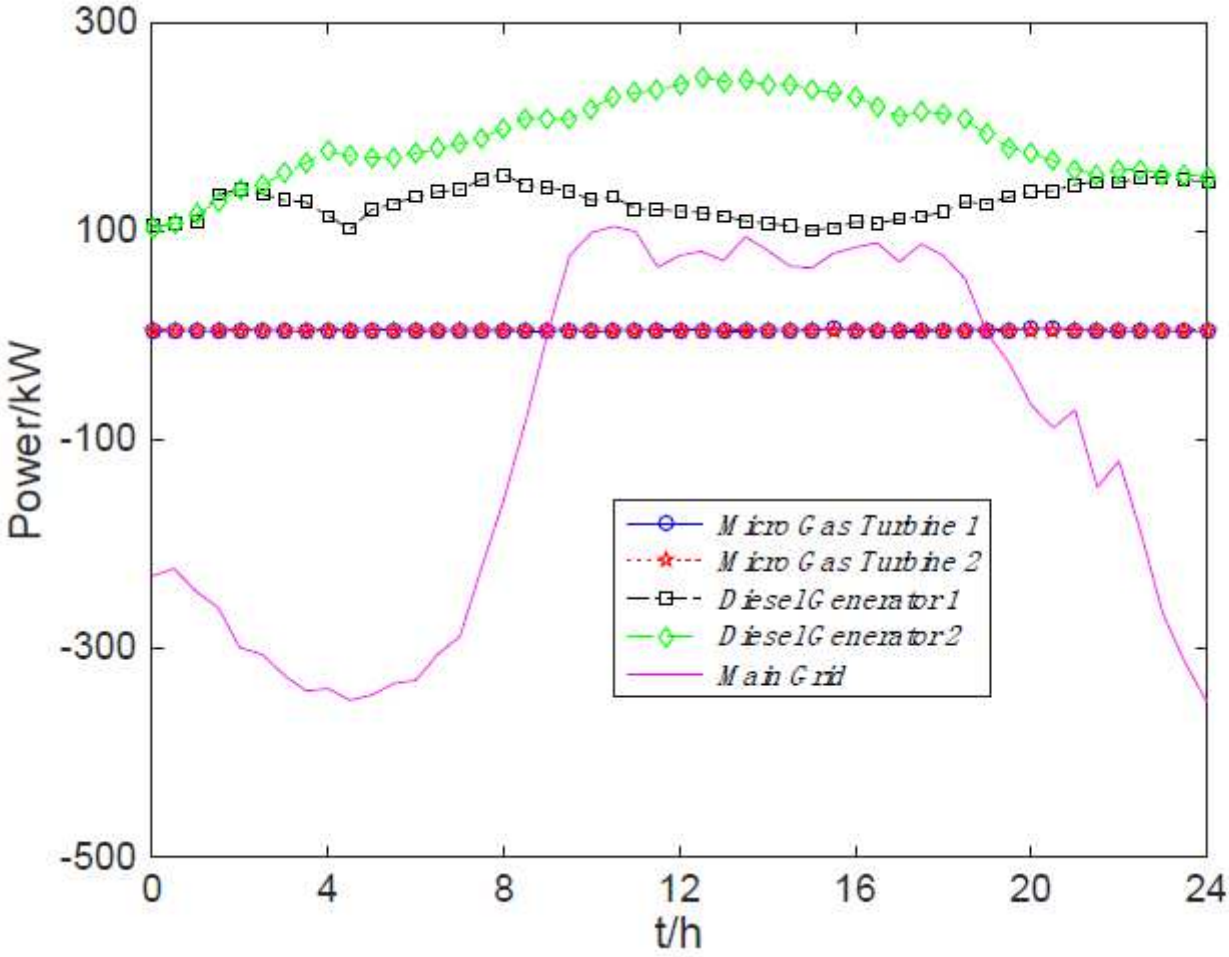

Figure 15

System output optimization results in grid connected operation mode