Parallel and Distributed Optimization Method With Constraint Decomposition for Energy Management of Microgrids

Qiang Li, *Member, IEEE*, Yuexi Liao, Kunming Wu, Leiqi Zhang, Jiayang Lin, Minyou Chen, *Senior Member, IEEE*, Josep M. Guerrero, *Fellow, IEEE*, and Derek Abbott, *Fellow, IEEE*

***Abstract*—Energy management in power systems is a thorny optimization problem. With the sizes of systems rising, centralized optimization methods are restricted by their complexities of communications, while distributed optimization methods have emerged as a powerful tool for dealing with increasingly complex systems. However, convergence rates of some widely used distributed optimization methods, such as the standard alternating direction method of multipliers (ADMM), still have room for improvement. In this paper, a parallel and distributed optimization method for energy management of microgrids (MGs) is proposed to boost the convergence rate without sacrificing the accuracy of the optima, in which agents calculate, exchange and update in parallel. At first, a decomposition method is presented, where the objective functions and constraints of an original optimization problem with separable variables are decomposed into local objective functions and constraints for agents, which is the key to our method. Further, agents solve their local optimization problems independently and then exchange determined optima with their neighbors. Finally, the method is evaluated to solve economic dispatch with demand response for microgrids. The simulation results show that compared to the standard ADMM, for a given accuracy, the number of iterations in our method is only one third or even less than that of ADMM. Furthermore, our method can minimize the cost functions of distributed generation on supply side and maximize the profit functions of flexible loads on the demand side.**

***Index Terms*—Microgrids, energy management, demand response, distributed optimization, multi-agent systems.**

Manuscript received December 25, 2020; revised May 6, 2021; accepted July 6, 2021. Date of publication July 14, 2021; date of current version October 21, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61105125 and Grant 51177177; in part by the National "111" Project of China under Grant B08036; and in part by the Science and Technology Project of State Grid Zhejiang Electric Power Company of China under Grant 5211DS19002F. Paper no. TSG-01911-2020. *(Corresponding author: Qiang Li.)*

Qiang Li, Yuexi Liao, Kunming Wu, and Minyou Chen are with the State Key Laboratory of Power Transmission Equipment and System Security and New Technology, School of Electrical Engineering, Chongqing University, Chongqing 400044, China (e-mail: qiangli9@outlook.com).

Leiqi Zhang is with the Zhejiang Key Laboratory of Distributed Generations and Microgrid Technology, State Grid Zhejiang Electric Power Research Institute, Hangzhou 310014, China.

Jiayang Lin is with the Department of Production Technology, State Grid Wenzhou Power Supply Company, Wenzhou 330302, China.

Josep M. Guerrero is with the Department of Energy Technology, Aalborg University, 9220 Aalborg, Denmark (e-mail: joz@et.aau.dk).

Derek Abbott is with the School of Electrical and Electronic Engineering, University of Adelaide, Adelaide, SA 5005, Australia (e-mail: derek.abbott@adelaide.edu.au).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2021.3097047.

Digital Object Identifier 10.1109/TSG.2021.3097047

I. Introduction

THE RAPID development in world population and economic growth poses a significant challenge in terms of both energy demand and protection of the environment [1]. Thus an increase number of countries are progressively turning to the utilization of renewable energies, such as wind and solar, which shifts the paradigm towards distributed generation (DG). However, the intermittent outputs of DGs may disturb the performance of the main grid, if DGs connect to the main grid directly. To reduce the disturbance, a microgrid (MG) technology is developed, which integrates DGs, loads, energy storage systems (ESS), and monitoring and protection devices as a system, where advanced inverters and converters are widely used [2], [3]. As a controllable system, an AC or DC MG can connect to the main grid by the point of common coupling (PCC) or run independently, i.e., grid-connected or islanded mode [4], [5]. In the islanded mode, the control and economic dispatch of MGs become more difficult without the support of the main grid.

In recent years, centralized optimization methods have been widely employed to solve the economic operation and optimal dispatch of MGs, where dynamic programming (DP) [6], [7], genetic algorithm (GA) [8], [9], particle swarm optimization (PSO) [10], [11], mixed integer linear programming (MILP) [12] and second-order cone relaxation (SOCR) [13] have demonstrated reasonable performance in terms of economic dispatch of MGs. For example, advanced DP was applied to improve the energy utilization efficiency and decrease gas emissions [6]. To reduce the operating costs of community MGs with ESS, a modified PSO was proposed to find the optimal strategies for battery management with the change of real-time electricity rates [11]. Additionally, new methods are also developed. To solve economic dispatch problems with numerous constraints and decision criteria, an improved GA and an enhanced MILP were combined [14]. If the long-term operating costs of grid-connected MGs are considered, a robust two-stage optimization approach to schedule the power generation and ESS, and manage energy trading with the main grid under uncertainties can be employed [15]. For more general convex cost functions in economic dispatch problems, a secant approximation method was proposed to do the one-dimensional search and solve economic dispatch in power systems with high penetration of renewable energies [16].

More recently, prosumers have the opportunity to take part in economic dispatch via use of demand response (DR) technology, where there are two typical programs, (i) incentive-based programs (IBP) and (ii) price-based programs (PBP) [17]. Moreover, IBP has two forms: (i) direct load control (DLC) [18] and (ii) interruptible load programs (ILP) [18], [19], e.g., ILP can be realized by MILP, which utilizes both spinning reserve and interruptible loads as the operating reserve [19]. In [18], DLC and ILP are combined to provide instantaneous reserves for ancillary services, while instead of scheduling loads on the demand-side directly, PBP influences the power consumption behaviors of consumers by changing electricity rates [20]–[22]. To better deal with uncertainties in the system, Tushar *et al.* proposed a real-time decentralized method for demand-side management, where flexible loads were regulated real-time to follow a day-ahead plan [23]. For MGs with large numbers of electric vehicles (EVs), a multi-objective optimization model for load dispatch was proposed to minimize the operating costs, when stochastic access of EVs was considered [24]. In fact, MGs were also established for buildings or small districts, to minimize the costs to consumers in buildings by employing an energy management framework to especially deal with electricity and heat demand response [25].

Centralized optimization methods for economic operation and dispatch can more readily find optima due to having global information, when compared to distributed optimization methods. Nevertheless, it increases the communication complexity due to collecting information from all DGs and loads. And it is worse that it adds a heavy burden to the central controller due to processing large scale optimization problems, which may cause failure of the central controller. Therefore, research methods for energy management are shifting from centralized methods to distributed or decentralized methods [26]–[29]. The alternating direction method of multipliers (ADMM) [30], as a distributed optimization method, has been widely used for economic dispatch problems. To decrease the operating costs of MGs, ADMM was used to coordinate central controller and local controllers [31], and it was also extended to multi-time scale optimization of MG clusters to eliminate the influence of uncertainties [32]. Moreover, a distributed algorithm based on ADMM with convergence assurance was proposed to minimize the overall energy costs among multiple MGs clusters [26]. Alternatively, another widely applied distributed method is the consensus-based distributed algorithm [33]. For example, a consensus-based algorithm was presented to achieve optimal economic dispatch, when delay effects were considered [34]. Moreover, economic dispatch with DR has also been studied [35]. Further, the average consensus algorithm was also combined with ADMM, which was applied to solve dynamic economic dispatch problems [36].

In summary, the economic dispatch problems of minimizing the costs can be solved by the widely-used consensus algorithm in a distributed manner, when incremental costs reach a consensus. However, for more general optimization problems, it is not always true to find a variable for consensus. In this case, the consensus algorithm does not work, so we have to turn to the alternating direction method of multipliers (ADMM), a widely-used distributed optimization method. Unfortunately, the standard ADMM can solve more general optimization problems, but it works in cascade, which means that on a ring network composed of agents the next agent starts to calculate and update, after it receives the information from the previous agent. For a complex optimization problem with many variables, this serial process may cause ADMM needs more number of iterations to find optima with the rise of sizes of optimization problems. If a distributed optimization method works in parallel, maybe it will converge faster, because information can be exchanged and processed among more neighbor agents instead of one by one and agents can search their subspaces simultaneously (in parallel), which accelerates the process of finding optima. Therefore, in this paper, to increase the convergence rate, a parallel and distributed optimization method (PDOM) has been proposed, and then this method is applied to the optimization of economic dispatch with demand response in MGs.

There are three main technical contributions. *First*, the decomposition method for objective functions and constraints of an optimization problem with separable variables is presented. After the decomposition, local objective functions and local constraints for agents are formed, which is the necessary preparation for the parallel and distributed optimization method. *Second*, the parallel and distributed optimization method with decomposition of objective functions and constraints is proposed, where agents calculate, exchange and update in parallel. *Third*, two propositions and a corollary are proved, which guarantee the sum of local constraints are consistent with the original constraints, and the optima can be iteratively obtained. Compared to the standard ADMM, our method can find better optima and more importantly the number of iterations in our method is only one third or even less than that of ADMM.

The rest of the paper is organized as follows. Section II introduces the preliminaries of Graph Theory and then the parallel and distributed optimization method with decomposition of objective functions and constraints is proposed. In Section III, an optimization model of economic dispatch with demand response is presented, where supply-side and demand side cooperate with each others. An MG model for tests are built in Section IV and the parameters of DGs and loads are listed. In Section V, cases are designed to compare the accuracy and rate of convergence of our method and other typical methods. We evaluate the performance of the MG, when our method is applied to the optimization of economic dispatch with demand response.

II. Parallel and Distributed Optimization Method (PDOM)

In this section, some terms to describe graphs or networks composed of agents are introduced first. And then local areas on the network of agents are given, which are formed only by agents and their neighbors, and each of which corresponds to a local objective function and local constraints. Next, the method to derive local objective functions and local constraints is presented, and then a parallel and distributed optimization

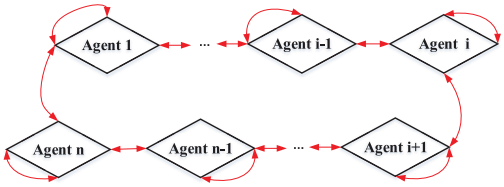


Fig. 1. A ring network composed of agents.

method is proposed. Finally, the convergence of the method is analyzed, where two propositions and a corollary are proved.

