Distributed Demand and Response Algorithm for Optimizing Social-Welfare in Smart Grid

Qifen Dong
College of Information Engineering
Zhejiang University of Technology
Hangzhou, China
qdong@cs.gsu.edu

Li Yu
College of Information Engineering
Zhejiang University of Technology
Hangzhou, China
lyu@zjut.edu.cn

Wen-Zhan Song
Department of Computer Science
Georgia State University
Atlanta, USA
wsong@gsu.edu

Lang Tong
School of Electrical and Computer Engineering
Cornell University
Ithaca, NY 14853
ltong@ece.cornell.edu

Shaojie Tang
Department of Computer Science
Illinois Institute of Technology
Chicago, IL 60616
stang7@hawk.iit.edu

***Abstract*—This paper presents a distributed Demand and Response algorithm for smart grid with the objective of optimizing social-welfare. Assuming the power demand range is known or predictable ahead of time, our proposed distributed algorithm will calculate demand and response of all participating energy demanders and suppliers, as well as energy flow routes, in a fully distributed fashion, such that the social-welfare is optimized. During the computation, each node (e.g., demander or supplier) only needs to exchange limited rounds of messages with its neighboring nodes. It provides a potential scheme for energy trade among participants in the smart girds. Our theoretical analysis proves that the algorithm converges even if there is some random noise induced in the process of our distributed Lagrange-Newton based solution. The simulation also shows that the result is close to that of centralized solution.**

***Keywords*-Demand Response, Lagrange-Newton method, distributed, Social-Welfare**

I. Introduction

Smart grid is a type of electrical grid which attempts to predict and intelligently respond to the behavior and actions of all electric power users connected to it - suppliers, consumers and those that do both, in order to efficiently deliver reliable, economical, and sustainable electricity services. There are many open research problems that need to be solved before it comes into service. Demand Response (DR) [1], which refers to the dynamic demand mechanisms to manage customer consumption of electricity in response to supply conditions, is one of the most important functions of smart grid. The traditional DR mechanism, such as Critical Peak Pricing, Time-of-Use Pricing, and Real-Time Pricing, are relatively mature in traditional electricity grids. However, a traditional power grid is a one-way energy broadcasting network. Most DR schemes are executed by the power plant in a centralized manner. In the future, more renewable energy sources will be integrated into the grid, and this could fundamentally change the operation paradigm. The energy suppliers and demanders are distributively interconnected with each other. It is desirable that the DR solution is executed in a fully distributed manner. The advancement of smart grid technologies including digital communication devices and advanced metering infrastructures facilitate information exchange between users and electric utilities, and provide necessary infrastructure to support distributed DR.

In this paper, we propose an innovative distributed Demand and Response algorithm for optimizing social-welfare in smart grid. Here, social-welfare is the difference of the sum of users' utilities and the total cost of energy generators and transmission networks. We assume the algorithm can be run periodically and the range of energy demand and supply in the next time period is known or predictable. Before the next time slot starts, our algorithm will compute the consumption/generation amount of each consumer and energy provider that maximizes social-welfare. The computation will be done fully distributively and each node (e.g., demander or supplier) only needs to exchange limited rounds of messages with its neighboring nodes. Our distributed DR algorithm is based on the distributed Lagrange-Newton method initially developed in [2], [3] for network utility optimization. The main contributions of this paper are summarized as follows:

- We build an optimization model for scheduling users' energy demand amounts, and generation capacity of various energy generators. All of them fall into their own pre-defined regions. A distributed Lagrange-

The work is supported by Zhejiang-ORFG-20110804, NSF-CPS-1135814.This work is also supported by the scholarship under the State Scholarship Fund, China

Newton algorithm is introduced to solve it.

- The proposed DR program can handle energy transactions among demanders and suppliers. The values of Locational Marginal Prices (LMPs) which achieve a market equilibrium point are also determined. LMP is the cost to serve the next MW of load at a specific location, using the lowest production cost of all available generation, while observing all transmission limits. The LMPs emerge as the Lagrange multipliers. i.e. dual variables, corresponding to power flow balance constraints [4].
- We propose an innovative algorithm to compute dual variables and step-size in a distributed manner. Because the constraints in our system are more complex than those in [2], [3], the distributed computation of dual variables and step size could not be applied directly.
- The convergence is analyzed when a certain error is introduced in computing dual variables and step-size.

