ADMM-based Distributed Algorithm for Economic Dispatch in Power Systems With Both Packet Drops and Communication Delays

Qing Yang, Gang Chen, *Member, IEEE*, and Ting Wang

Abstract—By virtue of alternating direction method of multipliers (ADMM), Newton-Raphson method, ratio consensus approach and running sum method, two distributed iterative strategies are presented in this paper to address the economic dispatch problem (EDP) in power systems. Different from most of the existing distributed ED approaches which neglect the effects of packet drops or/and time delays, this paper takes into account both packet drops and time delays which frequently occur in communication networks. Moreover, directed and possibly unbalanced graphs are considered in our algorithms, over which many distributed approaches fail to converge. Furthermore, the proposed schemes can address the EDP with local constraints of generators and nonquadratic convex cost functions, not just quadratic ones required in some existing ED approaches. Both theoretical analyses and simulation studies are provided to demonstrate the effectiveness of the proposed schemes.

Index Terms—Alternating direction method of multipliers (ADMM), average consensus, directed graph (digraph), distributed algorithm, economic dispatch, packet drops, time delays.

I. INTRODUCTION

AS a key concern in power systems, the economic dispatch problem (EDP) has attracted considerable research interests during the past few decades, which is to achieve optimal power allocation with minimal cost while satisfying both supply-demand balance constraint and local constraints of generators. Since the EDP is essentially an optimization problem, many conventional centralized optimization schemes, such as lambda iteration method [1], [2], genetic algorithm [3], and particle swarm optimization [4], [5], have been applied to solve it. However, these centralized approaches need a single control center to access the system-wide aggregated information, and thus they may be subject to some performance limitations, such as intolerance of a single-point failure, requirement of high-level of connectivity, and lack of privacy

The authors are with the State Key Laboratory of Power Transmission Equipment and System Security and New Technology, and the School of Automation, Chongqing University, Chongqing 400044, China (e-mail: yangqingjlu@163.com; chengang@cqu.edu.cn; 201713021019@cqu.edu.cn).

protection. Therefore, it is more desirable to design distributed algorithms to overcome these limitations.

Recently, many distributed approaches have been developed to address the EDP in power grids. Especially, a large number of consensus-based approaches (see [6]–[13] and references therein) have been proposed for solving the EDP in a distributed way. In [6], [7], distributed optimal power dispatch schemes are put forward without considering local constraints of generators. Based on incremental cost consensus, the work in [8] presents a distributed algorithm for the EDP with local constraints of generators. In [9], a distributed continuous-time approach is presented to address the EDP, which can obtain the optimal power generation in finite time. The work in [10] further studies the EDP with both transmission loss and communication failure. The work in [11] presents a consensus-based continuous-time approach to address the EDP, where the communication delays are taken into account. However, it is worth noting that the results in [6]–[11] only consider the EDP over undirected graphs rather than directed ones. Actually, directed graphs are more general and practical in network systems because of packet loss, equipment failure and asymmetric bandwidth restriction. There also exist some consensus-based results (see, e.g., [12], [13]) which are able to solve the EDP over directed graphs. However, all these approaches in [6]–[13] require local cost functions to be quadratic, which may fail to solve the EDP with nonquadratic cost functions. In recent years, some distributed approaches (see, e.g., [14]–[20]) have been presented to solve the EDP with general convex cost functions. The work in [19] proposes a distributed iterative scheme with packet drops and the work in [20] takes into account communication delays. However, there are few reports on EDP taking into account both packet drops and communication delays, even though they always occur simultaneously in network systems. Recently, some ADMM-based distributed approaches (see, e.g., [21], [22]) have also been proposed to address the EDP in power grids. However, so far, the ADMM-based distributed algorithms for solving the EDP with both packet drops and communication delays have not been well investigated.

Motivated by the above observation and with the aid of the ADMM [23], [24], Newton-Raphson method, the ratio consensus algorithm [25] and running sum method [26], this paper presents two distributed iterative algorithms to address the EDP over directed graphs without/with both packet drops and communication delays. In comparison with existing results on the EDP, the main features of this study are

concluded as follows:

1) Although there exist some ADMM-based distributed approaches (see, e.g., [21], [22]) for the EDP, to the best of our knowledge, this paper for the first time presents an ADMM-based distributed algorithm to solve the EDP with both packet drops and communication delays which are ignored in [6]–[22].

2) Different from most of the existing studies (see, e.g., [6]–[11]) on the EDP which require the communication graphs to be undirected, the proposed schemes can solve the EDP on general digraphs. Since general digraphs are considered, the symmetry property of graphs required in [6]–[11] is no longer necessary, but our algorithms still converge to the optimal solution.

3) Unlike the works in [6]–[13] requiring the cost functions to be quadratic, general convex cost functions are considered in this paper. In addition, different from the works in [6], [7] and [16] only considering the coupled equality constraint, both the coupled equality constraint and local constraints of generators are taken into account in this paper.

This paper is organized as follows. Some preliminaries are given in Section II and the ED problem is formulated in Section III. In Section IV, two distributed iterative algorithms for the EDP over digraphs are presented and analyzed. Simulation studies are given in Section V. Finally, concluding remarks are drawn in Section VI.

II. PRELIMINARIES

A. Communication Network Model

The communication network in smart grids is modeled as a general directed graph (digraph) $G = (V, E, Q)$ with $V = \{1, 2, \ldots, n\}$ being the set of nodes, $E \subseteq V \times V$ being the edge set, and $Q \in \mathbb{R}^{n \times n}$ being the weighted adjacency matrix of G. The notation $(i, j) \in E$ represents there exists a directed communication link from node i to node j. Meanwhile, j is said to be an out-neighbour of node i and i is called an in-neighbour of node j. Although each node always has access to its own information, for the sake of notational convenience, it is assumed that $(i, i) \notin E$ for all $i \in V$. A digraph is said to be strongly connected if for any pair of nodes (i, j), there exists a directed path from node i to node j, i.e., a sequence of directed edges of the form $(i, i_1), (i_1, i_2), \ldots, (i_l, j)$ with $i_r \in V$, $r = 1, \ldots, l$. Let $N_i^- = \{j \in V | (i, j) \in E\}$ denote the set of out-neighbors of node i. The cardinality of N_i^- is denoted as $d_i^- = |N_i^-|$, which is called the out-degree of node i. Similarly, let $N_i^+ = \{j \in V | (j, i) \in E\}$ denote the set of in-neighbors of node i and let $d_i^+ = |N_i^+|$ be the in-degree of node i. A digraph is called a balanced digraph, if $d_i^+ = d_i^-$. A nonnegative matrix $Q \in \mathbb{R}^{n \times n}$ is column stochastic if each column sums to 1. The matrix Q is primitive if there exists an integer $m > 0$ such that each entry of Q^m is positive.

B. Standard ADMM Algorithm

Consider the following minimization problem:

$$\min_{x, y} f_1(x) + f_2(y) \tag{1a}$$

$$\text{s.t.} \quad S_1 x + S_2 y = b \tag{1b}$$

where $x \in \mathbb{R}^{n_1}$ and $y \in \mathbb{R}^{n_2}$ are the decision variables; $S_1 \in \mathbb{R}^{q \times n_1}$ and $S_2 \in \mathbb{R}^{q \times n_2}$ are two constant matrices and $b \in \mathbb{R}^q$ is a constant vector.

According to [23], the standard ADMM for solving the problem (1) consists of the following iterations:

$$x^{t+1} = \arg\min_x L_\rho(x, y^t, z^t) \tag{2a}$$

$$y^{t+1} = \arg\min_y L_\rho(x^{t+1}, y, z^t) \tag{2b}$$

$$z^{t+1} = z^t + \rho(S_1 x^{t+1} + S_2 y^{t+1} - b) \tag{2c}$$

where $L_\rho(x, y, z)$ is the augmented Lagrangian function of the problem (1), defined by

$$L_\rho(x, y, z) = f_1(x) + f_2(y) + z^T(S_1 x + S_2 y - b)$$
$$+ (\rho/2) \|S_1 x + S_2 y - b\|_2^2$$

with $z \in \mathbb{R}^q$ being the Lagrange multiplier associated with the equality constraint (1b) and ρ being a positive penalty parameter.

Let η_1^t and η_2^t respectively denote the primal and dual residuals of the ADMM iterations (2) at step t, defined by

$$\eta_1^t = S_1 x^t + S_2 y^t - b, \text{ and } \eta_2^t = \rho S_1^T S_2(y^t - y^{t-1}).$$

The convergence properties of the ADMM iterations (2) are described in the following lemma.