A. Topology of Network Composed of Agents

In this paper, the distributed optimization is carried out by agents cooperatively. Therefore, a network composed of agents has to be formed first, where a network can be built by adding links among agents and no isolated agents are permitted on the network, such as that shown in Fig. 1. On the network, agents can send or collect information, if there is a link between them. Note that a self-loop is added to collect and process its own information.

In terms of Graph Theory, the formed network is a bidirectional graph $G(V, E)$, where V is a set of nodes (agents), and E is a set of edges or links. To describe the relationships among agents, an adjacency matrix $A = [a_{ij}]_{n\times n}$ is defined. If there is an outgoing link from agents i to j, then $a_{ij} = 1$. Otherwise, $a_{ij} = 0$. Noting that the adjacency matrix of the network is a symmetric matrix satisfying $a_{ij} = a_{ji}$, because of the property of a bidirectional graph. Moreover, all diagonal entries of the adjacency matrix are one, since agents have self-loops. Further, a diagonal matrix D is defined to indicate the outdegrees of agents, where the outdegree d_{ii} of an agent i denotes the number of outgoing links of the agent, namely,

$$d_{ii} = \sum_{j=1}^{n} a_{ij}. \tag{1}$$

Moreover, the outdegree is at least equal to two, because an agent has a self-loop and at least an outgoing link. Further, a weighted matrix W can be defined as follows,

$$W = A \cdot D^{-1}, \tag{2}$$

and it has the property

$$\begin{aligned}\sum_{i=1}^{n} w_{ij} &= \frac{a_{1j} + \cdots + a_{ij} + \cdots + a_{nj}}{d_{jj}} \\ &= \frac{a_{j1} + \cdots + a_{ji} + \cdots + a_{jn}}{d_{jj}} = 1,\end{aligned} \tag{3}$$

which means the sum of elements of a column of the weighted matrix W is one. Note that a network with similar degrees, such as a nearest neighbor network, is recommended, but avoiding using a scale-free network whose degree follows the power-law degree distribution. In this type of networks, there exists some nodes with huge degrees who look like central controllers, e.g., collecting a lot of information and having heave burden, which violates the original intention of distributed optimization.

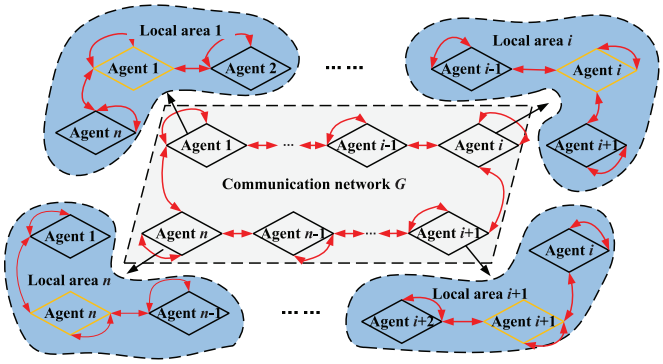


Fig. 2. Local areas for agents on a communication network $G(V, E)$.

On the network G, a local area of an agent i is defined as the subgraph of G induced by all agents adjacent to the agent i, i.e., it consists of the agent i itself (indicated by a yellow diamond) and its neighbors (indicated by black diamonds) connecting to the agent i, as shown in Fig. 2. In a local area, there are a local objective function and local constraints derived from the original optimization problem. So, agents solve the local objective functions with local constraints simultaneously, and then the obtained optima are exchanged among agents. Finally, the optima obtained over local areas will iteratively approach to the optima of the original optimization problem in a parallel and distributed manner. This is the idea of the parallel and distributed optimization method, and it will be developed in the successive sections.

B. Decomposition of Objective Functions for Local Areas

Assume the original optimization problem is a convex optimization problem with an equality constraint (an often used paradigm for optimization problems [30], [37]), which can be described as

$$\min \sum_{i=1}^{n} f_i(x_i), \tag{4a}$$

$$\text{s.t.} \sum_{i=1}^{n} c_i x_i = b, \tag{4b}$$

where $X = [x_1, x_2, \ldots, x_n] \in R^{n\times 1}$ is the vector for minimizing the objective function, namely the optima, $f_i(x_i)$ is a convex function, and $C = [c_1, c_2, \ldots, c_n] \in R^{1\times n}$ contains the coefficients of constraints, $b \in R$ is the value of constraints.

If the original optimization problem needs to be solved in a distributed manner, such as solved by ADMM, a network composed of n agents will be formed first according to the previous section, where an agent i handles a function $f_i(x_i)$. However, the distributed optimization by ADMM is carried out in a serial way, which slows down the convergence rate of the method. Therefore, in this paper, a parallel and distributed optimization method is proposed to reduce the number of iterations. Accordingly, the above mentioned objective function and the constraint will have to be decomposed into n local objective functions and n local constraints for n local areas, respectively. In this section, the decomposition method for

objective functions is introduced, while the decomposition of constraints is given in the next section.

For a local area i, it only contains the agent i and its neighboring agents, so the local objective function is made up of functions of the agent i and its neighboring agents. Thus, the local objective function for the local area i in Fig. 2 takes form as

$$\begin{aligned}\min F_i(X_i) &= w_{i1}f_1(x_1) + \cdots + w_{i,i-1}\cdot f_{i-1}(x_{i-1}) + w_{ii}f_i(x_i) \\ &\quad + w_{i,i+1}\cdot f_{i+1}(x_{i+1}) + \cdots + w_{in}f_n(x_n) \\ &= w_{i,i-1}f_{i-1}(x_{i-1}) + w_{ii}f_i(x_i) + w_{i,i+1}f_{i+1}(x_{i+1})\end{aligned} \tag{5}$$

where $F_i(X_i)$ is the local objective function of the local area i and $X_i = [a_{i1}x_1, a_{i2}x_2, \ldots, a_{in}x_n]$. In terms of the weighted matrix W, we can derive the local objective functions for all local areas as

$$\begin{bmatrix} F_1(X_1) \\ \vdots \\ F_i(X_i) \\ \vdots \\ F_n(X_n) \end{bmatrix} = \begin{bmatrix} w_{11} & \cdots & w_{1i} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ w_{i1} & \cdots & w_{ii} & \cdots & w_{in} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{n1} & \cdots & w_{ni} & \cdots & w_{nn} \end{bmatrix} \cdot \begin{bmatrix} f_1(x_1) \\ \vdots \\ f_i(x_i) \\ \vdots \\ f_n(x_n) \end{bmatrix}. \tag{6}$$

Observing Eq. (6), it can be seen that there are n local objective functions for n local areas. If one of these local objective functions, e.g., $F_i(X_i)$, is solved, we will find a vector of optima of X, which means that there are n vectors of optima of X for all n local areas. To indicate the vector of optima of X found by a local area, we define a matrix Z as follows,

$$Z(k) = \begin{bmatrix} a_{11}z_{11}(k) & \cdots & a_{1i}z_{1i}(k) & \cdots & a_{1n}z_{1n}(k) \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{i1}z_{i1}(k) & \cdots & a_{ii}z_{ii}(k) & \cdots & a_{in}z_{in}(k) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1}z_{n1}(k) & \cdots & a_{ni}z_{ni}(k) & \cdots & a_{nn}z_{nn}(k) \end{bmatrix}. \tag{7}$$

For example, the ith row of Z represents the vector of optima found on the local area i by the kth iteration. Therefore, the relationship between X and Z at the kth iteration can be defined as

$$\begin{aligned} X &= \begin{bmatrix} x_1(k) \\ \vdots \\ x_i(k) \\ \vdots \\ x_n(k) \end{bmatrix} = \left[Z(k)\cdot D^{-1}\right]^T_{n\times n}\cdot \mathbf{1}_{n\times 1} \\ &= \begin{bmatrix} w_{11}z_{11}(k) & \cdots & w_{1i}z_{1i}(k) & \cdots & w_{1n}z_{1n}(k) \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ w_{i1}z_{i1}(k) & \cdots & w_{ii}z_{ii}(k) & \cdots & w_{in}z_{in}(k) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{n1}z_{n1}(k) & \cdots & w_{ni}z_{ni}(k) & \cdots & w_{nn}z_{nn}(k) \end{bmatrix}^T \cdot \mathbf{1}_{n\times 1}, \end{aligned} \tag{8}$$

where $\mathbf{1}_{n\times 1}$ is a column vector of ones and $(\cdot)^T$ denotes the transpose of a matrix. From Eq. (8), it can be seen that agents exchange obtained optima z_{ij} with their neighbors, and then find the average to determine $x_i(k)$ at the kth iteration. Thus, non-zero elements in a column of Z will iteratively converge to a certain final value, which also makes X converge.

C. Decomposition of Equality Constraints for Local Areas

This paper focuses on the optimization problem with an equality constraint, $\sum_{i=1}^{n} c_i x_i = b$. As is mentioned above, to develop the parallel and distributed optimization, decomposition of the equality constraint is necessary. Similarly to that performed in the previous section, i.e., in the local area i its local equality constraints are only associated with itself and its neighbors, so the decomposition of the equality constraints for the local area i (as shown in Fig. 2) on the kth iteration can be represented as

$$\begin{aligned} &w_{i1}c_1x_1 + \cdots + w_{i,i-1}c_{i-1}x_{i-1} + w_{ii}c_ix_i + \cdots \\ &\quad + w_{i,i+1}c_{i+1}x_{i+1} + \cdots + w_{in}c_nx_n \\ &= w_{i,i-1}c_{i-1}x_{i-1} + w_{ii}c_ix_i + w_{i,i+1}c_{i+1}x_{i+1} \\ &= b_i(k). \end{aligned} \tag{9}$$

Before calculating $b_i(k)$, the value of $c_ix_i(k-1)$ is transmitted to its neighboring agents $i-1$ and $i+1$, so do the neighbors. Therefore, $b_i(k)$ can be calculated as follows,

$$\begin{aligned} b_i(k) &= \begin{bmatrix} w_{i1} & \cdots & w_{ii} & \cdots & w_{in} \end{bmatrix} \cdot \begin{bmatrix} c_1x_1(k-1) \\ \vdots \\ c_ix_i(k-1) \\ \vdots \\ c_nx_n(k-1) \end{bmatrix} \\ &= w_{i,i-1}c_{i-1}x_{i-1}(k-1) + w_{ii}c_ix_i(k-1) \\ &\quad + w_{i,i+1}c_{i+1}x_{i+1}(k-1). \end{aligned} \tag{10}$$

Noting that $B(k) = [b_i(k)]_{n\times 1}$ is initialised as

$$B(0) = \begin{bmatrix} b_1(0) \\ \vdots \\ b_i(0) \\ \vdots \\ b_n(0) \end{bmatrix} = \begin{bmatrix} b \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{11}$$

which means that at the beginning only one agent knows the value of the constraint in Eq. (4b).

