2015 European Control Conference (ECC)
July 15-17, 2015. Linz, Austria

Distributed Algorithm for Economic Power Dispatch Including Transmission Losses

Hao Xing, Yuting Mou, Minyue Fu, and Zhiyun Lin

Abstract— **The economic dispatch problem (EDP) is of vital importance in the operation and control of power systems. In this paper, we consider the extended EDP that takes into account transmission losses and propose a distributed algorithm based on the average consensus algorithm and the bisection method. We assume that the total transmission losses can be represented by the *B matrix* loss formula, and the communication network between generators (nodes) is a connected undirected graph. A leader node is used to broadcast the total demand information by communicating with merely a small part of the nodes. Our algorithm is distributed in the sense that the nodes conduct local computations and communicate with their neighbors. Simulation results are given to show the performance of our algorithm.**

I. INTRODUCTION

Due to the decrease of petroleum reservoirs and the massive usage of petroleum in power generation nowadays, the efficient utilization of petroleum energy has raised a big concern in the community of power systems. To deal with this situation, the economic dispatch problem (EDP), which targets the minimum aggregate costs of power generation in a cooperative way, has been intensively investigated for the past decades. Many centralized algorithms have been developed to solve the EDP, e.g., the lambda iteration method, the binary search method, and the dynamic programming based algorithm [1].

Although many well-designed centralized algorithms are available nowadays, researchers are forced to develop distributed algorithms for the EDP by the unstoppable trend of smart grids. Future smart grid, which will likely incorporate numerous distributed generation systems, is a typical large scale system [2]. The widely spatial distribution of power generation systems adds extra difficulties in solving the EDP. Fortunately, for large scale systems, many distributed algorithms for control, estimation, and optimization have been proposed [3]. Compared with centralized algorithms, distributed algorithms exhibit the benefits including the feasibility in large scale systems, the reinforced robustness, and the evenly dispatched computation and communication burdens.

The following references are recent works on distributed algorithms for the EDP in smart grid. In [4] and [5], the authors propose the incremental cost consensus (ICC) algorithm to solve the EDP, where the average consensus algorithm is used to guarantee the balance between demand and supply. In [6], a distributed algorithm based on the ratio consensus is developed, which requires that the nodes have enough storage capacities for other nodes' parameters. A consensus + innovation approach is proposed in [7], where the consensus term ensures the commonly shared optimal incremental cost while the innovation term guarantees the balance of demand and supply. In [8], the authors propose a consensus based distributed algorithm, which enables the generators to collectively learn the mismatch between demand and total supply for feedback. Two fully Distributed algorithms for the EDP are also proposed in our previous works [9] and [10], respectively. The algorithm proposed in [9] deals with the EDP with quadratic cost functions on connected undirected graphs, and then is extended to deal with the EDP with general convex functions on strongly connected directed graphs in [10].

However, transmission losses are neglected and only the basic EDP is considered in [4]–[10]. Transmission losses, however, shall not be neglected in practical operations of power systems. If we schedule the generators according to the optimal solution to the basic EDP, the transmission losses in reality will cause an imbalance between demand and supply, resulting in frequency drop and possibly threatening the stability of power systems [11]. In the extended EDP, i.e., the EDP including transmission losses, the transmission losses will lead to the incremental losses, thus making the centralized solution to the extended EDP much more complicated than the solution to the basic EDP [1]. Since all the algorithms proposed in [4]–[10] can be viewed as various distributed implementations of the centralized solution for the basic EDP based on the Lagrange dual method, these algorithms can hardly be further adjusted for the extended EDP. However, few algorithms for the extended EDP have been proposed.

Forced by the urgent need for distributed algorithms to solve the extended EDP including transmission losses and continuing along our previous works [9] and [10], we propose a distributed algorithm in this paper. Our algorithm is based on the average consensus algorithm, and adopts the idea of bisection. A leader node is used to inform the generators of the aggregate demand information, which

H. Xing and Y. Mou are with the State Key Laboratory of Industrial Control Technology and the Department of Control Science and Engineering, Zhejiang University, 38 Zheda Road, Hangzhou, 310027 P. R. China. E-mail: haoxing@zju.edu.cn, ytmou@zju.edu.cn