Lemma 1 [23], [24]: Assume that $\rho > 0$ and the following conditions hold:

1) The functions $f_1(x) : \mathbb{R}^{n_1} \to \mathbb{R} \cup \{+\infty\}$ and $f_2(y) : \mathbb{R}^{n_2} \to \mathbb{R} \cup \{+\infty\}$ are proper, closed and convex;

2) The unaugmented Lagrangian function L has a saddle point;

3) The matrices S_1 and S_2 have full column ranks.

Then the ADMM algorithm (2) is asymptotically convergent to the optimal solution (x^*, y^*) and the optimal Lagrange multiplier z^* of the problem (1), with $\lim_{t \to \infty} \|\eta_1^t\|_2 = 0$, and $\lim_{t \to \infty} \|\eta_2^t\|_2 = 0$.

III. PROBLEM DESCRIPTION

The EDP investigated in this paper is to achieve optimal power scheduling through local information interaction and coordination among agents (generators) in the network while meeting the power balance constraint and local constraints of generators. To be specific, the EDP is formulated as the following constrained optimization problem:

$$\min_x f(x) = \sum_{i=1}^n f_i(x_i) \tag{3a}$$

$$\text{s.t.} \quad \sum_{i=1}^n x_i = d \tag{3b}$$

$$x_i \in \Theta_i, \ i = 1, \ldots, n \tag{3c}$$

where $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ with x_i being the power generation of generator i; $f_i(x_i) : \mathbb{R} \to \mathbb{R}_+$ and Θ_i are respectively the local cost function and local constraints of generator i; d represents the total load demand. The equality (3b) is the supply-

demand balance constraint. As a general problem description, all the local inequality constraints are described in the form of (3c). In other words, the set Θ_i is determined by the local inequality constraints of generators.

For the convenience of later analyses, we impose two assumptions on the problem (3) and one assumption on the communication graph G.

Assumption 1: For all $i \in \{1, \cdots, n\}$, the local cost function $f_i(x_i): \mathbb{R} \to \mathbb{R}_+$ is convex, closed, proper, and twice continuously differentiable. Moreover, the local constraint set Θ_i is closed and convex.

Assumption 2 (Slater's Condition): There exists $\bar{x}_i \in \text{int}(\Theta_i)$ for all $i \in \{1, \ldots, n\}$ such that $\sum_{i=1}^{n} \bar{x}_i = d$, where $\text{int}(\cdot)$ represents the set of interiors of a set.

Remark 1: Assumption 1 implies that the problem (3) is a convex optimization problem and Assumption 2 is used to guarantee the existence of feasible solutions of the problem (3). In addition, according to Assumption 1, the local cost function does not have to be quadratic, which generalizes the results in [6]–[13].

Assumption 3: The digraph $G = (V, E, Q)$ is strongly connected and the weighted adjacency matrix Q is primitive and column stochastic. Moreover, each node i has access to its own out-degree d_i^-.

Remark 2: According to Assumption 3, the communication graphs considered in this paper are general directed graphs rather than the undirected ones considered in [6]–[11]. Thus the symmetric property of communication graphs required in [6]–[11] is not needed here.

To apply the ADMM, a reformulation of (3) is considered. Define two convex sets: $\Gamma_1 = \{x \in \mathbb{R}^n | \sum_{i=1}^{n} x_i = d\}$, and $\Gamma_2 = \{x \in \mathbb{R}^n | x_i \in \Theta_i, i = 1, \ldots, n\}$. Let h_1 and h_2 be two indicator functions for the sets Γ_1 and Γ_2, respectively defined by

$$h_1(x) = \begin{cases} 0, & \text{if } x \in \Gamma_1 \\ +\infty, & \text{otherwise} \end{cases}$$

$$h_2(y) = \begin{cases} 0, & \text{if } y \in \Gamma_2 \\ +\infty, & \text{otherwise.} \end{cases}$$

Then, the problem (3) can be rewritten as

$$\min_{x,y} \ f(x) + h_1(x) + h_2(y) \tag{4a}$$

$$\text{s.t.} \quad x - y = 0 \tag{4b}$$

where $x, y \in \mathbb{R}^n$ are decision variables.

The objective of this paper is to present distributed iterative strategies to solve the ED problem (3) (or (4)) over directed graphs even in the presence of both packet drops and communication delays.

IV. ADMM-BASED DISTRIBUTED ALGORITHMS ON DIRECTED GRAPHS

In this section, two ADMM-based distributed algorithms are proposed to solve the problem (4) over directed graphs, one for reliable communication networks and the other for unreliable communication networks with both packet drops and communication delays.

A. Distributed Algorithm with Reliable Communication Networks

By applying the ADMM [23] to the problem (4), one has

$$x^{t+1} = \arg\min_{x} L_\rho\left(x, y^t, z^t\right) \tag{5a}$$

$$y^{t+1} = \arg\min_{y} L_\rho\left(x^{t+1}, y, z^t\right) \tag{5b}$$

$$z^{t+1} = z^t + \rho\left(x^{t+1} - y^{t+1}\right) \tag{5c}$$

where $L_\rho(x, y, z)$ is the augmented Lagrangian function for the problem (4) and is given by

$$L_\rho(x, y, z) = f(x) + h_1(x) + h_2(y) + z^T(x - y) + (\rho/2)\|x - y\|_2^2. \tag{6}$$

Note that the update of z, i.e., (5c) is decentralized and thus we only need to solve the subproblems (5a) and (5b) in a distributed way to obtain the updated values of variables x and y.

First, consider the subproblem (5a). By combining (5a) with (6), one has

$$\begin{aligned} x^{t+1} &= \arg\min_{x}(f(x) + h_1(x) + h_2(y^t) + (z^t)^T(x - y^t) \\ &\quad + \frac{\rho}{2}\|x - y^t\|_2^2) \\ &= \arg\min_{x}(f(x) + h_1(x) + (z^t)^T(x - y^t) \\ &\quad + \frac{\rho}{2}\|x - y^t\|_2^2) \\ &= \arg\min_{x}(f(x) + h_1(x) + (z^t)^T(x - y^t) \\ &\quad + \frac{\rho}{2}\|x - y^t\|_2^2 + \frac{1}{2\rho}\|z^t\|_2^2) \\ &= \arg\min_{x}\left(f(x) + h_1(x) + \frac{\rho}{2}\left\|x - y^t + \frac{1}{\rho}z^t\right\|_2^2\right) \\ &= \arg\min_{x \in \Gamma_1}\left(f(x) + \frac{\rho}{2}\left\|x - y^t + \frac{1}{\rho}z^t\right\|_2^2\right) \end{aligned} \tag{7}$$

where the second and third equalities have used the fact that $h_2(y^t)$ and $1/(2\rho)\|z^t\|_2^2$ are both constants with respect to the subproblem (5a), and the last equality follows from the definition of $h_1(x)$.

Note that the problem (7) is equivalent to the following constraint problem:

$$\min_{x} \ \bar{f}(x) = \sum_{i=1}^{n} \bar{f}_i(x_i) \tag{8a}$$

$$\text{s.t.} \quad 1_n^T x = d \tag{8b}$$

where $\bar{f}(x) = f(x) + (\rho/2)\left\|x - y^t + (1/\rho)z^t\right\|_2^2$, $\bar{f}_i(x_i) = f_i(x_i) + (\rho/2)(x_i - y_i^t + (1/\rho)z_i^t)^2$, and $1_n = [1, \ldots, 1]^T \in \mathbb{R}^n$.