Therefore, for all local areas, the decomposition of constraints has the form as

$$\begin{aligned} &\begin{bmatrix} w_{11}c_1x_1 + \cdots + w_{1i}c_ix_i + \cdots + w_{1n}c_nx_n \\ \vdots \\ w_{i1}c_1x_1 + \cdots + w_{ii}c_ix_i + \cdots + w_{in}c_nx_n \\ \vdots \\ w_{n1}c_1x_1 + \cdots + w_{ni}c_ix_i + \cdots + w_{nn}c_nx_n \end{bmatrix} = W\cdot \begin{bmatrix} c_1x_1 \\ \vdots \\ c_ix_i \\ \vdots \\ c_nx_n \end{bmatrix} \\ &\qquad = \begin{bmatrix} b_1(k) \\ \vdots \\ b_i(k) \\ \vdots \\ b_n(k) \end{bmatrix}, \end{aligned} \tag{12}$$

Algorithm 1 Pseudocodes for Our Method (PDOM)

1: Find the adjacency matrix A, the degree matrix D and the weighted matrix W on a given network $G(V, E)$
2: Initialisation of $B(0)$ in terms of Eq. (11), r, and ϵ
3: Decompose the objective function and the constraint in Eq. (4) into local objective functions and constraints for agents in terms of Eqs. (6), (12) and (12)
4: Do{
5: Solve local objective functions with constraints and form the matrix Z in terms of Eq. (7)
6: Exchange $[z_{ij}]_{n\times n}$ with neighbors and then calculate X in terms of Eq. (8)
7: Exchange $[x_i]_{n\times 1}$ with neighbors and update local constraints in terms of Eq. (12)
8: Calculates r according to (15)
9: }While $r > \epsilon$
10: Return X.

$$B(k) = \begin{bmatrix} b_1(k) \\ \vdots \\ b_i(k) \\ \vdots \\ b_n(k) \end{bmatrix} = W \cdot \begin{bmatrix} c_1x_1(k-1) \\ \vdots \\ c_ix_i(k-1) \\ \vdots \\ c_nx_n(k-1) \end{bmatrix}, \tag{13}$$

and they satisfy the following two equations,

$$\mathbf{1}_{1\times n} \cdot \left(W \cdot \begin{bmatrix} c_1x_1 \\ \vdots \\ c_ix_i \\ \vdots \\ c_nx_n \end{bmatrix} \right) = \sum_{i=1}^{n} c_ix_i, \tag{14a}$$

$$\sum_{i=1}^{n} b_i(k) = \mathbf{1}_{1\times n} \cdot B(k) = b, \tag{14b}$$

where in fact Eqs. (14a) and (14b) are the left side and right side of Eq. (12), and it is worth noting that the weighted matrix W does not always have an inverse matrix. The proofs of Eqs. (14a) and (14b) are given in Proposition 1 in the next section.

Finally, the stopping criterion for our method is the presence of non-improving iterations, as is expressed as

$$r = \sum_{j=k-3}^{k-1} \sqrt{(x_i(k) - x_i(j))^2} < \epsilon. \tag{15}$$

So, the steps of our parallel and distributed optimization method are summarized in Algorithm 1.

D. Convergency Analysis

In terms of Eqs. (6) and (12), the objective functions and constraints of the original optimization problem Eq. (4) can be decomposed for all local areas. By exchanging information with neighbors simultaneously among agents on local areas, the optima of the original optimization problem can be found in a parallel and distributed manner. In this section, propositions are proven, which indicates convergency for this work. First, Proposition 1 is given as follows.

Proposition 1: Assume a network $G(V, E)$ composed of n agents is built for the optimization problem Eq. (4). If the constraints are decomposed by Eqs. (12) and (12), then Eqs. (14a) and (14b) will always hold.

Proof: First, we prove that Eq. (14a) holds, so we begin with the left side of Eq. (14a). Calculating the left side of Eq. (14a), we have

$$\begin{aligned} &\mathbf{1}_{1\times n} \cdot \left(W \cdot \begin{bmatrix} c_1x_1 \\ \vdots \\ c_ix_i \\ \vdots \\ c_nx_n \end{bmatrix} \right) \\ &= w_{11}c_1x_1 + \cdots + w_{1i}c_ix_i + \cdots + w_{1n}c_nx_n \\ &\quad + \cdots + w_{i1}c_1x_1 + \cdots + w_{ii}c_ix_i + \cdots + w_{in}c_nx_n \\ &\quad + \cdots + w_{n1}c_1x_1 + \cdots + w_{ni}c_ix_i + \cdots + w_{nn}c_nx_n \\ &= c_1x_1 \sum_{j=1}^{n} w_{j1} + \cdots + c_ix_i \sum_{j=1}^{n} w_{ji} + \cdots + c_nx_n \sum_{j=1}^{n} w_{jn}. \end{aligned} \tag{16}$$

Applying Eq. (3) to Eq. (16), it yields

$$\mathbf{1}_{1\times n} \cdot \left(W \cdot \begin{bmatrix} c_1x_1 \\ \vdots \\ c_ix_i \\ \vdots \\ c_nx_n \end{bmatrix} \right) = c_1x_1 + \cdots + c_ix_i + \cdots c_nx_n = \sum_{i=1}^{n} c_ix_i. \tag{17}$$

So, Eq. (14a) holds.

Next, we prove Eq. (14b) and calculate its left side as follows,

$$\begin{aligned} \sum_{i=1}^{n} b_i(k) &= \mathbf{1}_{1\times n} \cdot B(k) = \mathbf{1}_{1\times n} \cdot W \cdot \begin{bmatrix} c_1x_1(k-1) \\ \vdots \\ c_ix_i(k-1) \\ \vdots \\ c_nx_n(k-1) \end{bmatrix} \\ &= w_{11}c_1x_1(k-1) + \cdots + w_{1i}c_ix_i(k-1) + \cdots \\ &\quad + w_{1n}c_nx_n(k-1) + \cdots \\ &\quad + w_{i1}c_1x_1(k-1) + \cdots + w_{ii}c_ix_i(k-1) + \cdots \\ &\quad + w_{in}c_nx_n(k-1) + \cdots \\ &\quad + w_{n1}c_1x_1(k-1) + \cdots + w_{ni}c_ix_i(k-1) + \cdots \\ &\quad + w_{nn}c_nx_n(k-1) \\ &= c_1x_1(k-1) \sum_{j=1}^{n} w_{j1} + \cdots + c_ix_i(k-1) \sum_{j=1}^{n} w_{ji} + \cdots \\ &\quad + c_nx_n(k-1) \sum_{j=1}^{n} w_{jn}. \end{aligned} \tag{18}$$

Still applying Eq. (3) to Eq. (18), it yields

$$\begin{aligned}\mathbf{1}_{1\times n}\cdot B(k) &= c_1x_1(k-1)+\cdots+c_ix_i(k-1)\\ &\quad+\ \cdots+c_nx_n(k-1)\\ &=\sum_{i=1}^{n}c_ix_i(k-1)=b.\end{aligned}\tag{19}$$

So, Eq. (14b) holds. ■

Moreover, there is a special case for Proposition 1, when the degrees of all nodes are identical, so the proposition 1 has a corollary below.

Corollary: Assume a network $G(V,E)$ composed of n agents is built for the optimization problem Eq. (4). When the degrees of all nodes are identical to d, Eqs. (12) and (12) can be reduced to

$$A\cdot\begin{bmatrix}c_1x_1\\ \vdots\\ c_ix_i\\ \vdots\\ c_nx_n\end{bmatrix}=A\cdot\begin{bmatrix}c_1x_1(k-1)\\ \vdots\\ c_ix_i(k-1)\\ \vdots\\ c_nx_n(k-1)\end{bmatrix}.\tag{20}$$

If the constraints are decomposed by the above equation, then Eqs. (14a) and (14b) still hold.

Proof: If the degrees of all nodes are identical to d, the matrix D is reduced to

$$D=\begin{bmatrix}d_{11}&\cdots&0&\cdots&0\\ \vdots&\ddots&\vdots&\vdots&\vdots\\ 0&\cdots&d_{ii}&\cdots&0\\ \vdots&\vdots&\vdots&\ddots&\vdots\\ 0&\cdots&0&\cdots&d_{nn}\end{bmatrix}=d\cdot\begin{bmatrix}1&\cdots&0&\cdots&0\\ \vdots&\ddots&\vdots&\vdots&\vdots\\ 0&\cdots&1&\cdots&0\\ \vdots&\vdots&\vdots&\ddots&\vdots\\ 0&\cdots&0&\cdots&1\end{bmatrix}.\tag{21}$$

So, the weighted matrix W is reexpressed as

$$W=A\cdot D^{-1}=\frac{1}{d}\cdot A.\tag{22}$$

According to Proposition 1, we have

$$W\cdot\begin{bmatrix}c_1x_1\\ \vdots\\ c_ix_i\\ \vdots\\ c_nx_n\end{bmatrix}=\begin{bmatrix}b_1(k)\\ \vdots\\ b_i(k)\\ \vdots\\ b_n(k)\end{bmatrix}=W\cdot\begin{bmatrix}c_1x_1(k-1)\\ \vdots\\ c_ix_i(k-1)\\ \vdots\\ c_nx_n(k-1)\end{bmatrix}.\tag{23}$$

Applying Eq. (22) to Eq. (23), it yields

$$A\cdot\begin{bmatrix}c_1x_1\\ \vdots\\ c_ix_i\\ \vdots\\ c_nx_n\end{bmatrix}=A\cdot\begin{bmatrix}c_1x_1(k-1)\\ \vdots\\ c_ix_i(k-1)\\ \vdots\\ c_nx_n(k-1)\end{bmatrix}.\tag{24}$$

In terms of Proposition 1, Eq. (24) satisfies Eqs. (14a) and (14b), where it is worth noting that the inverse of the matrix A does not always exist. ■

Furthermore, after decomposition of objective functions and constraints, our parallel and distributed optimization method can approach the optima of the optimization problem Eq. (6) by a number of iterations. We take the Proposition 2 as an example to show how our method works.