M. Fu is with the School of Electrical Engineering and Computer Science, University of Newcastle, NSW 2308 Australia. He also holds a Qian-ren Professorship at the State Key Laboratory of Industrial Control Technology and the Department of Control Science and Engineering, Zhejiang University, China. E-mail: minyue.fu@newcastle.edu.au

Z. Lin is with the State Key Laboratory of Industrial Control Technology and the College of Electrical Engineering, Zhejiang University, 38 Zheda Road, Hangzhou, 310027 P. R. China. E-mail: linz@zju.edu.cn

This work was supported by Zhejiang Provincial Natural Science Foundation of China LR13F030002.

communicates with a small part of the generators only. Our algorithm is distributed in the sense that the nodes conduct local computations and communicate with their neighbors bidirectionally. The penalty factors and the total transmission losses are computed based on the average consensus algorithm in a distributed fashion, and the optimal incremental cost is determined using the bisection method locally at each node.

The rest of the paper is organized as follows. Some basics on graph theory, the average consensus algorithm, the problem formulation, and a centralized solution to the extended EDP are introduced in Section II. A distributed algorithm for the extended EDP is presented in Section III. In Section IV, numerical results are given to show the performance of our algorithm. We conclude our paper in Section V.

II. Preliminaries and Problem formulation

In this section, we first introduce some basics on graph theory and the average consensus algorithm on undirected graphs, and then give the problem formulation of the extended EDP. Finally we present a centralized solution for the extended EDP.

A. Basics of Graph Theory

An undirected graph $G = (V, E)$ consists of a non-empty finite set of nodes $V = \{1, 2, ..., n\}$ and a finite set of unordered edges $E \subseteq V \times V$. Let us denote the neighbor set of node $i \in V$ by $N_i = \{j \in V - \{i\} : (j, i) \in E\}$, which implies that node i can communicate with its neighbors bidirectionally. The degree of node i, denoted by $d_i = |N_i|$, is defined as the cardinality of N_i. Since G is undirected, for any i and j, $(i, j) \in E$ implies $(j, i) \in E$. An undirected graph is connected if there is a path from any node to any other node. The diameter of the connected undirected graph G, denoted by D_G, is the length of the longest among the shortest paths connecting any two nodes. Self-loops are included, i.e., $\forall i \in V,\ (i, i) \in E$. A non-negative matrix $Q \in \mathbb{R}^{n\times n}$ is associated with graph G, where $[Q]_{ij} > 0$ if and only if $(j, i) \in E$.

B. Average Consensus Algorithm

Let us consider the undirected graph $G = (V, E)$, where $V = \{1, \dots, n\}$. Each node $i \in V$ holds a state denoted by $x \in \mathbb{R}$. Denote by $x = [x_1, \dots, x_n]^T \in \mathbb{R}^n$ the aggregate state. Define the *Metropolis* weight matrix $Q \in \mathbb{R}^{n\times n}$ associated with graph G as

$$q_{ij} = \begin{cases} \dfrac{1}{\max(d_i, d_j) + 1}, & \text{if } j \in N_i, \\ 1 - \sum_{j\in N_i} q_{ij}, & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where q_{ij} is the entry of Q on the ith row and the jth column.

With the iteration index denoted by $\kappa = 0, 1, \dots,$ the average consensus algorithm is given by

$$x(\kappa + 1) = Qx(\kappa), \quad (2)$$

where $x(0)$ is the initial aggregate state at $\kappa = 0$. Rewrite iteration (2) in the following distributed fashion,

$$x_i(\kappa + 1) = q_{ii}x_i(\kappa) + \sum_{j\in N_i} q_{ij}x_j(\kappa),\ \forall i = 1, \dots, n. \quad (3)$$

We say that algorithm (2) solves the *average consensus problem asymptotically*, i.e., for any initial states $x_i(0)$'s, it follows

$$\lim_{\kappa\to\infty} x_i(\kappa) = \left(\sum_{j=1}^{n} x_j(0)\right) /n,\ \forall i = 1, \dots, n.$$

Note that matrix Q can be obtained locally, and algorithm (3) can be implemented in a distributed fashion. Therefore the average consensus algorithm (2) is fully distributed. See more details in [12]–[14].

C. Problem Formulation

Now we present the problem formulation of the extended EDP, which aims at minimizing the total cost for power generation of multiple generators to provide the desired amount of power within the generators' capabilities.