The Lagrangian function of the problem (8) is $\bar{L}(x, \zeta) = \sum_{i=1}^{n} \bar{f}_i(x_i) + \zeta(d - \sum_{i=1}^{n} x_i)$. It is worth pointing out that $\bar{f}_i(x_i)$ is strongly convex since $\rho > 0$ and $f_i(x_i)$ is convex from Assumption 1. Thus, the solution to the problem (8) is unique and the stationary point of the Lagrangian function $\bar{L}$ is exactly the solution to the problem (8). By virtue of the Newton-Raphson approach, the stationary point of $\bar{L}$ can be obtained by solving the following equation:

$$G_\lambda[\triangle x_1,\ldots,\triangle x_n,\triangle \zeta]^T = \nabla \bar{L}|_\lambda \tag{9}$$

where $\lambda = [x_1,\ldots,x_n,\zeta]^T$, $\nabla \bar{L}|_\lambda$ and G_λ are respectively the gradient and the Hessian matrix of $\bar{L}$ calculated at the point λ. According to (9) and Assumption 1, one has

$$\begin{cases} \triangle x_i = \dfrac{\dfrac{d\bar{f}_i}{dx_i}|_\lambda + \triangle \zeta - \zeta}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda}, \quad i = 1,\ldots,n \\[3mm] -\sum\limits_{i=1}^{n} \triangle x_i = d - \sum\limits_{i=1}^{n} x_i = \sum\limits_{i=1}^{n}(d_i - x_i) \end{cases} \tag{10}$$

where d_i, $i = 1,\ldots,n$, is regarded as a virtual local demand satisfying $\sum_{i=1}^{n} d_i = d$. From (10), it can be obtained that

$$\sum_{i=1}^{n} \dfrac{\dfrac{d\bar{f}_i}{dx_i}|_\lambda + \triangle \zeta - \zeta}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda} = \sum_{i=1}^{n}(x_i - d_i). \tag{11}$$

Furthermore, in light of (11), $\zeta - \triangle \zeta$ can be computed by

$$\zeta - \triangle \zeta = \dfrac{\sum\limits_{i=1}^{n}\left(\dfrac{\dfrac{d\bar{f}_i}{dx_i}|_\lambda}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda} + d_i - x_i\right)}{\sum\limits_{i=1}^{n}\left(\dfrac{1}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda}\right)}. \tag{12}$$

It is worth noting that a control center is needed to gather system-wide information if $\zeta - \triangle \zeta$ is computed directly by (12). To avoid the usage of any control center, in the following, we devote to calculate $\zeta - \triangle \zeta$ in a distributed fashion. For each node i, define two auxiliary variables $u_i[k]$ and $w_i[k]$, respectively initialized as

$$\begin{cases} u_i[0] = \dfrac{\dfrac{d\bar{f}_i}{dx_i}|_\lambda}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda} + d_i - x_i \\[4mm] w_i[0] = \dfrac{1}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda} \end{cases}, \quad i = 1,\ldots,n. \tag{13}$$

Consider the ratio consensus algorithm [25] used to solve the average consensus problem on digraphs, in which the state variables are updated according to

$$\varphi_i[k+1] = \sum_{j\in N_i^+ \cup \{i\}} q_{ji}\varphi_j[k] \tag{14a}$$

$$\psi_i[k+1] = \sum_{j\in N_i^+ \cup \{i\}} q_{ji}\psi_j[k] \tag{14b}$$

where $q_{ji} = 1/(1+d_j^-)$ for $j \in N_i^+ \cup \{i\}$ and zeros otherwise. The initial conditions of iterations (14) are given as:

$\varphi[0] = [\varphi_1[0],\ldots,\varphi_n[0]]^T$ and $\psi[0] = 1_n$. The convergence of the consensus algorithm (14) is given in the following lemma.

Lemma 2 [25]: Let $\{\varphi_i[k],\psi_i[k]\}$ be the sequence generated by (14). Then under Assumption 3, the average consensus value can be asymptotically obtained via

$$\lim_{k\to\infty} \frac{\varphi_i[k]}{\psi_i[k]} = \frac{\sum\limits_{i=1}^{n}\varphi_i[0]}{n}, \quad i = 1,\ldots,n. \tag{15}$$

Now, let $\varphi_i[0] = u_i[0]$ for all $i \in \{1,\ldots,n\}$. By performing the consensus algorithm in (14) and in light of Lemma 2, the average consensus value for the variables $u_i[k]$ can be asymptotically obtained via (15), i.e., $\varphi_i[k]/\psi_i[k] \to u^* = \frac{1}{n}\sum_{i=1}^{n}(\frac{d\bar{f}_i}{dx_i}|_\lambda/\frac{d^2 \bar{f}_i}{dx_i^2}|_\lambda + d_i - x_i)$ as $k \to \infty$. Similarly, by letting $\varphi_i[0] = w_i[0]$, the average consensus value for the variables $w_i[k]$ is also asymptotically obtained via (15), which is denoted by w^*, i.e., $w^* = \frac{1}{n}\sum_{i=1}^{n}(1/\frac{d^2 \bar{f}_i}{dx_i^2}|_\lambda)$. By combining these with (12), at each node $\zeta - \triangle \zeta$ can be calculated by

$$\zeta - \triangle \zeta = \frac{u^*}{w^*} \tag{16}$$

in a distributed fashion. Moreover, by substituting (16) into the first equality in (10), $\triangle x_i$ can be calculated via

$$\triangle x_i = \frac{\dfrac{d\bar{f}_i}{dx_i}|_\lambda - (\zeta - \triangle \zeta)}{\dfrac{d^2 \bar{f}_i}{dx_i^2}|_\lambda}. \tag{17}$$

To this end, the optimal solution x_i^*, $i = 1,\ldots,n$, to the problem (8) can be computed by

$$x_i^* = x_i - \triangle x_i. \tag{18}$$

In other words, according to (16)–(18), the optimal solution x^* to the problem (8), i.e., the solution x^{t+1} to the subproblem (5a) is computed in a distributed way.

In the following, combining (5b) with (6) yields

$$\begin{aligned} y^{t+1} &= \arg\min_y (f(x^{t+1}) + h_1(x^{t+1}) \\ &\quad + h_2(y) + (z^t)^T(x^{t+1} - y) + \frac{\rho}{2}\|x^{t+1} - y\|_2^2) \\ &= \arg\min_y \left(h_2(y) + \frac{\rho}{2}\left\|x^{t+1} - y + \frac{1}{\rho}z^t\right\|_2^2\right) \\ &= \arg\min_{y\in\Gamma_2} \frac{\rho}{2}\left\|x^{t+1} - y + \frac{1}{\rho}z^t\right\|_2^2 \\ &= P_{\Gamma_2}\left(x^{t+1} + \frac{1}{\rho}z^t\right) \end{aligned}$$

where P_{Γ_2} represents the projection operator onto the convex set Γ_2. To be specific, let $\breve{\vartheta}_i^{t+1} = x_i^{t+1} + (1/\rho)z_i^t$, $i = 1,\ldots,n$ and the solution to the subproblem (5b) can be given as:

$$y_i^{t+1} = \begin{cases} \breve{\vartheta}_i^{t+1}, & \breve{\vartheta}_i^{t+1} \in \Theta_i \\ \breve{\xi}_i^{t+1}, & \breve{\vartheta}_i^{t+1} \notin \Theta_i \end{cases} \tag{19}$$

with $\breve{\xi}_i^{t+1} \in \partial\Theta_i$ and $\partial\Theta_i$ being the set of boundary points of Θ_i, $i = 1,\ldots,n$. By substituting x^{t+1} and y^{t+1} into (5c), it fol-

lows that

$$z_i^{t+1} = z_i^t + \rho(x_i^{t+1} - y_i^{t+1}), \; i = 1, \ldots, n. \qquad (20)$$

Based on the above discussion, the proposed distributed algorithm to solve the EDP (3) or (4) on digraphs with reliable communication networks is summarized in Algorithm 1. The convergence of Algorithm 1 is described in the following theorem.

Algorithm 1 ADMM-based distributed algorithm for the EDP (3) on digraphs with reliable communication networks

---

Input: The digraph $G = (V, E, Q)$ with the edge weight q_{ij} defined in (14); ρ, x_i^0, y_i^0, z_i^0, $\forall i \in V$.
Output: The optimal solution x^*.
1: **for** $t = 0, 1, 2, \ldots$ **do**
2:      **for** $k = 0, 1, 2, \ldots$ **do**
3:          For each node, run the iterations (14) for the variable
            $u_i[k]$ with initial values $\varphi_i[0] = u_i[0]$ and $\psi_i[0] = 1$,
            $\forall i \in V$.
4:          Compute the average consensus value u^* by
$$u^* = \lim_{k \to \infty} \frac{\varphi_i[k]}{\psi_i[k]}.$$
5:          Compute the average consensus value w^* for the variable
            $w_i[k]$ with the same way.
6:      **end for**
7:      Compute x_i^{t+1} via (16)–(18), y_i^{t+1} via (19) and z_i^{t+1} by (20),
        $\forall i \in V$.
8:      **if** $|\eta_1^t(i)| \le \epsilon_1$ and $|\eta_2^t(i)| \le \epsilon_2$, $\forall i \in V$ **then**
9:          Let $x_i^* = x_i^{t+1}$ $(i = 1, \ldots, n)$ and break.
10:     **end if**
11: **end for**

Theorem 1: If Assumptions 1–3 and $\rho > 0$ hold, then the sequence $\{(x^t, y^t, z^t)\}$ generated by Algorithm 1 converges to (x^*, y^*, z^*) in a distributed manner, where (x^*, y^*) and z^* are, respectively, the optimal solution and the optimal Lagrange multiplier of the problem (4), with the primal residual $\eta_1^t = x^t - y^t$ and dual residual $\eta_2^t = -\rho(y^t - y^{t-1})$ converging to zero, i.e., $\lim_{t \to \infty} \left\| \eta_1^t \right\|_2 = 0$ and $\lim_{t \to \infty} \left\| \eta_2^t \right\|_2 = 0$.