Proposition 2: Assume a ring network $G(V,E)$ composed of $n=10$ agents, as shown in Fig. (2), is built for an optimization problem as below,

$$\begin{cases}\min\ \sum_{i=1}^{10}x_i^2\\ \text{s.t.}\ \ \sum_{i=1}^{10}ix_i=50.\end{cases}\tag{25}$$

Applying our parallel and distributed optimization method, after 50 iterations, the maximum deviation between the global optimal solutions X^* and the solutions X found by our method is less than or equal to 2×10^{-4}.

Proof: From the ring network $G(V,E)$, the adjacency matrix A and the degree matrix D can be found and then the weighted matrix W can be calculated,

$$W=\begin{bmatrix}\frac13&\frac13&&&&&&&&\frac13\\ \frac13&\frac13&\frac13&&&&&&&\\ &\frac13&\frac13&\frac13&&&&&&\\ &&\frac13&\frac13&\frac13&&&&&\\ &&&\frac13&\frac13&\frac13&&&&\\ &&&&\frac13&\frac13&\frac13&&&\\ &&&&&\frac13&\frac13&\frac13&&\\ &&&&&&\frac13&\frac13&\frac13&\\ &&&&&&&\frac13&\frac13&\frac13\\ \frac13&&&&&&&&\frac13&\frac13\end{bmatrix}.\tag{26}$$

In terms of Eqs. (6) and (12), the objective functions and constraints of the optimization problem Eq. (25) can be decomposed for all local areas as follows,

$$\begin{aligned}&\text{Local area 1:}\quad\begin{cases}\min\ \frac13x_1^2+\frac13x_2^2+\frac13x_{10}^2\\ \text{s.t.}\ \ \frac13x_1+\frac23x_2+\frac{10}{3}x_{10}=b_1(0)=50,\end{cases}\\ &\qquad\vdots\\ &\text{Local area } i\text{:}\quad\begin{cases}\min\ \frac13x_{i-1}^2+\frac13x_i^2+\frac13x_{i+1}^2\\ \text{s.t.}\ \ \frac{i-1}{3}x_{i-1}+\frac{i}{3}x_i+\frac{i+1}{3}x_{i+1}=b_i(0)=0,\end{cases}\\ &\qquad\vdots\\ &\text{Local area 10:}\quad\begin{cases}\min\ \frac13x_9^2+\frac13x_{10}^2+\frac13x_1^2\\ \text{s.t.}\ \ 3x_9+\frac{10}{3}x_{10}+\frac13x_1=b_{10}(0)=0.\end{cases}\end{aligned}\tag{27}$$

Applying the centralized interior-point algorithm [38], the global optimal solution $X^*=[x^*]_{n\times1}$ of Eq. (25) can be found. The absolute values of the deviation between X^* and X after 10 and 50 iterations are listed below, respectively,0

$$\left|\begin{bmatrix}x_1(10)-x_1^*\\ x_2(10)-x_2^*\\ x_3(10)-x_3^*\\ x_4(10)-x_4^*\\ x_5(10)-x_5^*\\ x_6(10)-x_6^*\\ x_7(10)-x_7^*\\ x_8(10)-x_8^*\\ x_9(10)-x_9^*\\ x_{10}(10)-x_{10}^*\end{bmatrix}\right|=\left|\begin{bmatrix}0.1359-0.1299\\ 0.3173-0.2597\\ 0.5373-0.3896\\ 0.6942-0.5195\\ 0.8092-0.6494\\ 0.8846-0.7792\\ 0.9386-0.9091\\ 0.9900-1.0390\\ 1.0566-1.1688\\ 1.1488-1.2987\end{bmatrix}\right|=\begin{bmatrix}0.0060\\ 0.0575\\ 0.1477\\ 0.1748\\ 0.1598\\ 0.1053\\ 0.0295\\ 0.0489\\ 0.1123\\ 0.1499\end{bmatrix},\tag{28}$$

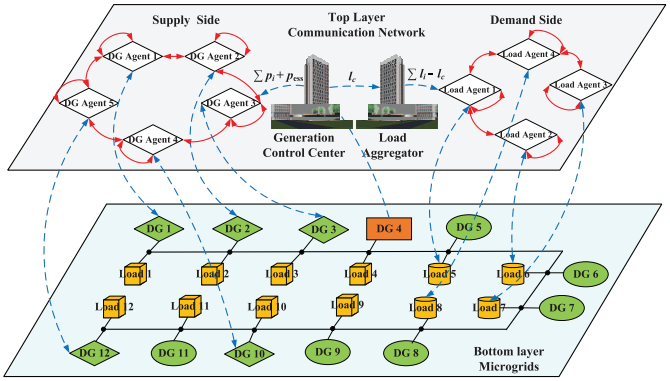


Fig. 3. The two layer optimization model of economic dispatch with DR, where dispatchable DGs are indicated by diamonds, non-dispatchable DGs are indicated by circles, and the ESS operates as a DG indicated by the rectangle. Moreover, flexible loads are indicated by cylinders while conventional loads are indicated by cubes.

$$\left\|\begin{bmatrix} x_1(50)-x_1^* \\ x_2(50)-x_2^* \\ x_3(50)-x_3^* \\ x_4(50)-x_4^* \\ x_5(50)-x_5^* \\ x_6(50)-x_6^* \\ x_7(50)-x_7^* \\ x_8(50)-x_8^* \\ x_9(50)-x_9^* \\ x_{10}(50)-x_{10}^* \end{bmatrix}\right\| = \left\|\begin{bmatrix} 0.1299-0.1299 \\ 0.2598-0.2597 \\ 0.3898-0.3896 \\ 0.5197-0.5195 \\ 0.6496-0.6494 \\ 0.7794-0.7792 \\ 0.9091-0.9091 \\ 1.0389-1.0390 \\ 1.1687-1.1688 \\ 1.2985-1.2987 \end{bmatrix}\right\| = \begin{bmatrix} 0.0000 \\ 0.0001 \\ 0.0002 \\ 0.0002 \\ 0.0002 \\ 0.0002 \\ 0.0000 \\ 0.0001 \\ 0.0001 \\ 0.0002 \end{bmatrix}. \tag{29}$$

According to (29), it can be seen that the maximal deviation is 2×10^{-4}. Thus, it is concluded that the optimal solutions X found by our method approximately converge to the global optimal solutions X^*. Moreover, it can be predicted that a lower deviation will be achieved after more iterations. ■

III. Energy Management for MGs

Generally, economic dispatch for MGs regulates outputs of DGs for achieving optimal system performance. When demand response is introduced to economic dispatch, the supply side and demand side can cooperate with each other and then improved performance can be reached. Recently, flexible loads of consumers, as important sources for demand response, have played a significant role in the economic dispatch of MGs, such as for peak shaving and valley filling, in response to time-based rates or other forms of financial incentives.

A. Model of Economic Dispatch With Demand Response

In this paper, an optimization model of economic dispatch with demand response is shown in Fig. 3, which is built as a communication network composed of agents (as the top layer) over an islanded MG (as the bottom layer). In an islanded MG, there is a DG working in the voltage and frequency control (V/F control) mode, called V/F DG (indicated as a rectangle in Fig. 3), which provides the voltage and frequency reference for the MG. Therefore, if there is a mismatch in the MG, the V/F DG will inject or absorb power to balance the system, so the outputs $p_{\text{ess}}(t)$ of the V/F DG indicate the mismatch of the system. Here, a battery energy storage system (BESS) is employed as the V/F DG. In addition, to improve the penetration of renewable energies, the outputs of photovoltaics (PVs) and wind turbines (WTs) are not regulated, so these DGs are called non-dispatchable DGs (indicated as circles). Conversely, the outputs of some DGs, like micro-turbines (MTs), can be regulated according to control signals, so they are called dispatchable DGs (indicated as diamonds). On the demand side, besides the conventional loads, there are some flexible loads that also can be regulated in terms of the needs of the system.

On the top layer, the communication network consists of two subgraphs (a subgraph with DG agents and a subgraph with load agents) that are connected by the generation control center and the load aggregator. Information is collected from DGs and loads first, and then it is exchanged with neighbor agents. Here, agents connecting to dispatchable DGs are called DG agents and those connecting to flexible loads are called load agents, while the generation control center connects to the V/F DG. However, no agents connect to non-dispatchable DGs and conventional loads, because the outputs $p_{\text{ess}}(t)$ of the V/F DG indicate the mismatch of the system, which is a way to reduce the communication complexity. If a mismatch between supply and demand occurs, the BESS will balance the system by injecting or absorbing power immediately. However, the stored energy of the BESS is limited, so the BESS cannot inject or absorb constantly for a long time. In this case, the generation control center sends the mismatch information to a DG agent. Further, all information is processed on the subgraph with DG agents, i.e., dispatchable DGs are regulated in order to share the mismatch and minimize the generation cost functions using our parallel and distributed optimization method. After this operation, the outputs of the V/F DG will return to zeros gradually.