The remainder of the paper is organized as follows. Related works are summarized in Section II. Section III presents the system model, and the demand scheduling is formulated into a convex optimization. In section IV, the optimization problem is solved through distributed Lagrange-Newton algorithm. The convergence analysis of the solution is given in section V. Section VI shows simulation results. The conclusion is given in section VII.

II. Related Works

The DR algorithms for smart grid have drawn much research attention in recent years. According to the decision variable, these DR algorithms can be roughly categorized into two groups. One aims at deciding when to start requested electrical appliances, the other refers to how much energy to allocate to users during each time slot.

The DR algorithms in the first category aim at controlling when the electrical appliances shall run, with consideration of several factors, e.g. available energy, and pre-defined deadlines. For example, a refrigerator could delay or advance the start time of its cooling cycle within certain time periods. Authors in [5] design a mechanism for a household to compete with neighborhoods for the available power. Then Dynamic Programming is introduced to optimize the timing of appliance operation. An electricity bill minimization problem of cooperative users is studied in [6]. The basic idea is to schedule user requests for appliance operation at different times during a fixed interval based on dynamic energy prices and available power capacity during that interval. A Consumer Automated Energy Management System (CAES) is proposed in [7]. A user selects appliances indicating his desire to run them, then CAES determines the optimal time to run the appliances and how much energy will be allocated, with the aim at minimizing the sum of infinite horizon average financial cost of consuming energy and the average dis-utility to the user for delaying operation of the selected appliances. *Stephane* and *George* aim at reducing operating cost of electric utility during the intended time periods by scheduling the start time of users' demands [8].

The goal of the second category is to estimate the amount of energy consumed by consumers in a given time slot, subject to some constraints, e.g. minimum consumption requirements of the energy consumers, and maximum generation capacity during this time slot. For example, in summer, people feel much cooler when the air-conditioner is set at $22^{\circ}C$. However, people are still comfortable when the temperature is at $28^{\circ}C$. Thus, the temperature of the air conditioner should be adjusted to match available generation in this time slot. In [9], a smart power infrastructure in which several energy consumers share a common energy resource is considered. It focuses on finding the energy consumption of each energy consumer and the generation level of the energy provider within their minimum and maximum intervals to optimize social-welfare. Their social-welfare function is defined as the sum of all energy consumers' utilities functions minus the cost imposed on the energy provider. Further, a sub-gradient method is used to solve this problem in a distributed fashion. During each time period, each energy consumer estimates its power consumption through iterative computation, and the energy provider determines generation amount. Authors in [10] investigate problem similar to that in [9]. The difference is that they consider distributed energy suppliers with different retail prices, instead of a single provider. In addition, the energy transmission constraint is taken into account. Then an alternative solution based on a sub-gradient algorithm is proposed to solve it. In the two papers, the LMPs are also determined during the computation. In addition, the authors in [11] focus on perturbation analysis of market equilibrium in the presence of fluctuations in renewable energy resources and demand, with a model similar to that used in [10]. Additional results can be found in [12]–[15].

This paper focuses on the latter category, especially a problem similar to [9]–[11]. However, the differences in this paper mainly include the following two aspects:

1) Besides consumer utilities and generation cost of various energy suppliers in social-welfare, we also consider the energy demand/generation decisions that reduce transmission loss of the whole grid.
2) We utilize the distributed Lagrange-Newton method recently developed in [2], [3] for network utility optimization to solve the DR problem in a fully distributed manner. However, it requires global information to determine the energy price in [9]–[11].

III. System model

Consider a smart grid system containing n nodes (buses) and L transmission lines, as shown in Fig. 1. One or more energy generators are installed at some of the nodes and there are a total of m generators. For simplicity, the demand

is assumed homogeneous, hence, all demands connected to one node are treated as a single consumer [10]. It also assumes that there is one consumer located at each node. Each consumer i has a utility function $u_i(x)$ representing the monetary benefit that the consumer derives from consuming x units of electricity. Similarly, to each energy generator i, we associate a cost function $c_i(x)$ capturing the monetary cost of generating x units of energy. The two functions fulfill the following assumptions:

Assumption 1: For each consumer i, the utility function $u_i(x)$ is non-decreasing and strictly concave. That is, the consumer is always interested in meeting its demand if possible. However the satisfaction level for consumers can gradually get saturated when reaching their maximum consumption level. Mathematically, it implies $\frac{\partial u_i(x)}{\partial x} \geq 0,\ \frac{\partial^2 u_i(x)}{\partial x \partial x} < 0$

Assumption 2: For each generator i, the cost function $c_i(x)$ is non-decreasing and strictly convex. In other words, the production cost increases as the production amount increases, and the unit cost increases quickly when the amount of generation exceeds a threshold. It can be stated mathematically as $\frac{\partial c_i(x)}{\partial x} \geq 0,\ \frac{\partial^2 c_i(x)}{\partial x \partial x} > 0$.