Proof: First, we show that for each node, the iteration values x_i^{t+1}, y_i^{t+1} and z_i^{t+1} in (5) can be calculated in a distributed manner via Algorithm 1. According to Assumption 3, the digraph G is strongly connected and its weighted adjacent matrix Q is primitive and column stochastic. Carry out the consensus iterations in (14) for the variables $u_i[k]$ and $w_i[k]$ respectively. Then based on Lemma 2, we can obtain the average consensus values u^* and w^* in a distributed way. Furthermore, in light of (16)–(18), the optimal solution x^{t+1} to the subproblem (5a) is calculated in a distributed manner. Finally, the optimal solution y^{t+1} to the subproblem (5b) can be distributively obtained according to (19) and z^{t+1} in (5c) can be computed by (20) in a decentralized way. In a word, according to Algorithm 1, x_i^{t+1}, y_i^{t+1} and z_i^{t+1} in (5) can be calculated in a distributed fashion.

In the following, we show that if $\rho > 0$, the sequence $\{(x^t, y^t, z^t)\}$ generated by the ADMM iterations (5) is asymptotically convergent to (x^*, y^*, z^*) with (x^*, y^*) being the

optimal solution to the problem (4). First, note that the cost function $f(x) + h_1(x) + h_2(y)$ in the problem (4) is proper, closed, and convex because $f(x)$ is convex from Assumption 1 and the indicator functions $h_1(x)$, $h_2(y)$ of the convex sets are all proper, closed, and convex. Moreover, the equality constraint in (4b) is affine. From Assumptions 1 and 2, the constraint set $\Theta_1 \times \cdots \times \Theta_n$ is a closed convex set and the Slater's Condition holds. Thus, the strong duality property holds [27] and the optimal solution to the problem (4) exists. In light of the saddle point theorem [28], the existence of the optimal solution (x^*, y^*) to the problem (4) implies that there exists a dual optimal solution z^* such that (x^*, y^*, z^*) is a saddle point of the Lagrangian function of the problem (4). Furthermore, note that S_1 and S_2 in the problem (1) respectively correspond to I and $-I$ in the problem (4) which have full column ranks. Therefore, the three conditions required in Lemma 1 are all satisfied. Thus, the remainder of the convergence proof is similar to that in [24] only by replacing $f_1(x)$, $f_2(y)$, S_1, S_2 and b in (1) with $f(x) + h_1(x)$, $h_2(y)$, I, $-I$ and 0 in the problem (4), respectively. ∎

Remark 3: In Algorithm 1, the outer loop iterations are ADMM iterations and the inner loop iterations are average consensus iterations. In addition, it is worth pointing out that the inner loop iterations in Algorithm 1 are only asymptotically convergent to the average consensus values. Therefore, the stopping criteria (threshold or number of iterations) for the inner loop iterations are needed in actual implementation.

B. Distributed Algorithm With Both Packet Drops and Time Delays

While both packet drops and bounded communication delays are considered, an ADMM-based distributed algorithm for solving the problem (4) on digraphs is presented in this subsection.

First, some notations about packet drops and time delays are introduced. For a digraph $G = (V, E, Q)$, let $p_{ji}[k]$ be the indicator variables to denote whether there exist packet drops over $(j, i) \in E$ or not, which is defined by: $p_{ji}[k] = 0$ if there exist packet drops over $(j, i) \in E$ at time instant k and $p_{ji}[k] = 1$ otherwise. Let the integer $\tau_{ji}[k] \ge 0$ represent the delay of a message sent from node j to node i at time instant k and it is required that $0 \le \tau_{ji}[k] \le \bar{\tau}_{ji} \le \bar{\tau}$ for all $k \ge 0$, where $\bar{\tau} = \max\{\bar{\tau}_{ji}\}$. The following assumption on packet drops and time delays can be found in [25], [26].

Assumption 4: For packet drops, we assume that the packet drop probability pd_{ji} is strictly less than one, i.e., $pd_{ji} < 1$. Let $p[k]$ be a random binary vector including all the variables in $\{p_{ji}[k] | (j, i) \in E\}$. Then $p[0], p[1], \ldots$ are independent and identically distributed (i.i.d.), but at time instant k, any two variables contained in $p[k]$ may be of dependency. Moreover, $p_{ii}[k] = 1$ and $pd_{ii} = 0$ for all $i \in V$ (i.e., the own value of a node is always available without packet drop). For time delays, we assume that the delay τ_{ji} on each link $(j, i) \in E$ is time-invariant. There exists a finite $\bar{\tau}$ that uniformly bounds the delays, i.e., $\tau_{ji} \le \bar{\tau} < \infty$ for all the links $(j, i) \in E$ with time delays. In addition, $\tau_{ii} = 0$ for all $i \in V$ (i.e., the own value of a node is always available without time delay).

To solve the EDP (4) over digraphs with both packet drops and time delays, the running-sum method [26] and the ratio consensus algorithm with time delays [25] are employed here to compute the average consensus values. In particular, each node i, $i \in V$, is assigned with six variables. Besides the variables $\varphi_i[k]$ and $\psi_i[k]$, node i also maintains the variables $r_i[k]$, $s_i[k]$, $\pi_{ji}[k]$, and $\theta_{ji}[k]$ for $j \in N_i^+ \cup \{i\}$, where $r_i[k]$ and $s_i[k]$ are respectively the running sums of the variables $\varphi_i[k]$ and $\psi_i[k]$; $\pi_{ji}[k]$ and $\theta_{ji}[k]$ keep track of the running sum $r_i[k]$ and $s_i[k]$ respectively. Let $r_i[0] = 0$, $s_i[0] = 0$ for all $i \in V$ and $\pi_{ji}[0] = 0$, $\theta_{ji}[0] = 0$ for all $j \in N_i^+ \cup \{i\}$ and all $i \in V$. Then each node i, $i \in V$, computes the running sums $r_i[k+1]$ and $s_i[k+1]$ by

$$r_i[k+1] = r_i[k] + q_{il}\varphi_i[k] \tag{21a}$$

$$s_i[k+1] = s_i[k] + q_{il}\psi_i[k] \tag{21b}$$

where $k = 0, 1, 2, \ldots$; $q_{il} = 1/(1 + d_i^-)$ for $l \in N_i^- \cup \{i\}$ and zeros otherwise. Based on Assumption 4, the tracking variables $\pi_{ji}[k+1]$ and $\theta_{ji}[k+1]$ are computed as

$$\pi_{ji}[k+1] = \begin{cases} r_j[k+1], & p_{ji}[k] = 1 \text{ or } j = i \\ \pi_{ji}[k], & p_{ji}[k] = 0 \end{cases} \tag{22a}$$

$$\theta_{ji}[k+1] = \begin{cases} s_j[k+1], & p_{ji}[k] = 1 \text{ or } j = i \\ \theta_{ji}[k], & p_{ji}[k] = 0. \end{cases} \tag{22b}$$

According to (22), π_{ji} and θ_{ji} remain unchanged if there exist packet drops on link $(j,i) \in E$. Moreover, each node i always knows the running sum of itself, i.e., $\pi_{ii}[k+1] = r_i[k+1]$ and $\theta_{ii}[k+1] = s_i[k+1]$ for all $i \in V$. Let τ_{ji} be the delay on link $(j,i) \in E$. The variables $\varphi_i[k]$ and $\psi_i[k]$ of each node are updated according to:

$$\varphi_i[k+1] = \sum_{j \in N_i^+ \cup \{i\}} \left(\pi_{ji}\left[k+1-\tau_{ji}\right] - \pi_{ji}\left[k-\tau_{ji}\right] \right) \tag{23a}$$

$$\psi_i[k+1] = \sum_{j \in N_i^+ \cup \{i\}} \left(\theta_{ji}\left[k+1-\tau_{ji}\right] - \theta_{ji}\left[k-\tau_{ji}\right] \right) \tag{23b}$$

where $\varphi[0] = [\varphi_1[0], \ldots, \varphi_n[0]]^T$ and $\psi[0] = 1_n$. Note that $\varphi_i[k]$ and $\psi_i[k]$ are set as zero if $k < 0$. Then the convergence of the iterations (21)–(23) is described in the following lemma.