On the other hand, if peak shaving or valley filling is needed, the generation control center will send the values $l_c(t)$ of peak shaving or valley filling to the load aggregator, who is an important coordinator between supply side and demand side. Next, the load aggregator sends the information to a load agent and then the information is dealt with on the subgraph with load agents by maximizing consumers profit functions in a distributed manner. Finally, flexible loads are regulated.

B. Supply-Side: Generation Cost Functions

On the supply-side, minimizing the generation costs in MGs is a main task for economic dispatch. Generally, only the generation costs of active power of dispatchable DGs, e.g., MTs, are considered because fossil flue is consumed, while the costs of renewables and reactive power are not considered. Therefore, in this paper, the generation cost function for MTs is given below and it is a quadratic convex function of active power p_i [39],

$$f_i(p_i) = \alpha_i p_i^2 + \beta_i p_i + \gamma_i, \tag{30}$$

where the nonnegative values, α_i, β_i and γ_i, are the cost coefficients of an MT$_i$.

If there are n MTs in an MG, the economic dispatch is to minimize the total generation costs of these MTs. So, the

optimization model for the supply-side can be described as follows,

$$\begin{cases} \min \sum_{i=1}^{n} f_i(p_i(t)) \\ \text{s.t.} \ \sum_{i=1}^{n} p_i(t) = \sum_{i=1}^{n} p_i(t-1) + p_{\text{ess}}(t-1) \end{cases}. \quad (31)$$

Further, the optimization model Eq. (31) can be decomposed for DG agents in terms of the steps in Section II. Applying our method, the minima of the problem can be found iteratively and in a distributed way. When the minimized generation cost is achieved, the incremental costs will reach a consensus, where the incremental cost is defined as [39],

$$\lambda_i = \frac{\partial f_i(p_i)}{\partial p_i} = 2\alpha_i p_i + \beta_i \quad (32)$$

which is the first derivative of the generation cost function with respect to active power.

C. Demand-Side: Consumers Profit Functions

On demand-side, flexible loads are regulated to maximize consumers' profit functions that consists of two parts, a utility function U_i and a cost function for power consumption E_i. The utility function of a flexible load has to follow the three properties, i.e., a) the utility function should be nondecreasing; b) the utility function is zero when no power is consumed by the flexible load; c) there is a saturation point for the utility function. Therefore, the utility function for a flexible load can be defined as a quadratic function as below,

$$U_i(l_i(t)) = \begin{cases} \delta_i l_i(t) - \frac{\omega_i}{2}(l_i(t))^2, & 0 < l_i(t) < \frac{\delta_i}{\omega_i} \\ \frac{\delta^2}{2\omega_i}, & l_i(t) \geq \frac{\delta_i}{\omega_i} \end{cases}, \quad (33)$$

where $l_i(t)$ is the magnitude of the flexible load i at time t and the saturation point is at δ_i/ω_i. Both ω_i and δ_i are the utility coefficients to differentiate flexible loads.

In addition, a cost function for power consumption of the flexible load i can be expressed as

$$E_i(l_i(t)) = g(t) \cdot l_i(t), \quad (34)$$

where $g(t)$ is the electricity rate at time t. Thus, the profit function of the flexible load i takes form as

$$H_i(l_i(t)) = U_i(l_i(t)) - E_i(l_i(t)). \quad (35)$$

Assume there are m flexible loads, the optimization problem on demand side can be described as follows,

$$\begin{cases} \min \sum_{i=1}^{m} -H_i(l_i(t)) \\ \text{s.t.} \ \sum_{i=1}^{m} l_i(t) = \sum_{i=1}^{m} l_i(t-1) - l_c(t-1) \end{cases}, \quad (36)$$

where $l_c(t-1)$ is the values of peak shaving or valley filling received from the load aggregator.

IV. Microgrid System Architecture and Setup

For simulations, an islanded MG with twelve DGs and twelve loads is established in MATLAB/Simulink as shown in Fig. 4, where there are six PVs and WTs, five MTs, an BESS and four conventional loads (CL) and eight flexible loads (FL). PVs and WTs, $\{\text{DG}_i | i = 2, 6, 8, 9, 10, 12\}$, work in the maximum power point tracking (MPPT) mode, while MTs, $\{\text{DG}_i | i = 1, 3, 5, 7, 11\}$, work in in the active and reactive power control (PQ control) mode. And the BESS, $\{\text{DG}_i | i = 4\}$, works in the V/F control mode. Additionally, conventional loads, $\{\text{Load}_i | i = 4, 6, 8, 10\}$, can only be connected to or cut from the system as a whole, while flexible loads, $\{\text{Load}_i | i = 1, 2, 3, 5, 7, 9, 11, 12\}$, can be controlled according to load agents.

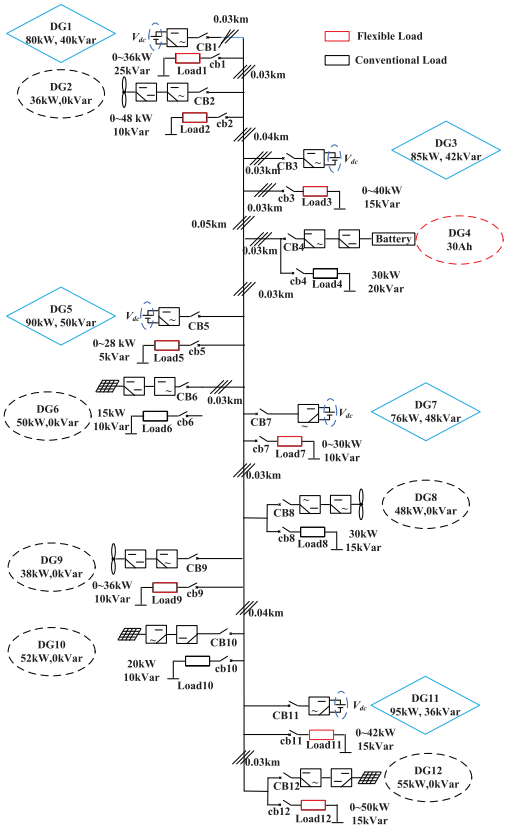


Fig. 4. The diagram of an islanded MG.

TABLE I
Setup and Parameters of DGs

Sources	DG ratings	Control	α	β	γ
DG_1	80 kW, 40 kVar	PQ	0.059	6.71	80
DG_2	36 kW, 0 kVar	MPPT			
DG_3	85 kW, 42 kVar	PQ	0.066	6.29	43
DG_4	30 Ah	V/F			
DG_5	90 kW, 50 kVar	PQ	0.046	7.53	35
DG_6	50 kW, 0 kVar	MPPT			
DG_7	76 kW, 48 kVar	PQ	0.069	4.57	48
DG_8	48 kW, 0 kVar	MPPT			
DG_9	38 kW, 0 kVar	MPPT			
DG_{10}	52 kW, 0 kVar	MPPT			
DG_{11}	95 kW, 36 kVar	PQ	0.052	5.89	42
DG_{12}	55 kW, 0 kVar	MPPT			

In all simulations, MTs can provide active and reactive power, while WTs and PVs are restricted to only provide active power. The frequency and line voltage of the MG are set at 50 Hz and 380 V, respectively. Besides, the line losses are considered for the line impedance is set at 0.641+j0.101 Ω/km. Furthermore, assume the MG works in a balanced state initially. In summary, the detailed parameters and setups of DGs and loads are listed in Table I and Table II, respectively.

TABLE II
SETUP AND PARAMETERS OF LOADS

Load	Max.Demand	Load Type	ω	σ
$Load_1$	0~36kW, 25kVar	Flexible Load	-0.041	2.43
$Load_2$	0~48kW, 10kVar	Flexible Load	-0.040	2.86
$Load_3$	0~40kW, 15kVar	Flexible Load	-0.026	1.98
$Load_4$	30kW , 20kVar	Conventional Load		
$Load_5$	0~28kW, 5kVar	Flexible Load	-0.081	3.21
$Load_6$	15 kW , 10kVar	Conventional Load		
$Load_7$	0~30kW, 10kVar	Flexible Load	-0.053	2.52
$Load_8$	30 kW , 15kVar	Conventional Load		
$Load_9$	0~36kW, 10kVar	Flexible Load	-0.029	1.98
$Load_{10}$	25 kW , 10kVar	Conventional Load		
$Load_{11}$	0~42kW, 15kVar	Flexible Load	-0.063	3.62
$Load_{12}$	0~50kW, 15kVar	Flexible Load	-0.026	2.24

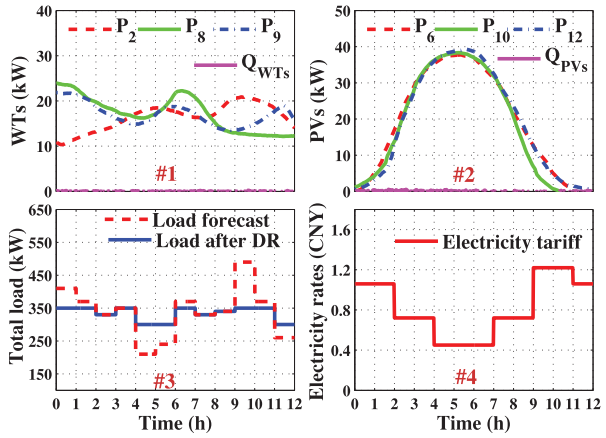


Fig. 5. Setups of WTs, PVs, loads and electricity rates. (a) Outputs of WTs. (b) Outputs of PVs. (c) Total load forecast and total load after DR. (d) The time-of-use electricity tariff.

Throughout all simulations, the outputs of WTs and PVs are shown in Fig. 5(#1) and (#2), respectively. Moreover, the load forecast (curve in red) and the expected loads after DR (curve in blue) are shown in Fig. 5(#3), so the values $l_c(t)$ of peak shaving or valley filling can be obtained by subtracting the expected loads from the load forecast, while the time-of-use electricity tariff (electricity rate) is shown in Fig. 5(#4).