Suppose that there are in total n generators in the power grid, labelled from 1 to n. Let us denote the total load demand and the output of the ith generator by P_L and P_i, respectively. With the line capacity constraints neglected, we formulate the extended EDP as follows:

- Objective

$$\min \sum_{i=1}^{n} F_i(P_i) = \sum_{i=1}^{n} \left(\frac{1}{2}\alpha_i P_i^2 + \beta_i P_i + \gamma_i\right), \quad (4)$$

  where $F_i(P_i)$ is the cost function associated with the ith generator, and $\alpha_i > 0$, β_i, and γ_i are the cost parameters [1].
- Capacity constraints of generators

$$\underline{P}_i \leqslant P_i \leqslant \bar{P}_i,\ \forall i \in V, \quad (5)$$

  where $\underline{P}_i$ and $\bar{P}_i$ are the lower bound and upper bound of the output of the ith generator, respectively.
- Power balance constraint

$$\sum_{i=1}^{n} P_i - P_{loss} - P_L = 0, \quad (6)$$

  where P_{loss} is the total transmission losses all over the power grid.

In this paper, we assume that the total transmission losses are a function of the generator outputs P_i's and we use the **B** *matrix loss formula* (**B** coefficients) to represent P_{loss}, given by

$$P_{loss} = \sum_{i=1}^{n}\sum_{j=1}^{n} P_i B_{ij} P_j + \sum_{i=1}^{n} B_{0i}P_i + B_{00}, \quad (7)$$

where $B_{ij} = B_{ji}$, B_{0i}, and B_{00} are computed according to the line parameters and the average daily operating status of the power systems [11, Chap. 13.3]. Though total transmission losses can also be computed using the power flow equations, we use the **B** *matrix loss formula*, for it can give a sufficiently accurate estimation of the total transmission losses in the off-line mode with a small amount of computation [1].

In this paper, we impose a communication network on the power grid with each node in the communication network associated with a generation bus. The nodes (generators) only communicates with their neighbors in a connected undirected graph $G = (V, E)$, where $V = \{1, \dots, n\}$. Furthermore, to highlight the meaning of the decentralization of our algorithm, we further assume that G is a sparse graph in the sense:

$$\max_{i=1,\dots,n} d_i \ll n.$$

D. Centralized Solution to Extended EDP

We then introduce a centralized solution to the extended EDP (4)-(6). This centralized algorithm is based on the Lagrange dual method, which is usually referred to as the Lambda iteration method [1].

The Lagrange function of the extended EDP (4)-(6) is given by

$$L = \sum_{i=1}^{n} F_i(P_i) - \lambda \left(\sum_{i=1}^{n} P_i - P_{loss} - P_L \right),$$

where λ is the Lagrange multiplier.

The optimal solution and the optimal Lagrange multiplier, denoted by P_i^* and λ^* respectively, satisfy

$$\frac{\partial L}{\partial P_i^*} = \frac{\mathrm{d}F_i(P_i^*)}{\mathrm{d}P_i^*} - \lambda^* \left(1 - \frac{\partial P_{loss}}{\partial P_i^*}\right) = 0, \forall i. \quad (8)$$

Combining the cost functions (4) and the inequality constraints (5), we have for all $i \in V$,

$$\begin{cases} (\alpha_i P_i^* + \beta_i) pf_i > \lambda^*, & \text{for } P_i^* = \underline{P}_i, \\ (\alpha_i P_i^* + \beta_i) pf_i = \lambda^*, & \text{for } \underline{P}_i < P_i^* < \bar{P}_i, \\ (\alpha_i P_i^* + \beta_i) pf_i < \lambda^*, & \text{for } P_i^* = \bar{P}_i, \end{cases} \quad (9)$$

where pf_i is the *penalty factor*:

$$pf_i = 1 / \left(1 - \frac{\partial P_{loss}}{\partial P_i}\right).$$

From the loss formula (7), it follows that

$$pf_i = 1 / \left(1 - 2\sum_{j=1}^{n} B_{ij} P_j - B_{0i}\right). \quad (10)$$

We can obtain the optimal solution P_i^*'s and the optimal Lagrange multiplier λ^* by combining (6) and (9). But P_{loss} has a quadratic term, making it more complex to compute P_i^*'s and λ^*. We now introduce the following iterative algorithm in [1], with the iteration step denoted by $k = 0, 1, \dots$

Step 1: At $k = 0$, the generators pick initial values $P_i[0]$'s such that

$$\sum_{i=1}^{n} P_i[0] - P_L = 0.$$

Step 2: Compute the penalty factors $pf_i[k]$'s and the total transmission losses $P_{loss}[k]$ according to (10) and (7), respectively.