Lemma 3: Let $\{\varphi_i[k], \psi_i[k]\}$ be the sequence generated by iterations (21)–(23). Then under Assumptions 3 and 4, the average consensus value can be asymptotically obtained via

$$\lim_{k \to \infty} \frac{\varphi_i[k]}{\psi_i[k]} = \frac{\sum_{i=1}^n \varphi_i[0]}{n}, \quad i = 1, \ldots, n. \tag{24}$$

Proof: First, we show that with the aid of the augmented graph representation method used in [25] and [26], the directed graph $G = (V, E, Q)$ with both packet drops and bounded time delays can be converted to an augmented directed graph without packet drop and time delay. "Virtual buffers" and "virtual nodes" are introduced here to model the packet drops and the bounded time delays, respectively. To be specific, for each link $(j,i) \in E$, we introduce a "virtual buffer" b_{ji}, which stores the information that may be lost due to packet drops over (j,i). The information stored in b_{ji} will

be released to node i if there is no packet drop on link $(j,i) \in E$. That is to say, if $p_{ji}[k] = 1$, both $q_{ji}\varphi_j[k]$ ($q_{ji}\psi_j[k]$) and the information stored in b_{ji} will be sent to node i; when $p_{ji}[k] = 0$, no information is received at node i, but the information from node j will be accumulated in b_{ji}. On the other hand, for each node $i \in V$, we introduce $\bar{\tau}$ "virtual nodes": $i^{(1)}, i^{(2)}, \ldots, i^{(\bar{\tau})}$. At each time step k, the virtual node $i^{(l)}$ ($l = 1, \ldots, \bar{\tau}$) possesses the information which is ready to reach node i after l steps. The weights between the original nodes, "virtual buffers" and "virtual nodes" are shown in Fig. 1.
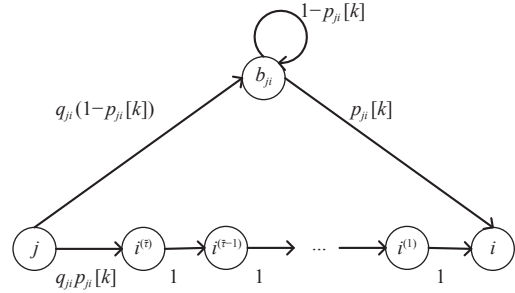

Fig. 1. "Virtual buffer" b_{ji} and "virtual node" $i^{(1)}, \ldots, i^{(\bar{\tau})}$ on link (j,i) and corresponding weights.

Let $G_1 = (V_1, E_1, \tilde{Q}[k])$ denote the augmented graph described above. It is easy to see that the augmented graph G_1 has $n_1 = |V_1| = n + m + n\bar{\tau}$ nodes, which are labeled in the following order: n original nodes from G, m "virtual buffers", n "virtual nodes" with one time delay, n "virtual nodes" with two time delays and so on. Moreover, the augmented graph G_1 has $m_1 = |E_1|$ edges. For each "virtual buffer" b_{ji}, define two variables φ_{pdi} and ψ_{pdi}. Let $\varphi_{pd}[k]$ and $\psi_{pd}[k]$ be the column vectors with each entry being $\varphi_{pdi}[k]$ and $\psi_{pdi}[k]$, respectively, for all $i = 1, \ldots, m$. For each "virtual node" $i^{(l)}$, define two variables $\varphi_i^{(l)}$ and $\psi_i^{(l)}$, and let $\varphi^{(l)}[k]$ and $\psi^{(l)}[k]$ be the column vectors with each entry being $\varphi_i^{(l)}[k]$ and $\psi_i^{(l)}[k]$, respectively, for all $i = 1, \ldots, n$ and $l = 1, \ldots, \bar{\tau}$. Moreover, let

$$\tilde{\varphi}[k] = \left[\varphi^T[k], \varphi_{pd}^T[k], \left(\varphi^{(1)}[k]\right)^T, \ldots, \left(\varphi^{(\bar{\tau})}[k]\right)^T \right]^T$$

and

$$\tilde{\psi}[k] = \left[\psi^T[k], \psi_{pd}^T[k], \left(\psi^{(1)}[k]\right)^T, \ldots, \left(\psi^{(\bar{\tau})}[k]\right)^T \right]^T.$$

In addition, the initial values for the variables $\tilde{\varphi}[k]$ and $\tilde{\psi}[k]$ are respectively defined as

$$\tilde{\varphi}_l[0] = \begin{cases} \varphi_l[0], & l \in V \\ 0, & l \notin V \end{cases} \quad \text{and} \quad \tilde{\psi}_l[0] = \begin{cases} 1, & l \in V \\ 0, & l \notin V \end{cases}$$

where $V = \{1, \ldots, n\}$ is the set of original nodes.

According to the above discussion, the iterations in (21)–(23) with both packet drops and time delays can be converted to the following compact form without packet drop and time delay:

$$\tilde{\varphi}[k+1] = \tilde{Q}[k]\tilde{\varphi}[k] \tag{25a}$$

$$\tilde{\psi}[k+1] = \tilde{Q}[k]\tilde{\psi}[k] \tag{25b}$$

where

$$\tilde{Q}[k] = \begin{bmatrix} Q^{(0)}[k] & Q_{pd1}[k] & I_{n \times n} & 0 & \cdots & 0 \\ Q_{pd2}[k] & Q_{pd3}[k] & 0 & 0 & \cdots & 0 \\ Q^{(1)}[k] & 0 & 0 & I_{n \times n} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ Q^{(\bar{\tau}-1)}[k] & 0 & 0 & 0 & \cdots & I_{n \times n} \\ Q^{(\bar{\tau})}[k] & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (26)$$

with $Q^{(l)}[k]$ $(l = 0, 1, \ldots, \bar{\tau})$ being the weighted matrix that depends on both packet drops and time delays and is defined as

$$Q^{(l)}[k](j,i) = \begin{cases} p_{ji}[k] Q(j,i), & \text{if } \tau_{ji} = l, (j,i) \in E \\ 0, & \text{otherwise.} \end{cases}$$

Note that for each $(j,i) \in E$, only one of $Q^{(0)}[k](j,i)$, $Q^{(1)}[k](j,i), \ldots, Q^{(\bar{\tau})}[k](j,i)$ is equal to $p_{ji}[k] Q(j,i)$ and the others are zero. $Q_{pd1}[k] = \{p_{ji}[k]\}_{n \times m}$ is a weighted matrix which depends on the weights of the links from "virtual buffers" to original nodes. $Q_{pd2}[k] = \{(1 - p_{ji}[k])q_{ji}\}_{m \times n}$ is a weighted matrix which relies on the weights of the links from original nodes to "virtual buffers". $Q_{pd3}[k] = \{1 - p_{ji}[k]\}_{m \times m}$ is the self-loop weighted matrix of the "virtual buffers".

In the following, we show that the average consensus value $\sum_{i=1}^{n} \varphi_i[0]/n$ can be asymptotically obtained via the iterations (21)–(23) or the iterations (25). According to Assumption 3 and the definition of $\tilde{Q}[k]$, it can be verified that $\tilde{Q}[k]$ is a primitive and column-stochastic matrix. Then the weak ergodicity of the matrix product $\tilde{Q}[k]\tilde{Q}[k-1]\cdots\tilde{Q}[1]\tilde{Q}[0]$ can be used to show the convergence of (25). Based on the augmented graph $G_1 = (V_1, E_1, \tilde{Q}[k])$ and in light of Assumption 4, the subsequent convergence proof of (25) is the same as that in [26]. Thus, by implementing the iterations (21)–(23) or the iterations (25), the average consensus value can be asymptotically obtained by (24) or by

$$\lim_{k \to \infty} \frac{\tilde{\varphi}_i[k]}{\tilde{\psi}_i[k]} = \frac{\sum_{i=1}^{n} \varphi_i[0]}{n}, i = 1, \ldots, n. \quad (27)$$

∎

As both packet drops and communication delays are considered here, we run the iterations (21)–(23) instead of the iterations (14) at Step 3 in Algorithm 1 and get the ADMM-based distributed algorithm with both packets drops and time delays, which is named Algorithm 2 in the following statement.