V. RESULTS

To test the performance of our parallel and distributed optimization method, four cases are designed. The first case focuses on the comparisons of the accuracy of found optima and the convergence rate (or the number of iterations) for our method, the interior point algorithm and the standard ADMM, while the second case focuses on the comparison of the system performance under three different optimization methods. Finally, the last two cases examine the impacts of different topologies of communication networks on the performance of the MG, and the system performance when plug-and-play occurs.

A. Case1: Comparison of Accuracy and Convergency Rate of Different Optimization Methods

In this case, there are two parts, where in the first part, the accuracies of the optima found by our method and the centralized interior-point algorithm (IPA) are compared, while the numbers of iterations of our method are compared with those of the standard ADMM. To test the accuracy, four convex functions are selected or modified from standard functions as objective functions, like Eq. (4a), which are listed in Table III.

TABLE III
DESCRIPTION OF FUNCTIONS FOR TESTS

No.	Names of Functions for Tests	Description
I	Sphere Function	$\min \sum_{i=1}^{n} f_i(x_i) = x_i^2$
II	Sum of Squares Function	$\min \sum_{i=1}^{n} f_i(x_i) = ix_i^2$
III	Sum of Different Powers Functions	$\min \sum_{i=1}^{n} f_i(x_i) = \lvert x_i \rvert^{i+1}$
IV	Modified Zakharov Function	$\sum_{i=1}^{n}(x_i^2 + 0.5ix_i^2 + 0.5ix_i^4)$

To create an optimization problem with constraints, an equality constraint, such as Eq. (4b), is also needed, in which the range of coefficients is given below,

$$-20 \le c_i \le 20. \tag{37}$$

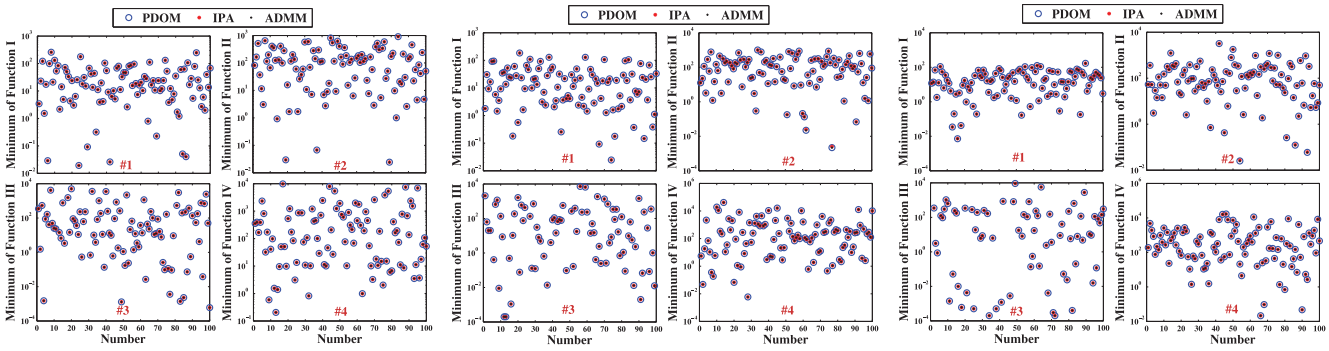
For a function for test in Table III, a constraint is produced by choosing a set of $\{c_i | i = 1, \ldots, n\}$ at random and the value of b should select carefully to guarantee the existence of minimums of each function for test, so an optimization problem with constraints, like Eq. (4), is formed. And then a communication network is built for agents. When the scales of optimization problems rise a lot, e.g., the number of $f_i(x_i)$ rises from ten to thirty, the number n of agents rises correspondingly. For our method, there is no restriction on the structure of a communication network, so a standard ring network as Fig. 1 is used when $n = 10$, while a ring network with degree of three (not counting self-loop) is used when $n = 20$ and 30. For the standard ADMM, a standard ring network is always used for different n due to its working in cascade.

To reduce randomness, for each function in Table III, 100 constraints are produced by choosing c_i at random, so 100 optimization problems with different constraints are formed. After these optimization problems are solve by IPA, our method and ADMM, respectively, minima are obtained, all of which are drawn on one of twelve sub-figures in Fig. 6.

Thus, Fig. 6 shows the minima of four functions for tests, where the minima obtained by IPA is represented as red points, those by our method as blue circles and those by ADMM as black crosses. From Fig. 6, it can be seen that the minima found by our method overlap or almost overlap with those by IPA and ADMM, no matter which test function is used, which means the accuracy of our method is consistent to that of IPA.

Under a different number n of agents, for each test function, the mean absolute errors (MAEs) of minima and the statistical averages of numbers of iterations are calculated, respectively, where the minima found by IPA are considered as the true values to find MAEs of our method and the standard ADMM. Finally, the MAEs and the computation time of different methods are listed in Table IV. Note that our parallel method is simulated on a common computer, because a parallel computer is not available.

As shown in Table IV, comparing the MAEs, it can be seen that all statistical averages are very close, which means no matter which method, centralized or distributed, is used, the



(a) n=10 (b) n=20 (c) n=30

Fig. 6. Comparison of accuracy between our method (PDOM), interior-point algorithm (IPA) and ADMM. Remarks: a logarithmic axis is used due to the significant differences among the found minimums, and those minimums very close to zeros are not drawn on the figures with a logarithmic axis.

TABLE IV
COMPARISONS OF ACCURACY, NUMBER OF ITERATIONS (NOI) AND COMPUTATION TIME (CT)

Functions	n	MAE (PDOM)	MAE (ADMM)	NoI (PDOM)	NoI (ADMM)	CT (s) (PDOM)	CT (s) (ADMM)
I	10	8.23×10^{-5}	2.30×10^{-3}	44	131	0.29	0.71
	20	5.52×10^{-5}	3.71×10^{-4}	63	348	0.65	2.18
	30	3.24×10^{-4}	3.23×10^{-5}	131	904	1.73	6.80
II	10	1.00×10^{-3}	0.02	58	186	0.17	0.30
	20	7.10×10^{-4}	0.01	113	345	0.48	0.56
	30	5.00×10^{-3}	0.01	242	507	1.24	0.83
III	10	1.30×10^{-6}	0.05	32	566	0.14	1.36
	20	6.47×10^{-5}	0.08	264	5940	1.94	11.31
	30	4.61×10^{-4}	0.04	279	10191	2.67	18.89
IV	10	8.53×10^{-6}	0.16	54	1015	0.22	2.67
	20	1.29×10^{-5}	0.11	94	1443	0.54	3.86
	30	9.91×10^{-5}	0.08	220	1734	1.54	4.66

minima can be found. On the other hand, when observing the middle two columns and focusing on the numbers of iterations, we can find the number of iterations grows with the increased scale of optimization problems and it also grows with the complexity of test functions. For example, the number of iterations when $n = 30$ is much greater than that when $n = 10$. Moreover, the number of iterations for the third function is significantly greater than that for the first function when $n = 30$.

Compared the number of iterations using our method with that using ADMM, one can find that the number of iterations using our method is much less than that using ADMM, no matter what n is employed. With the rise of complexity of functions, e.g., the third function, compared the number of iterations using our method and ADMM when $n = 30$, it can be seen that the number of iterations using ADMM reach a very large number, whereas this is significantly smaller using our method. In summary, under the same accuracy, the convergence rate of our parallel and distributed method (PDOM) is much faster than that of the standard ADMM. In other words, the number of iterations in our method is only one third or even less than that of ADMM.

B. Case2: Comparison of System Performance Under Different Optimization Methods

The economic dispatch with DR is a typical optimization problem, which is modeled in Section III. To compare the system performance under different optimization methods,

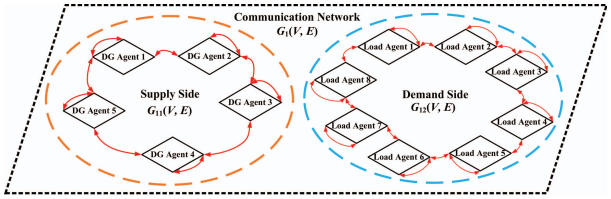


Fig. 7. Topology of a communication network $G_1(V, E)$.

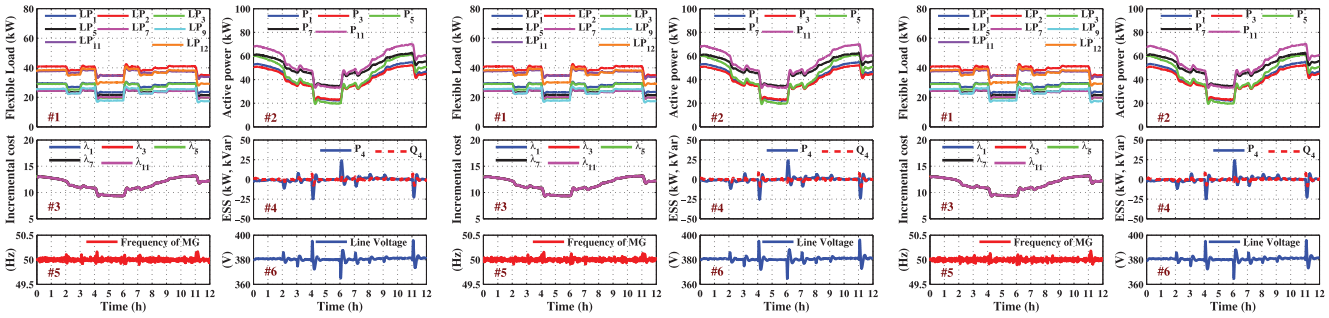
three methods, a centralized method (IPA), ADMM and our method, are applied to solve the economic dispatch problems in an MG. Note that Fig. 4 shows the structure of a radial MG that is used for simulations, where its parameters are listed in Table I and Table II, and the outputs of renewable energies and the values of peak shaving or valley filling are shown in Fig. 5. Before simulations, the topology of the communication network composed of agents $G_1(V, E)$ is shown in Fig. 7, where there are two subgraphs, i.e., $G_{11}(V, E)$ for supply side and $G_{12}(V, E)$ for demand side.

Under these settings, three optimization methods are applied and the simulation results are shown in Fig. 8.