Step 3: Solve the following equations to get $P_i[k+1]$'s and $\lambda[k+1]$.

$$\sum_{i=1}^{n} P_i[k+1] - P_{loss}[k] - P_L = 0, \quad (11)$$

$$\begin{cases} \frac{\mathrm{d}F_i(P_i[k+1])}{\mathrm{d}P_i[k+1]} pf_i[k] > \lambda[k+1], & P_i[k+1] = \underline{P}_i, \\ \frac{\mathrm{d}F_i(P_i[k+1])}{\mathrm{d}P_i[k+1]} pf_i[k] = \lambda[k+1], & P_i[k+1] \in (\underline{P}_i, \bar{P}_i), \\ \frac{\mathrm{d}F_i(P_i[k+1])}{\mathrm{d}P_i[k+1]} pf_i[k] < \lambda[k+1], & P_i[k+1] = \bar{P}_i, \end{cases} \quad (12)$$

Step 4: Go back to Step 2 and loop until convergence.

Remark 1: The algorithm above has been widely used in power industries [15]. The core idea is to use the penalty factors and the transmission losses at the previous iteration as an estimation of the current penalty factors and the transmission losses. In this way we get over the difficulty of directly solving (6) and (9).

III. Distributed Algorithm for Extended EDP

In this section we present our distributed algorithm, which is based on the centralized algorithm introduced in Section II-D, the average consensus algorithm (2), and adopts the idea of bisection.

To make our algorithm work, a leader node which knows the total demand P_L and the coefficient B_{00} is needed. The leader node only communicates with m nodes, where m and n are known by all the nodes and $m < n$. To make our distributed algorithm meaningful, we further assume that $m \ll n$. Denote the set of the nodes communicating with the leader node by $V_1 = \{1, \dots, m\}$, and define $V_2 = V - V_1 = \{m+1, \dots, n\}$, which is the subset made up of the nodes without communicating with the leader node. The integral procedures are as follows.

Step 1: broadcast the total demand information P_L. We use the average consensus algorithm with heterogeneous initialization of the nodes such that after convergence each node will get the total demand information.

The leader node broadcasts $(P_L + B_{00})/m$ to the nodes in V_1. Each node in V establishes a variable $x_i(\kappa)$. For initialization at $\kappa = 0$, assign

$$x_i(0) = \begin{cases} (P_L + B_{00})/m, & \text{for } i \in V_1, \\ 0, & \text{for } i \in V_2. \end{cases}$$

Run the following average consensus algorithm till convergence,

$$x_i(\kappa+1) = q_{ii} x_i(\kappa) + \sum_{j \in N_i} q_{ij} x_j(\kappa), \ \forall i \in V. \quad (13)$$

When iteration (13) converges, each node $i \in V$ will get a common value x^* given by

$$x^* = \lim_{\kappa\to\infty} x_i(\kappa) = \left(\sum_{j=1}^{n} x_j(0)\right)/n = \frac{P_L + B_{00}}{n}.$$

Step 2: determine $P_i[0]$'s satisfying the inequality constraints (5) and

$$\sum_{i=1}^{n} P_i[0] - P_L - B_{00} = 0.$$

The average consensus algorithm is also used here in order that each node gets the estimation of the aggregate generation capability of all the generators.

For this purpose, each node $i \in V$ establishes two auxiliary variables $\underline{x}_i(\kappa)$ and $\bar{x}_i(\kappa)$, initialized by

$$\underline{x}_i(0) = \underline{P}_i,\ \bar{x}_i(0) = \bar{P}_i.$$

Then run the following average consensus iterations till convergence,

$$\underline{x}_i(\kappa+1) = q_{ii}\underline{x}_i(\kappa) + \sum_{j\in N_i} q_{ij}\underline{x}_j(\kappa),\ \forall i \in V, \tag{14}$$

$$\bar{x}_i(\kappa+1) = q_{ii}\bar{x}_i(\kappa) + \sum_{j\in N_i} q_{ij}\bar{x}_j(\kappa),\ \forall i \in V. \tag{15}$$