Since Algorithm 2 is in some extent similar to Algorithm 1, for brevity, we only need to make the following modifications on the basis of Algorithm 1 to get Algorithm 2: 1) Add the information of packet drops and delays in the "Input" step; 2) Replace the iterations (14) in Step 3 with the iterations (21)–(23).

The following theorem shows the convergence properties of Algorithm 2.

Theorem 2: If Assumptions 1–4 and $\rho > 0$ hold, then the sequence $\{(x^t, y^t, z^t)\}$ generated by Algorithm 2 converges to (x^*, y^*, z^*) in a distributed manner even when there exist both packet drops and bounded time delays, where (x^*, y^*) and z^* are, respectively, the optimal solution and the optimal Lagrange multiplier of the problem (4), with the primal residual $\eta_1^t = x^t - y^t$ and dual residual $\eta_2^t = -\rho(y^t - y^{t-1})$

converging to zero, i.e., $\lim_{t \to \infty} \|\eta_1^t\|_2 = 0$ and $\lim_{t \to \infty} \|\eta_2^t\|_2 = 0$.

Proof: We first show that when there exist both packet drops and bounded time delays, the iteration values x_i^{t+1}, y_i^{t+1} and z_i^{t+1} in (5) still can be calculated in a distributed fashion via Algorithm 2. Consider the variables $u_i[k]$ and $w_i[k]$ with initial values defined in (13). First, let $\varphi_i[0] = u_i[0]$ and $\psi_i[0] = 1$ for all $i \in V$. Then carry out the iterations (21)–(23) for the variable $u_i[k]$. According to Lemma 3, the average consensus value u^* for the variable $u_i[k]$ can be asymptotically obtained by (24). With a similar method, the average consensus value w^* for the variable $w_i[k]$ can also be obtained. That is to say, based on Lemma 3, the average consensus values for the variables $u_i[k]$ and $w_i[k]$ are both calculated in a distributed fashion even though packet drops and delays appear in the communication network simultaneously. Then according to (16)–(18), x_i^{t+1} is obtained in a distributed way and in light (19)–(20), y_i^{t+1} and z_i^{t+1} can also be computed in a distributed manner. The remaining proof is similar to that of Theorem 1. ∎

Remark 4: Different from the results in [6]–[10], [12]–[18], and [21] only concerning reliable communication networks and the results only considering communication delays [11], [20] or packet drops [19], this work takes into account both packet drops and communication delays. Though the result in [25] can deal with communication delays while the work in [26] can handle packet drops, they may fail to solve the issues considering both packet drops and communication delays. In fact, the coexistence of packet drops and communication delays may bring difficulties in analyses and execution of the algorithm. However, these difficulties are conquered here by adopting the consensus iterations (21)–(23), in the convergence proof of which a new augmented graph representation has been constructed to transform the problem with both packet drops and time delays into the one without packet drop and delay.

Remark 5: Note that Algorithm 2 is more general and practical than Algorithm 1 since it takes into account both packet drops and time delays which usually occur simultaneously in communication networks. But on the other hand, the coexistence of packet drops and time delays may make an effect on the convergence rate of Algorithm 2. That is to say, compared with Algorithm 1, Algorithm 2 needs more iteration steps to get the optimal solution of the problem (3).

Remark 6: For the case of time-varying delays, if each node in the network has knowledge of the upper bound $\bar{\tau}$ of the time delays and all the time-varying delays $\tau_{ji}[k](\tau_{ji}[k] \neq 0)$ are set as $\bar{\tau}$, then the case with time-varying delays is equivalent to the time-invariant delay case. Thus, for the case with bounded time-varying delays, the average consensus values for the variables $u_i[k]$ and $w_i[k]$ can also be obtained in a distributed manner via Algorithm 2 and the result in Theorem 2 still holds.

V. SIMULATION STUDY

In this section, several case studies are provided to verify the effectiveness of the proposed strategies for solving the EDP over directed graphs without/with packet drops and communication delays.

Case Study 1: The simulation results with Algorithm 1

In this case study, the proposed algorithm with reliable communication networks is verified on a power system with three generators and the local cost function $f_i(x_i)$ of generator i is given as

$$f_i(x_i) = \begin{cases} \gamma_i x_i^2 + \beta_i x_i + 360\exp\left(\dfrac{x_i + 30}{60}\right), & i = 1, 2 \\ \gamma_i x_i^2 + \beta_i x_i + 4 \times 10^{-6} x_i^4, & i = 3 \end{cases}$$

where the γ_i and β_i are the cost coefficients given in Table I. The local constraint set Θ_i $(i = 1, 2, 3)$ and the initial conditions are also provided in Table I. Let $\rho = 1$ and the total demand $d = 90$ MW. The local virtual demands are all set as 30 MW. The directed communication graph is shown in Fig. 2, where each node chooses its weight and the weights of its outgoing links to be $1/(1 + d_i^-)$. It is easy to verify that the weighted adjacency matrix Q corresponding to this digraph is primitive and column stochastic. The threshold values ϵ_1 and ϵ_2 in Algorithm 1 are both set as 0.001. The stopping criteria for the inner loop iterations of Algorithm 1 are given as: for all i, $|\varphi_i[k+1]/\psi_i[k+1] - \varphi_i[k]/\psi_i[k]| < \varepsilon_1$ with $\varphi_i[0] = u_i[0]$, and $|\varphi_i[k+1]/\psi_i[k+1] - \varphi_i[k]/\psi_i[k]| < \varepsilon_2$ with $\varphi_i[0] = w_i[0]$, where $\varepsilon_1 > 0, \varepsilon_2 > 0$ are the threshold values which are both set as 0.001 here.

TABLE I
PARAMETERS AND INITIAL CONDITIONS OF THREE GENERATORS

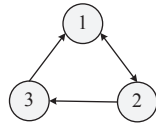| DG_i	γ_i	β_i	Θ_i (MW)	x_i^0 (MW)	y_i^0	z_i^0
1	0.085	4.95	[10, 50]	15	0	0
2	0.065	4.6	[10, 40]	15	0	0
3	0.32	0.72	[5, 20]	10	0	0

Fig. 2.   Communication graph of three generators.

By implementing Algorithm 1, the simulation results are shown in Figs. 3(a)–(f) and the number of iterations needed to reach the threshold is given in Table II. As can be seen, the incremental cost (IC) of each generator reaches consensus asymptotically and the consensus value is 27.722 $/MWh. The optimal solution to the problem (3) is $x^* = [33.038, 36.962, 20.000]^T$ MW, which satisfies the local constraint of each generator and the supply-demand balance constraint.

Case Study 2: The simulation results with Algorithm 2

In this subsection, both packet drops and fixed time delays are considered in the communication network shown in Fig. 2 and the other conditions are the same as those in Case Study 1. The packet drop probabilities on communication links are set as: $pd_{12} = 0.7$, $pd_{21} = 0.5$, $pd_{23} = 0.4$, $pd_{31} = 0.3$. The delay profile is given as: $\tau_{12} = \tau_{31} = \tau_{23} = 1$, $\tau_{21} = 2$. Note that the maximum time delay is $\bar{\tau} = 2$. By running Algorithm 2, the simulation results are illustrated in Figs. 4(a)–(f) and the number of iterations required to reach the threshold is
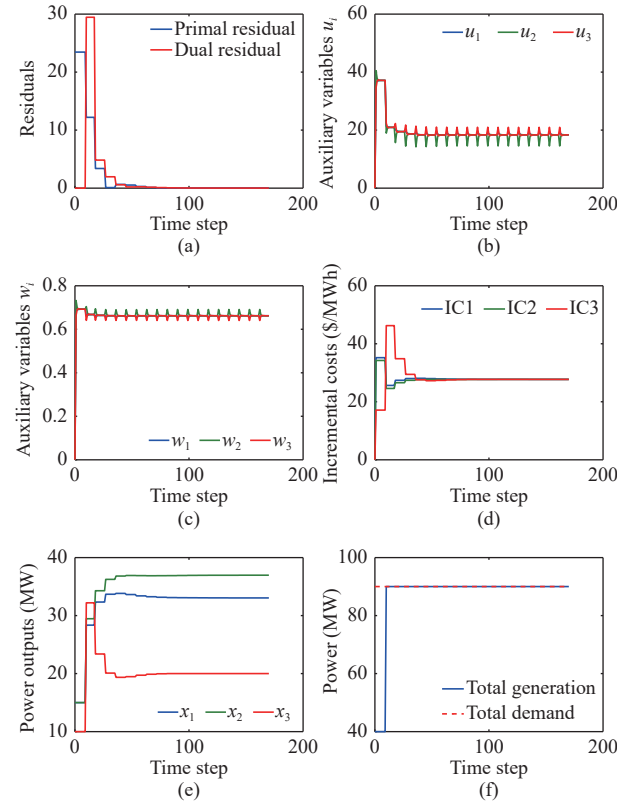

Fig. 3.   The simulation results of Case Study 1. (a) Primal and dual residuals; (b) Auxiliary variables u_i; (c) Auxiliary variables w_i; (d) Incremental cost ($/MWh); (e) Power outputs (MW); (f) Total outputs vs. demand (MW).