Comparing the results obtained by three methods, we can see the system performance is almost the same no matter what method is applied. Moreover, the voltage and the frequency stay very close to 380 V and 50 Hz, respectively, except the regulations of flexible loads. And the outputs of the BESS satisfy the desire that it balances the system instantaneously and then its outputs are shared by dispatchable DGs on supply side.

In terms of Eq. (32), the incremental costs λ can be calculated according to the active power outputs of dispatchable DGs. From the results, we can see that the incremental costs of all dispatchable DGs reach a consensus, which indicates that the minimal costs on the supply side are achieved. Summarily, from these observations, they means the system works well and its performance satisfies the requirements.

Moreover, the performance of DR on flexible loads of consumers are investigated, when the values of peak shaving or valley filling are given. So, simulations are carried out and results are shown in Fig. 9, where the profits of consumers with and without DR obtained by our method are compared.



(a) IPA (b) ADMM (c) PDOM

Fig. 8. Comparison of system performance when interior-point algorithm (IPA), ADMM and our method (PDOM) are applied, respectively.

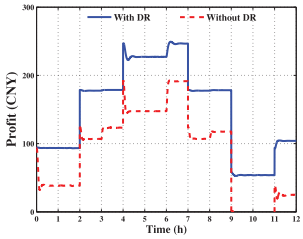


Fig. 9. Simulation results of the profits with DR versus without DR.

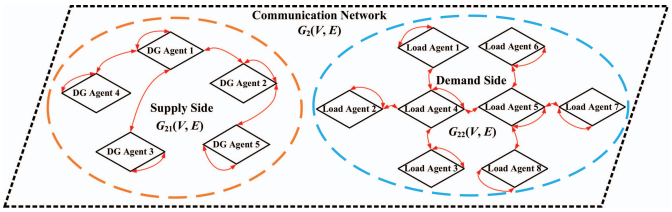


Fig. 10. Topology of a communication network $G_2(V, E)$.

On demand side, in response to the requirements for peak shaving or valley filling, flexible loads are regulated to join the demand response and increase consumers' profits. For example, when valley filling is needed, a relatively low electricity rate is given to attract consumers to increase their loads. In this way, valley filling is completed and at the same time consumers get more benefits. Compared the profits obtained with DR to those without DR, as shown in Fig. 9, it can be found that the profits with DR (curve in blue) are significantly greater than those without DR (curve in red).

C. Case3: Impacts of Different Topology on System Performance

In this section, another topology of a communication network $G_2(V, E)$ as shown in Fig. 10 is used to investigate the impacts of different topology on the system performance. Compared to the network $G_1(V, E)$ in Fig. 7, it can be found that the links of two subgraphs in $G_2(V, E)$ are less than those in $G_1(V, E)$, where less links lower the cost to build a communication network. To test the performance on the economic dispatch with DR on the network $G_2(V, E)$, all settings follow those in Case 2.

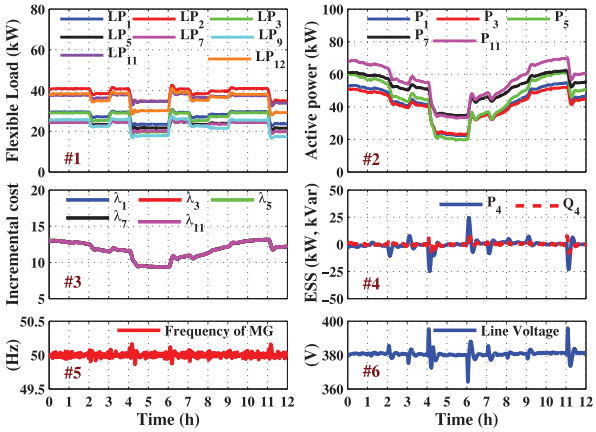


Fig. 11. Simulation results of economic dispatch on a different network $G_2(V, E)$.

After simulations, the results are shown in Fig. 11. From Fig. 11, observing the curves of frequency and voltage, they indicate the MG runs well owing to the regulation of dispatchable DGs and flexible loads, where the references for DGs and loads are obtained by our method. Compared to the results in Case 2, the results on $G_2(V, E)$ is very similar to them. In summary, it can be concluded that the our method can work on different topologies of networks, which offers a way to build communication networks without many constraints.

D. Case4: Impacts of Plug-and-Play on System Performance

In an MG, a common operation is DG cutoff due to failures or DG connection after recovery, a.k.a. plug-and-play. For our method, DG cutoff or connection in an MG corresponds to the increase or decrease of agents on the communication network, which only changes the size of the adjacency matrix A, so the proposed method still works due to its scalability. For example, if a DG is cut off from the MG, then the corresponding agent on the communication network will not deal with any information but only forward information it receives to its neighbors. It is worth noting that cutting off agents from the communication network directly may cause the communication network not connected, which makes the method not work. This is why we do not cut off agents but make them forward information.

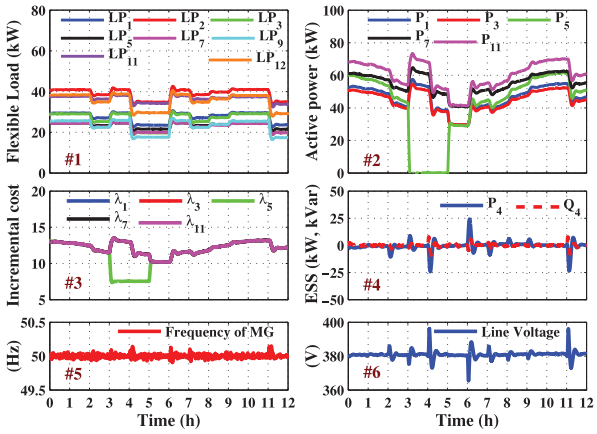


Fig. 12. Simulation results of economic dispatch, when the plug-and-play occurs.

Therefore, this case is designed to test the system performance when the plug-and-play occurs in the MG, where DG_5 fails at $t = 3$ h and it is cut off from the MG, while it recovers at $t = 5$ h and connects to the MG again. Other settings follows those in Case 2. Under these settings, simulations are carried out and the results are shown in Fig. 12.

From Fig. 12(#2), it can be seen that at $t = 3$ h DG_5 is cut off from the MG due to failure, so its outputs becomes zero. Correspondingly, its incremental cost λ_5 is not consistent with others. However, the system still works well for the voltage and frequency maintain at prescribed values. On the other hand, when the DG recovers at $t = 5$ h, it reconnects to the MG again. At this time, the corresponding agent restarts, so the incremental cost λ_5 returns to consistence again. As the simulation results demonstrate, our method still works well, even if plug-and-play occurs.

VI. Conclusion

We have proposed a parallel and distributed optimization method for energy management of MGs. The aim of our parallel and distributed optimization method is to increase convergence rate of distributed optimization methods. So, the objective functions and constraints of the original optimization problem have to be decomposed, which forms the local objective functions and local constraints for agents. The local objective functions and local constraints are only related to an agent and its neighbors, so agents can solve the decomposed optimization problems locally and independently. Further, the obtained optima are exchanged with their neighbors and then they update their local constraints. Consequently, the optima of the original optimization problem can be found in parallel by iterations. Also, two propositions and a corollary are proved to indicate the convergence of our method. Finally, an MG is built in MATLAB/Simulink, where a typical optimization problem, namely the economic dispatch with DR, is solved by our method in a distributed manner.

Simulations are carried out and the results are summarized specifically. First, the accuracy of optima using our method is compared to that using IPA and ADMM, where the results show that all these three methods find almost the same optima. However, when the number of iterations in our method is compared to that of ADMM, it can be seen that the convergence rate of our method is significantly faster than that of ADMM. Second, no matter what distributed methods (our method or ADMM) or centralized methods are used, the simulation results also show that both the minima of cost functions on the supply side and the maxima of profit functions on demand side are obtained, which means that our method can complete the economic dispatch of MGs. Third, when the startup and shutdown of DGs occur, the simulation results show that our distributed method still can run the MG well due to the scalability of the method. For future work, considering nonlinear constraints and analyzing the convergence in more general conditions in theory are significant questions.

References

[1] B. Chen, J. Wang, X. Lu, C. Chen, and S. Zhao, "Networked microgrids for grid resilience, robustness, and efficiency: A review," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 18–32, Jan. 2021.

[2] D. Y. Yamashita, I. Vechiu, and J.-P. Gaubert, "A review of hierarchical control for building microgrids," *Renew. Sustain. Energy Rev.*, vol. 118, Feb. 2020, Art. no. 109523.

[3] M. Parvez, A. T. Pereira, N. Ertugrul, N. H. E. Weste, D. Abbott, and S. F. Al-Sarawi, "Wide bandgap DC-DC converter topologies for power applications," *Proc. IEEE*, vol. 109, no. 7, pp. 1253–1275, Jul. 2021.

[4] D. E. Olivares *et al.*, "Trends in microgrid control," *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1905–1919, Jul. 2014.

[5] N. Ertugrul and D. Abbott, "DC is the future [point of view]," *Proc. IEEE*, vol. 108, no. 5, pp. 615–624, May 2020.

[6] H. Moradi, M. Esfahanian, A. Abtahi, and A. Zilouchian, "Optimization and energy management of a standalone hybrid microgrid in the presence of battery storage system," *Energy*, vol. 147, pp. 226–238, Mar. 2018.

[7] P. P. Vergara, J. C. Lopez, M. J. Rider, H. R. Shaker, L. C. P. da Silva, and B. N. Jorgensen, "A stochastic programming model for the optimal operation of unbalanced three-phase islanded microgrids," *Int. J. Electr. Power Energy Syst.*, vol. 115, Feb. 2020, Art. no. 105446.

[8] A. Askarzadeh, "A memory-based genetic algorithm for optimization of power generation in a microgrid," *IEEE Trans. Sustain. Energy*, vol. 9, no. 3, pp. 1081–1089, Jul. 2018.

[9] H. Shen *et al.*, "Economic optimisation of microgrid based on improved quantum genetic algorithm," *J. Eng.*, vol. 16, no. 16, pp. 1167–1174, Mar. 2019.