When iterations (14) and (15) converge, each node $i \in V$ will get the common values $\underline{x}^*$ and $\bar{x}^*$ similarly given by

$$\underline{x}^* = \left(\sum_{i=1}^{n} \underline{P}_i\right)/n,\ \bar{x}^* = \left(\sum_{i=1}^{n} \bar{P}_i\right)/n. \tag{16}$$

After obtaining $\underline{x}^*$ and $\bar{x}^*$, the nodes can get the $P_i[0]$'s following

$$P_i[0] = \underline{P}_i + \frac{x^* - \underline{x}^*}{\bar{x}^* - \underline{x}^*}(\bar{P}_i - \underline{P}_i),\ \forall i \in V. \tag{17}$$

Step 3: The computation of the penalty factors $pf_i[k]$'s and the total transmission losses $P_{loss}[k]$.

We assume that each node i knows the **B** coefficients associated with itself, i.e., $B_{ij},\ \forall j \in V$. The key to calculating $pf_i[k]$ is to calculate $\sum_{j=1}^{n} B_{ij}P_j[k]$ in a distributed fashion. For this purpose, each node $i \in V$ establishes an auxiliary variable $y_i^j(\kappa)$, where the superscript j represents y_i^j is meant for the calculation of $pf_j[k]$.

We initialize y_i^j's according to

$$y_i^j(0) = B_{ij}P_i[k]. \tag{18}$$

Then run the following iteration till convergence,

$$y_i^j(\kappa+1) = q_{ii}y_i^j(\kappa) + \sum_{l\in N_i} q_{il}y_l^j(\kappa),\ \forall i \in V. \tag{19}$$

Denote the convergence value of (19) by y^{j*}, it follows that

$$y^{j*} = \left(\sum_{i=1}^{n} B_{ij}P_i[k]\right)/n.$$

Therefore the penalty factor pf_j is given by

$$pf_j[k] = 1/\left(1 - 2ny^{j*} - B_{0j}\right). \tag{20}$$

Loop until all the nodes $j \in V$ obtains their $pf_j[k]$'s.

We then compute the total transmission losses $P_{loss}[k]$. Since the constant term B_{00} has already been included in x^*, we only need to compute

$$\begin{aligned} P'_{loss}[k] &= \sum_{i=1}^{n}\sum_{j=1}^{n} P_i[k]B_{ij}P_j[k] + \sum_{i=1}^{n} B_{0i}P_i[k] \\ &= \sum_{i=1}^{n}\left(ny^{i*}[k]P_i[k] + B_{0i}P_i[k]\right). \end{aligned} \tag{21}$$

For this purpose, each node establishes an auxiliary variable $y_i(\kappa)$, initialized by

$$y_i(0) = ny^{i*}[k]P_i[k] + B_{0i}P_i[k]. \tag{22}$$

And then run the following average consensus algorithm till convergence,

$$y_i(\kappa+1) = q_{ii}y_i(\kappa) + \sum_{j\in N_i} q_{ij}y_j(\kappa),\ \forall i \in V. \tag{23}$$

Denote the convergence value of (23) by y^*, it follows that

$$y^* = \sum_{i=1}^{n}\left(ny^{i*}[k]P_i[k] + B_{0i}P_i[k]\right)/n = P'_{loss}[k]/n.$$

Step 4: The calculation of $P_i[k+1]$'s and $\lambda[k+1]$.

Rewrite (12) in the following form:

$$P_i[k+1] = \begin{cases} \underline{P}_i, & \frac{\lambda[k+1]-\beta_i PF_i[k]}{\alpha_i PF_i[k]} < \underline{P}_i, \\ \frac{\lambda[k+1]-\beta_i PF_i[k]}{\alpha_i PF_i[k]}, & \frac{\lambda[k+1]-\beta_i PF_i[k]}{\alpha_i PF_i[k]} \in (\underline{P}_i, \bar{P}_i), \\ \bar{P}_i, & \frac{\lambda[k+1]-\beta_i PF_i[k]}{\alpha_i PF_i[k]} > \bar{P}_i. \end{cases} \tag{24}$$

Note that for all $i \in V$, $\alpha_i > 0$ and $pf_i[k] > 0$, therefore the function $\frac{\lambda[k+1]-\beta_i PF_i[k]}{\alpha_i PF_i[k]}$ is monotonically increasing with respect to $\lambda[k+1]$. So $P_i[k+1]$ is also monotonically increasing with respect to $\lambda[k+1]$, which enables us to use the bisection method. The detailed procedures are as follows.