TABLE II
COMPARISON OF DIFFERENT COMMUNICATION CONDITIONS AND NUMBER OF NODES

	3-WO	3-W	14-WO	14-W
Iteration number of inner loop	9	32	191	416
Iteration number of outer loop	19	19	28	28
Total iteration number	171	608	5348	11 648

Notations: "WO" denotes the case without packet drops and time delays; "W" denotes the case with both packet drops and time delays.

provided in Table II. It can be seen that though there exist both packet drops and fixed time delays, the optimal incremental cost and the optimal solution still can be obtained and are the same as those in Case Study 1. Thus, Algorithm 2 is robust to both packet drops and fixed time delays.

Case Study 3: Test on IEEE 118-bus system with Algorithm 1

In this case study, the IEEE 118-bus system with 14 generators is utilized to verify the scalability of Algorithm 1 and quadratic cost functions considered in [6]–[13] are employed here. The generator parameters are adopted from [29]. The local virtual demands are all set as 1200/14 MW. That is to say, the total demand $d = 1200$ MW. The local constraint of generator i is set as $\Theta_i = [50, 200]$ MW, $i = 1, \ldots, 14$. We set $\rho = 1$, $x_i^0 = 1000/14$ MW, $y_i^0 = 0$ MW, $z_i^0 = 0$ and the stopping criteria are the same as those in Case Study 1. The directed communication graph is illustrated in Fig. 5.

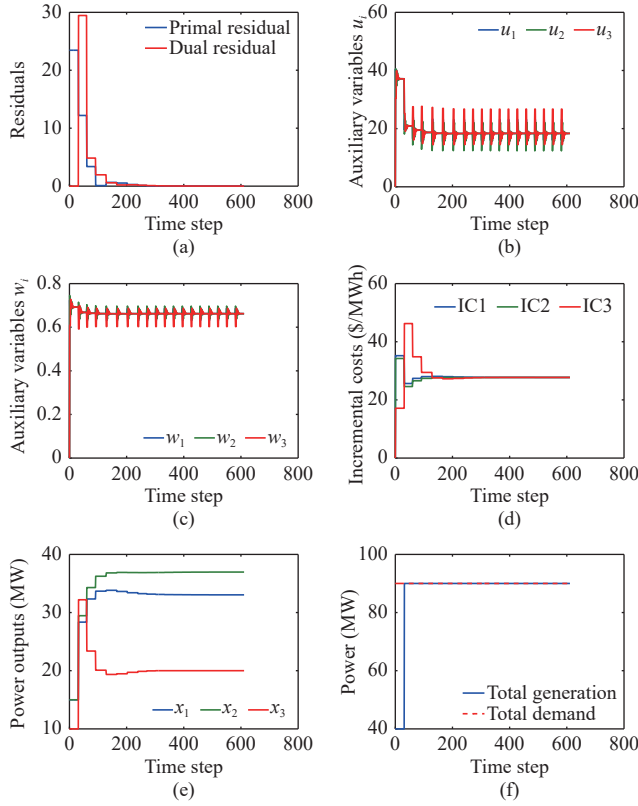By implementing Algorithm 1, the simulation results are

Fig. 4.   The simulation results of Case Study 2. (a) Primal and dual resid-uals; (b) Auxiliary variables u_i; (c) Auxiliary variables w_i; (d) Incremental cost ($/MWh); (e) Power outputs (MW); (f) Total generation vs. demand (MW).
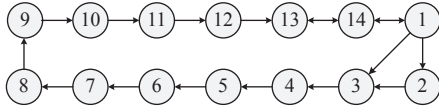

Fig. 5.   Communication graph of fourteen generators.

shown in Figs. 6(a)–(f). The number of iterations needed to reach the threshold is given in Table II. It can be seen that the consensus value of incremental costs is 347.2 $/MWh and the optimal solution is obtained as: $x^* = [158.2, 133.8, 50, 50, 50, 51.6, 50, 50, 102.2, 102.2, 102, 200, 50, 50]^T$ MW which meets the local constraint of each generator and supply-demand balance constraint. Hence, Algorithm 1 is still effective for a large network.

Case Study 4: Test on IEEE 118-bus system with Algorithm 2

In this subsection, both packet drops and time-varying delays are considered for the network shown in Fig. 5 and the other conditions are the same as those in Case Study 3. The packet drop probability on each communication link is randomly selected from the set [0.1, 0.9]. The upper bound of the time-varying delays is set as $\bar{\tau} = 2$. In particular, at each time step, the delays on communication links vary randomly in the set {1, 2}. According to Remark 6 and by running Algorithm 2, the simulation results are shown in Figs. 7(a)–(f) and the number of iterations needed to reach the threshold is provided in Table II. As can be seen, the optimal incremental cost and optimal solution are the same as those in Case Study
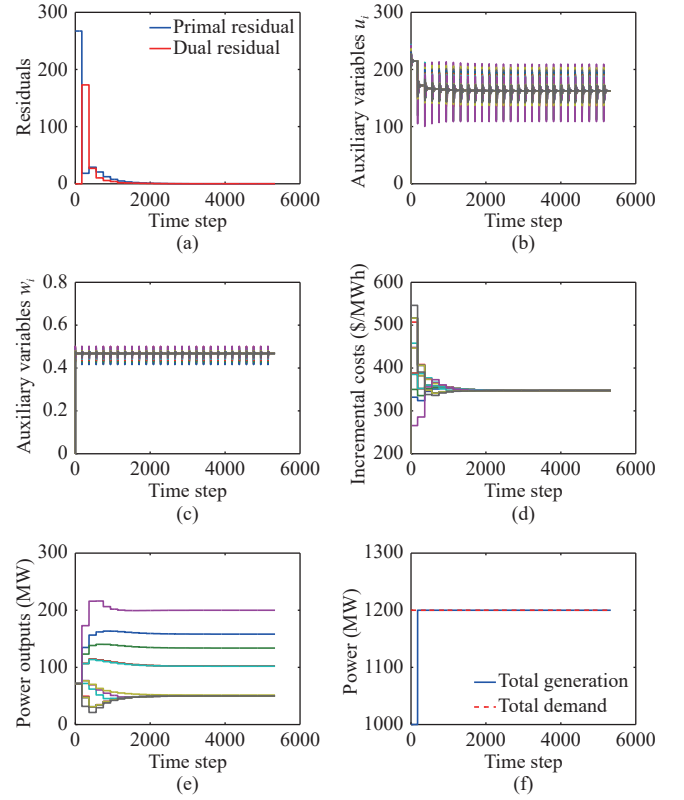

Fig. 6.   The simulation results of Case Study 3. (a) Primal and dual resid-uals; (b) Auxiliary variables u_i; (c) Average variables w_i; (d) Incremental cost ($/MWh); (e) Power outputs (MW); (f) Total outputs. vs. demand (MW).

3. Therefore, Algorithm 2 is effective for a large network with both packet drops and bounded time-varying delays.

Remark 7: According to Table II, it can be concluded that the convergence rate of Algorithm 1 is affected by the number of nodes. The more nodes, the more iteration steps are required. The convergence rate of Algorithm 2 is affected not only by the number of nodes but also by the packet drops and communication delays. That is to say, even if the number of nodes is the same, Algorithm 2 needs more iteration steps than Algorithm 1 to get the optimal solution, which confirms the statement in Remark 5. Moreover, from Table II, it can be seen that the iteration number of the outer loop is only associated with the number of nodes, not affected by communication conditions, but the iteration number of the inner loop is affected by both factors.