[10] K. Gholami and E. Dehnavi, "A modified particle swarm optimization algorithm for scheduling renewable generation in a micro-grid under load uncertainty," *Appl. Soft. Comput.*, vol. 78, pp. 496–514, May 2019.

[11] M. A. Hossain, H. R. Pota, S. Squartini, and A. F. Abdou, "Modified PSO algorithm for real-time energy management in grid-connected microgrids," *Renew. Energy.*, vol. 136, pp. 746–757, Jun. 2019.

[12] L. van Hoorebeeck, P.-A. Absil, and A. Papavasiliou, "MILP-based algorithm for the global solution of dynamic economic dispatch problems with valve-point effects," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, 2019, pp. 1–5.

[13] S. S. Torbaghan *et al.*, "Optimal flexibility dispatch problem using second-order cone relaxation of AC power flows," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 98–108, Jan. 2020.

[14] M. Nemati, M. Braun, and S. Tenbohlen, "Optimization of unit commitment and economic dispatch in microgrids based on genetic algorithm and mixed integer linear programming," *Appl Energy*, vol. 210, pp. 944–963, Jan. 2018.

[15] W. Hu, P. Wang, and H. B. Gooi, "Toward optimal energy management of microgrids via robust two-stage optimization," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 1161–1174, Mar. 2018.

[16] H. Zhong, X. Yan, and Z. Tan, "Real-time distributed economic dispatch adapted to general convex cost functions: A secant approximation-based method," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2089–2101, May 2021.

[17] M. H. Albadi and E. F. El-Saadany, "A summary of demand response in electricity markets," *Electr. Power Syst. Res.*, vol. 78, no. 11, pp. 1989–1996, 2008.

[18] K.-Y. Huang and Y.-C. Huang, "Integrating direct load control with interruptible load management to provide instantaneous reserves for ancillary services," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1626–1634, Aug. 2004.

[19] F. Aminifar, M. Fotuhi-Firuzabad, and M. Shahidehpour, "Unit commitment with probabilistic spinning reserve and interruptible load considerations," *IEEE Trans. Power Syst.*, vol. 24, no. 1, pp. 388–397, Feb. 2009.

[20] L. Xiong, P. Li, Z. Wang, and J. Wang, "Multi-agent based multi objective renewable energy management for diversified community power consumers," *Appl Energy*, vol. 259, Feb. 2020, Art. no. 114140.

[21] M. Norouzi, J. Aghaei, S. Pirouzi, T. Niknam, and M. Lehtonen, "Flexible operation of grid-connected microgrid using ES," *IET Gener. Transm. Distrib.*, vol. 14, no. 2, pp. 254–264, 2020.

[22] R. Sabzehgar, M. A. Kazemi, M. Rasouli, and P. Fajri, "Cost optimization and reliability assessment of a microgrid with large-scale plug-in electric vehicles participating in demand response programs," *Int. J. Green Energy.*, vol. 17, no. 2, pp. 127–136, 2020.

[23] M. H. K. Tushar, A. W. Zeineddine, and C. Assi, "Demand-side management by regulating charging and discharging of the EV, ESS, and utilizing renewable energy," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 117–126, Jan. 2018.

[24] X. Lu, K. Zhou, S. Yang, and H. Liu, "Multi-objective optimal load dispatch of microgrid with stochastic access of electric vehicles," *J. Clean Prod.*, vol. 195, pp. 187–199, Sep. 2018.

[25] N. Liu, L. He, X. Yu, and L. Ma, "Multiparty energy management for grid-connected microgrids with heat- and electricity-coupled demand response," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 1887–1897, May 2018.

[26] T. Liu, X. Tan, B. Sun, Y. Wu, and D. H. Tsang, "Energy management of cooperative microgrids: A distributed optimization approach," *Int. J. Electr. Power Energy Syst.*, vol. 96, pp. 335–346, Mar. 2018.

[27] T. Zhao and Z. Ding, "Distributed finite-time optimal resource management for microgrids based on multi-agent framework," *IEEE Trans. Ind. Electron.*, vol. 65, no. 8, pp. 6571–6580, Aug. 2018.

[28] Q. Li, C. Peng, M. Wang, M. Chen, J. M. Guerrero, and D. Abbott, "Distributed secondary control and management of islanded microgrids via dynamic weights," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 2196–2207, Mar. 2019.

[29] X. Zhou and Q. Ai, "Distributed economic and environmental dispatch in two kinds of CCHP microgrid clusters," *Int. J. Electr. Power Energy Syst.*, vol. 112, pp. 109–126, Nov. 2019.

[30] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[31] S. G. M. Rokni, M. Radmehr, and A. Zakariazadeh, "Optimum energy resource scheduling in a microgrid using a distributed algorithm framework," *Sustain. Cities Soc.*, vol. 37, pp. 222–231, Feb. 2018.

[32] L. Chen, X. Zhu, J. Cai, X. Xu, and H. Liu, "Multi-time scale coordinated optimal dispatch of microgrid cluster based on MAS," *Elect. Power Syst. Res.*, vol. 177, Dec. 2019, Art. no. 105976.

[33] K. Jiang, F. Wu, X. Zong, L. Shi, and K. Lin, "Distributed dynamic economic dispatch of an isolated AC/DC hybrid microgrid based on a finite-step consensus algorithm," *Energies*, vol. 12, no. 24, p. 4637, Dec. 2019.

[34] G. Chen and Z. Zhao, "Delay effects on consensus-based distributed economic dispatch algorithm in microgrid," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 602–612, Jan. 2018.

[35] P. U. Herath *et al.*, "Computational intelligence-based demand response management in a microgrid," *IEEE Trans. Ind. Appl.*, vol. 55, no. 1, pp. 732–740, Jan./Feb. 2019.

[36] X. He, Y. Zhao, and T. Huang, "Optimizing the dynamic economic dispatch problem by the distributed consensus-based ADMM approach," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3210–3221, May 2020.

[37] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," *SIAM J. Optim.*, vol. 25, no. 2, pp. 944–966, 2015.

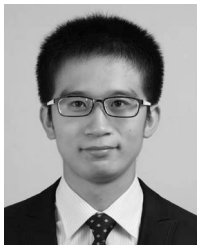
[38] S. Rojas-Labanda and M. Stolpe, "Benchmarking optimization solvers for structural topology optimization," *Struct. Multidiscipl. Optim.*, vol. 52, no. 3, pp. 527–547, 2015.

[39] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé, *Power Generation, Operation, and Control*, 3rd ed. New York, NY, USA: Wiley, 2013.

Qiang Li (Member, IEEE) received the Ph.D. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2009. He was a Postdoctoral Fellow with Chongqing University from 2009 to 2012, and a Visiting Postdoctoral Scholar with the University of Adelaide, Adelaide, SA, Australia, from 2011 to 2012. He is currently an Associate Professor with Chongqing University. His current research interests include networked control systems, optimization of microgrids, and evolutionary dynamics.

Yuexi Liao received the B.S. and M.S. degrees in electrical engineering from Chongqing University, Chongqing, China, in 2017 and 2020, respectively. His current research interests include multiagent systems and optimization of microgrids.

Kunming Wu received the B.S. and the M.S. degrees in electrical engineering from Chongqing University, Chongqing, China, in 2018 and 2021, respectively. His current research interests include distributed optimization of multimicrogrids.



Leiqi Zhang received the B.Eng. and Ph.D. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in July 2012 and July 2017, respectively. He is currently an Engineer with the State Grid Zhejiang Electric Power Research Institute, China. His research interests include distributed generation and control.

Jiayang Lin received the M.S. degree in electrical engineering from the China University of Geosciences, Wuhan, China. He is currently an Engineer with Wenzhou Power Supply Company.

Minyou Chen (Senior Member, IEEE) received the M.Sc. degree in control theory and engineering from Chongqing University, Chongqing, China, in 1987, and the Ph.D. degree in control engineering from the University of Sheffield, Sheffield, U.K., in 1998. He is currently a Full Professor with Chongqing University. His current research interests include intelligent modeling and control, multiobjective optimization, micro-grid control, and state monitoring in power distribution systems.

Josep M. Guerrero (Fellow, IEEE) received the B.S. degree in telecommunications engineering, the M.S. degree in electronics engineering, and the Ph.D. degree in power electronics from the Technical University of Catalonia, Barcelona, in 1997, 2000, and 2003, respectively. Since 2011, he has been a Full Professor with the Department of Energy Technology, Aalborg University, Denmark. His research interests are oriented to different microgrid aspects, including power electronics, hierarchical and cooperative control, energy management systems, and optimization of microgrids and islanded minigrids. He was the Chair of the Renewable Energy Systems Technical Committee of the IEEE Industrial Electronics Society. He is an Associate Editor for the IEEE TRANSACTIONS ON POWER ELECTRONICS, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, and the IEEE INDUSTRIAL ELECTRONICS MAGAZINE, and an Editor for the IEEE TRANSACTIONS ON SMART GRID.

Derek Abbott (Fellow, IEEE) was born in South Kensington, London, U.K., in 1960. He received the B.Sc. degree (Hons.) in physics from Loughborough University, Leicestershire, U.K., in 1982, and the Ph.D. degree in electrical and electronic engineering from The University of Adelaide, Adelaide, SA, Australia, in 1995, under K. Eshraghian and B. R. Davis. Since 1987, he has been with The University of Adelaide, where he is currently a Full Professor with the School of Electrical and Electronic Engineering. His interest is in the area of multidisciplinary physics and electronic engineering applied to complex systems. His research programs span a number of areas including networks, game theory, energy policy, stochastics, and biophotonics. He has served as an Editor and/or Guest Editor for a number of journals, including the IEEE JOURNAL OF SOLID-STATE CIRCUITS, PROCEEDINGS OF THE IEEE, the IEEE PHOTONICS JOURNAL, *PLOSONE*, and is currently on the editorial boards of Nature's *Scientific Reports*, IEEE ACCESS, *Royal Society Open Science*, and *Frontiers in Physics.*