Each node establishes two commonly shared variables λ^+ and λ^- such that the optimal Lagrange multiplier must lie in the interval $[\lambda^-, \lambda^+]$. The initial λ^+ and λ^- can be selected to be extremely large and small, respectively, for the convergence of bisection is very fast.

Let $t = 0, 1, \ldots$ denote the bisection steps. At step t, each node computes

$$\lambda(t+1) = (\lambda^-(t) + \lambda^+(t))/2.$$

Each node then obtains $P_i(\lambda(t+1))$ according to (24) with $\lambda[k+1]$ replaced by $\lambda(t+1)$, and then establishes a variable $z_i(\kappa)$, which is initialized by

$$z_i(0) = P_i(\lambda(t+1)). \tag{25}$$

Run the following iteration till convergence,

$$z_i(\kappa+1) = q_{ii}z_i(\kappa) + \sum_{j\in N_i} q_{ij}z_j(\kappa),\ \forall i \in V. \tag{26}$$

After convergence, denote the convergence value by z^*. Then each node updates $\lambda^-(t+1)$ and $\lambda^+(t+1)$ according to

- For $z^* < y^*[k] + x^*$,

$$\lambda^+(t+1) = \lambda^+(t),\ \lambda^-(t+1) = \lambda(t+1).$$

- For $z^* = y^*[k] + x^*$,

$$\lambda[k+1] = \lambda(t+1),\ \text{and the bisection stops.}$$

- For $z^* > y^*[k] + x^*$,

$$\lambda^+(t+1) = \lambda(t+1),\ \lambda^-(t+1) = \lambda^-(t).$$

Recompute $P_i(\lambda(t+1))$'s and circulate the bisection until convergence. Then each node obtains $P_i[k+1]$ and $\lambda[k+1]$.
Step 5: Go back to Step 2 and loop until convergence. Then each nodes obtains the optimal solution P_i^*.

The overall procedure of our distributed algorithm is summarized in Algorithm 1.

Algorithm 1 Distributed Algorithm for Extended EDP

Require: P_L: the total load demand;
Ensure: P_i^*: the optimal power assignment, $\forall i \in V$;

```
The leader node broadcasts the total demand information to the nodes in V_1;
All the nodes obtain x* using iteration (13);
Each node determines P_i[0]'s according to Step 2;
for k = 0, 1, 2, ... do
    Each node computes their penalty factors pf_i[k]'s and the transmission losses P'_loss[k] using (18)-(21);
    Each node initializes λ+(0) and λ-(0);
    for t = 0, 1, ... do
        Each node computes λ(t+1) = (λ+(t)+λ-(t))/2;
        Each node computes P_i(λ(t+1)) using (24)-(26);
        Each node updates λ+ and λ- accordingly;
    end for
end for
```

Remark 2: Our algorithm is distributed in the sense that the nodes conduct local computation and bidirectional communication with their neighbors, except for those nodes who also need to communicate with the aggregator.

IV. Numerical Simulation

In this simulation case, we apply our distributed algorithm to the extended EDP on the IEEE 30-bus system [17]. The generator parameters are adopted from [7], where we set $\gamma_i = 0$ MU for all i, for it does not affect the power dispatch. The following are the **B** coefficients for the IEEE 30-bus system [18].

B_{ij}	0.1382	-0.0299	0.0044	-0.0022	-0.0010	-0.0008
	-0.0299	0.0487	-0.0025	0.0004	0.0016	0.0041
	0.0044	-0.0025	0.0182	-0.0070	-0.0066	-0.0066
	-0.0022	0.0004	-0.0070	0.0137	0.0050	0.0033
	-0.0010	0.0016	-0.0066	0.0050	0.0109	0.0005
	-0.0008	0.0041	-0.0066	0.0033	0.0005	0.0244
B_0	-0.0107	0.0060	-0.0017	0.0009	0.0002	0.0030
B_{00}	0.00098573					

There are in total 6 generators in the IEEE 30-bus system, whose parameters are shown in Table I. We set $\underline{P}_i = 10$ MW, so they are not shown in the table. The communication network is a connected undirected graph as shown in Fig. 1, where the leader node is labelled 0. The solid lines represent the bidirectional communication between generators, while the dotted lines represent the unidirectional information paths from the leader node to the nodes in $V_1 = \{1, 2\}$. Note that the communication paths do not necessarily coincide with the power transmission lines.