VI. CONCLUSIONS

In this paper, two ADMM-based distributed strategies have been proposed to address the EDP with general convex cost functions over possibly unbalanced digraphs. The communication networks without/with both packet drops and bounded time delays were considered and the ratio consensus method was employed to solve the EDP in a distributed fashion. It has been proved that the ADMM iterations in both the reliable communication networks case and the unreliable communication networks case (with both packet drops and time delays) are convergent to the optimal solution of the ED problem in a distributed way. Simulation results have demonstrated the effectiveness of the proposed schemes.
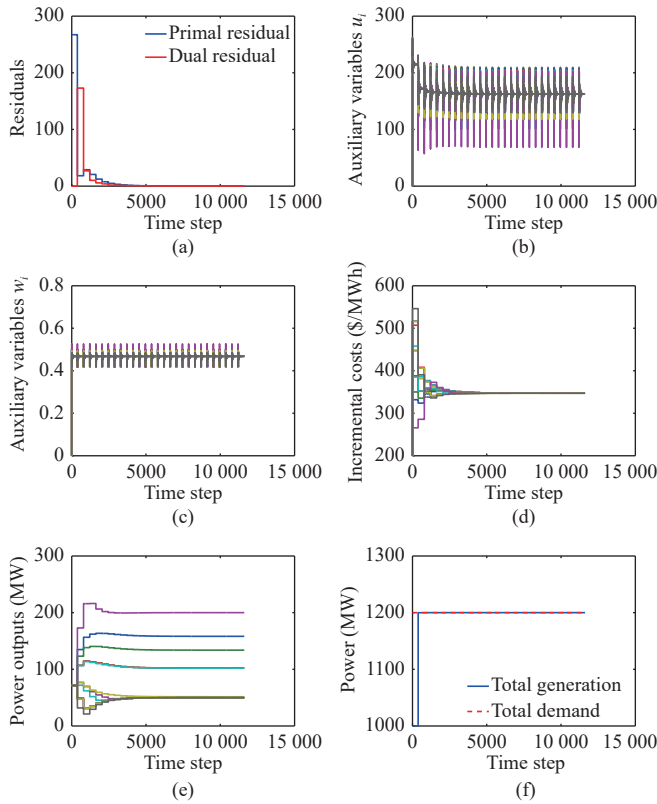
Fig. 7.   The simulation results of Case Study 4. (a) Primal and dual residuals; (b) Auxiliary variables u_i; (c) Auxiliary variables w_i; (d) Incremental cost ($/MWh); (e) Power outputs (MW); (f) Total generation vs. demand (MW).

REFERENCES

[1] C. E. Lin, S. T. Chen, and C. L. Huang, "A direct Newton-Raphson economic dispatch," *IEEE Trans. Power Syst.*, vol. 7, no. 3, pp. 1149–1154, Aug. 1992.

[2] J. Zhao, S. X. Liu, M. C. Zhou, X. W. Guo, and L. Qi, "Modified cuckoo search algorithm to solve economic power dispatch optimization problems," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 4, pp. 794–806, Jul. 2018.

[3] C.-L. Chiang, "Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels," *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 1690–1699, Nov. 2005.

[4] Z.-L. Gaing, "Particle swarm optimization to solving the economic dispatch considering the generator constraints," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1187–1195, Aug. 2003.

[5] Y. F. Tang, C. Luo, J. Yang and H. B. He, "A chance constrained optimal reserve scheduling approach for economic dispatch considering wind penetration," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 2, pp. 186–194, Apr. 2017.

[6] G. Binetti, M. Abouheaf, F. Lewis, D. Naso, and A. Davoudi, "Distributed solution for the economic dispatch problem," in *Proc. 21st MED*, 2013, pp. 243–250.

[7] G. Chen and E. N. Feng, "Distributed secondary control and optimal power sharing in microgrids," *IEEE/CAA J. Autom. Sinica*, vol. 2, no. 3,

pp. 304–312, 2015.

[8] Z. Zhang and M.-Y. Chow, "Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 1761–1768, Nov. 2012.

[9] G. Chen, J. Ren, and E. N. Feng, "Distributed finite-time economic dispatch of a network of energy resources," *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 822–832, Mar. 2017.

[10] Q. Li, D. W. Gao, H. Zhang, Z. Wu, and F.-Y. Wang, "Consensus-based distributed economic dispatch control method in power systems," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 941–954, Jan. 2019.

[11] G. Chen, and Z. Y. 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.

[12] S. P. Yang, S. C. Tan, and J. X. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4416–4426, Nov. 2013.

[13] T. Yang, D. Wu, Y. N. Sun, and J. M. Lian, "Minimum-time consensus-based approach for power system applications,," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1318–1328, Feb. 2016.

[14] F. H. Guo, C. Y. Wen, J. F. Mao, and Y.-D. Song, "Distributed economic dispatch for smart grids with random wind power," *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1572–1583, May 1572.

[15] H. Xing, Y. T. Mou, M. Y. Fu, and Z. Y. Lin, "Distributed bisection method for economic power dispatch in smart grid," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3024–3035, Nov. 2015.

[16] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *J. Optim. Theory Appl.*, vol. 129, no. 3, pp. 469–488, Jun. 2006.

[17] V. Loia, and A. Vaccaro, "Decentralized economic dispatch in smart grids by selforganizing dynamic agents," *IEEE Trans. Syst., Man, Cybern. A, Syst.*, vol. 44, no. 4, pp. 397–408, Apr. 2014.

[18] A. Cherukuri and J. Cortés, "Distributed generator coordination for initialization and anytime optimization in economic dispatch," *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 3, pp. 226–237, Sep. 2015.

[19] J. F. Wu, T. Yang, D. Wu, K. Kalsi, and K. H. Johansson, "Distributed optimal dispatch of distributed energy resources over lossy communication networks," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 3125–3137, Nov. 2017.

[20] T. Yang, J. Lu, D. Wu, J. F. Wu, G. D. Shi, Z. Y. Meng, and K. H. Johansson, "A distributed algorithm for economic dispatch over time-varying directed networks with delays," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5095–5106, Jun. 2017.

[21] G. Chen and Q. Yang, "An ADMM-based distributed algorithm for economic dispatch in islanded microgrids," *IEEE Trans. Ind. Inform.*, vol. 14, no. 9, pp. 3892–3903, Sep. 2018.

[22] B. A. Robbins and A. D. Dominguez-Garcia, "Optimal reactive power dispatch for voltage regulation in unbalanced distribution systems,"

IEEE Trans. Power Syst., vol. 31, no. 4, pp. 2903–2913, Jul. 2016.

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

[24] J. Mota, J. Xavier, P. Aguiar, and M. Pschel, "A proof of convergence for the alternating direction method of multipliers applied to polyhedral-constrained functions," *Mathematics*, 2011.

[25] C. N. Hadjicostis and T. Charalambous, "Average consensus in the presence of delays in directed graph topologies," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 763–768, Mar. 2014.

[26] C. N. Hadjicostis, N. H. Vaidya, and A. D. Dominguez-Garcia, "Robust distributed average consensus via exchange of running sums," *IEEE Trans. Autom. Control*, vol. 61, no. 6, pp. 1492–1507, Jun. 2016.

[27] S. Boyd, and L. Vandenberghe, Convex Optimization. Cambridge, UK: Cambridge University Press, 2004.

[28] D. Bertsekas, A. Nedić, and A. E. Ozdaglar, Convex Analysis and Optimization. Belmont, MA, USA: Athena Scientific, 2003.

[29] L. H. Wu, Y. N. Wang, X. F. Yuan, and S. W. Zhou, "Environmental/economic power dispatch problem using multi-objective differential evolution algorithm," *Elect. Power Syst. Res.*, vol. 80, no. 9, pp. 1171–1181, May 2010.
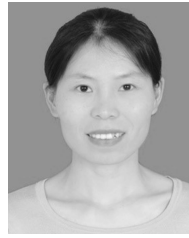
Qing Yang received the B.S. and M.S. degrees in mathematics from Jilin University, in 2008 and 2010, respectively. She is currently working toward the Ph.D. degree at the School of Automation, Chongqing University. Her research interests include distributed optimization, distributed algorithm, cooperative control, multiagent systems, and smart grids.
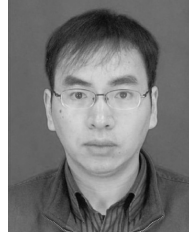
Gang Chen (M'16) received the Ph.D. degree in control engineering from Zhejiang University, in 2006. Since 2006, he has been with the School of Automation, Chongqing University, where he is currently a Professor. From 2009 to 2010, he was a Visiting Scholar at the Automation and Robotics Research Institute, University of Texas at Arlington, USA. His research interests include distributed control, cooperative control, intelligent control, distributed optimization, cyberphysical systems, smart grids, nonlinear control, and control applications.

Ting Wang received the B.S. degree in automation from Wuhan Polytechnic University, in 2017. She is currently working toward the M.S. degree at the School of Automation, Chongqing University. Her research interests include distributed control and optimization of microgrids.