TABLE I
Generator Parameters (MU = Money Unit)

Generator	α_i (MU/MW^2)	β_i (MU/MW)	$\bar{P}_i(MW)$
1	0.08	2.0	80
2	0.06	3.0	90
3	0.07	4.0	70
4	0.06	4.0	70
5	0.08	2.5	80
6	0.08	2.5	80

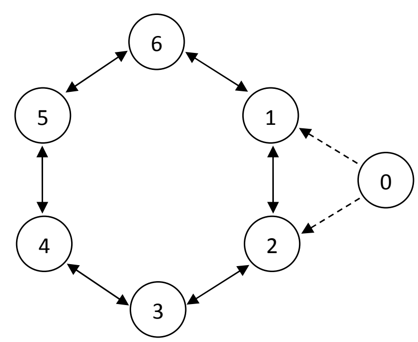

Fig. 1. Communication network of the 6 generators and the leader node.

In this case the total power demand is $P_L = 300$ MW. We set $\lambda^+(0) = 10$ MU/MW and $\lambda^-(0) = 0$ MU/MW, which are sufficient to ensure $\lambda^* \in [\lambda^-(0), \lambda^+(0)]$. In the calculation of $P_i[k]$'s and $\lambda[k]$, we artificially set the bisection number to 15, such that for each k, $|\lambda(15) - \lambda[k]| \leqslant \frac{1}{2}|\lambda^+(14) - \lambda^-(14)| = \frac{1}{2^{15}}|\lambda^+(0) - \lambda^-(0)| \approx 0.0003$.

We first give the results of this case neglecting the transmission losses. The optimal Lagrange multiplier $\lambda^* = 6.5944$ MU/MW, and the optimal power assignments are $P_1^* = 57.43$ MW, $P_2^* = 59.91$ MW, $P_3^* = 37.06$ MW, $P_4^* = 43.24$ MW, $P_5^* = 51.18$ MW, $P_6^* = 51.18$ MW. Note that generator 5 and generator 6 have the same optimal assignments because they are identical.

TABLE II
ITERATIVE RESULTS OF $\lambda[k]$ AND THE TOTAL COST

Iteration	$\lambda[k]$ (MU/MW)	$\sum_{i=1}^{n} F_i(P_i[k])$ (MU)
1	7.7335	1429.3
2	6.8521	1729.4
3	6.8668	1459.2
4	6.8582	1467.2
5	6.8600	1460.3
6	6.8600	1460.8
7	6.8600	1460.9

We then show the results of the EDP including transmission losses using the proposed algorithm. The updates of $\lambda[k]$ and $P_i[k]$'s are shown in Table. II and Fig. 2, respectively. We set the iteration step to 10, while the convergence is already reached at $k = 7$. The optimal Lagrange multiplier $\lambda^* = 6.8600$ MU/MW, and the optimal power assignments are $P_1^* = 52.36$ MW, $P_2^* = 60.05$ MW, $P_3^* = 41.38$ MW, $P_4^* = 45.99$ MW, $P_5^* = 53.44$ MW, $P_6^* = 51.88$ MW. The aggregate generation output is $\sum_{i=1}^{n} P_i^* = 305.11 > 300$ MW, which is caused by the transmission losses.

We can see that the presence of transmission losses slightly increase the optimal incremental cost (Lagrange multiplier) by $6.8600 - 6.5944 = 0.2656$ MU/MW. Note that in this case the optimal power assignments of generator 5 and generator 6 are different. The reason is that the **B** coefficients associated with generator 5 and generator 6 are heterogenous, thus leading to the difference in penalty factors.

Moreover, from an intuitive thought, since the optimal incremental cost increases when transmission losses are considered, the optimal power assignment for each generator in the presence of transmission losses should be also larger than that when transmission losses are neglected. But in fact sometimes it is not the case. The power assignment for generator 1 neglecting transmission losses is 57.43 MW, which is larger than 52.63 MW, the power assignment including transmission losses. This is also due to the transmission losses and the consequent penalty factors. In this case, the penalty factor of generator 1 is $pf_1 = 1.1084 > 1$.

V. CONCLUDING REMARKS

In this paper, we propose a distributed algorithm based on the average consensus algorithm and the bisection method to solve the EDP including transmission losses. Our algorithm is distributed in the sense that the nodes conduct local computations. Through numerical experiments we show the effectiveness of the proposed algorithm. Future work would include the extension of our algorithm to the EDP with other practical constraints, e.g., line capacity constraints, the spinning reserve, and prohibited operating zones of generators.

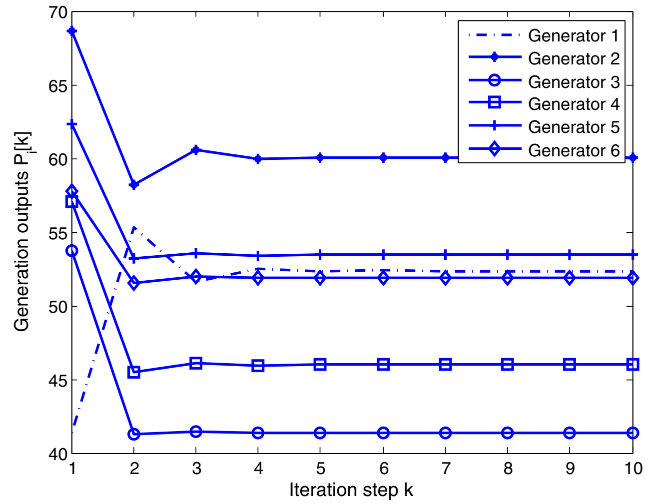

Fig. 2. Iteration results of the generators' outputs $P_i[k]$.

REFERENCES

[1] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation, and Control*. John Wiley & Sons, 1996.
[2] K. Moslehi and R. Kumar, "A reliability perspective of the smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 1, pp. 57–64, 2010.
[3] L. Bakule, "Decentralized control: an overview," *Annual Reviews in Control*, vol. 32, no. 1, pp. 87–98, 2008.
[4] Z. Zhang, X. Ying, and M.-Y. Chow, "Decentralizing the economic dispatch problem using a two-level incremental cost consensus algorithm in a smart grid environment," in *Proc. IEEE North American Power Symposium (NAPS)*, Boston, MA, 2011, pp. 1–7.
[5] Z. Zhang and M.-Y. Chow, "Incremental cost consensus algorithm in a smart grid environment," in *Proc. IEEE Power and Energy Society General Meeting*, San Diego, CA, 2011, pp. 1–6.
[6] A. D. Dominguez-Garcia, S. T. Cady, and C. N. Hadjicostis, "Decentralized optimal dispatch of distributed energy resources," in *Proc. IEEE Conference on Decision and Control*, Maui, Hawaii, 2012, pp. 3688–3693.
[7] S. Kar and G. Hug, "Distributed robust economic dispatch in power systems: A consensus + innovations approach," in *Proc. IEEE Power and Energy Society General Meeting*, San Diego, CA, 2012, pp. 1–8.
[8] S. Yang, S. 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, 2013.
[9] H. Xing, Y. Mou, M. Fu, and Z. Lin, "Consensus based bisection approach for economic power dispatch," in *Proc. IEEE Conference on Decision and Control*, 2014, accepted.
[10] ——, "Distributed bisection method for economic power dispatch in smart grid," *IEEE Trans. Power Syst.*, under review, 2014.
[11] J. J. Grainger and W. D. Stevenson, *Power System Analysis*. McGraw-Hill New York, 1994, vol. 621.
[12] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
[13] Z. Lin, *Distributed Control and Analysis of Coupled Cell Systems*. VDM Publishing, 2008.
[14] A. Bemporad, M. Heemels, and M. Johansson, *Networked Control Systems*. Springer, 2010, vol. 406.
[15] H. Happ, "Optimal power dispatch?a comprehensive survey," *IEEE Trans. Power App. Syst.*, vol. 96, no. 3, pp. 841–854, 1977.
[16] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*. Prentice-hall Englewood Cliffs, NJ, 1987, vol. 2.
[17] "Power system test case archive." [Online]. Available: http://www.ee.washington.edu/research/pstca/
[18] D.-w. Gong, Y. Zhang, and C.-l. Qi, "Environmental/economic power dispatch using a hybrid multi-objective optimization algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 32, no. 6, pp. 607–614, 2010.