Regarding transmission lines, the line resistance is linearly proportional to the length of the transmission line. The energy losses in transmission lines cannot be ignored. We consider the following monetary loss function that is caused by transmission losses:

Assumption 3: When x units of current flow through a transmission line l whose line resistance is r_l, the monetary cost of loss is denoted by $w_l(x) = cx^2 r_l$, where c is a constant. It is a strictly convex function of the current.

In addition, we suppose there is an Energy Consumption Controller (ECC) unit and Energy Generation Controller (EGC) unit embedded in the consumer's and energy generator's smart meter, respectively. The role of ECC is to control the consumer's energy consumption, while the EGC controls energy production.

Under this smart grid structure, it should optimally match energy supplies and demands. On one hand, from a social fairness point of view, it is desirable to utilize the available power provided by the energy generators in such a way that the sum of all consumers' utilities is maximized and the cost imposed to all the energy generators is minimized. On the other hand, as pointed out above, energy losses in transmission lines should be taken into account. For example, it is more efficient for a consumer to use energy supplied by a nearby generator than by those far-away generators. A social-welfare function is induced to address these factors. The social-welfare is defined as the sum of all consumers utilities minus the total cost experienced by all the generators and wastage cost caused by transmission losses. To be more specific, let d_i, g_i and I_i denote the amount of energy consumed by consumer i, energy provided by generator i, and the current flow through line i in a time slot, respectively. All measurements are in ampere. Then the social-welfare is formulated as follows:

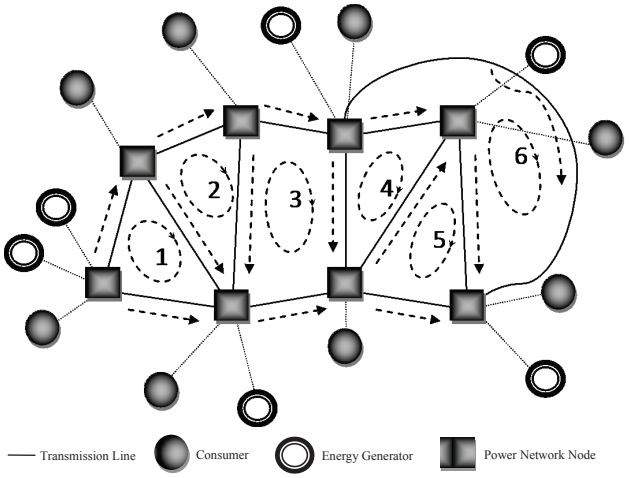

Figure 1. Smart Grid System Modeling

$$S = \sum_{i=1}^{n} u_i(d_i) - \sum_{i=1}^{m} c_i(g_i) - \sum_{i=1}^{L} w_i(I_i)$$

Generally, for each time slot, each consumer has a minimum and maximum energy demand requirement, the available energy provided by each generator is limited, and maximum current flow of each transmission line is also limited. Further, the grid should also be constrained by Kirchhoff's current and voltage laws (KCL and KVL). There are n independent KCL equations and $p = L - n$ independent KVL equations in the smart grid model shown in Fig. 1. The p loops can be described by several methods. One of the most simple and straightforward ways is observing the meshes. As shown in Fig. 1, the six meshes correspond to $p = 6$ independent loops. To deal with Kirchhoff's law, the reference direction of the line current should be specified, e.g. from left to right and top to bottom, as shown in Fig.1. Likewise, it also needs to specify the loop direction, e.g. clockwise or counterclockwise.

Overall, our goal is to find the values of d_i, g_i and I_i that maximize social-welfare, subject to KCL and KVL constraints ((1b) and (1c)) as well as demand requirement constraints (1d), generation capacity constraints (1e), and transmission line constraints (1f), i.e. solving the following constrained optimization problem:

Problem 1:

$$\text{maximize} \quad S \tag{1a}$$

subject to constraints

$$\sum_{j \in s(i)} g_j + \sum_{j \in L\_in(i)} I_j - \sum_{j \in L\_out(i)} I_j - d_i = 0,\ i = 1, 2, \cdots n \tag{1b}$$

$$\sum_{l\in T(i)_+} r_l I_l - \sum_{l\in T(i)_-} r_l I_l = 0,\ i = 1, 2, \cdots, p \tag{1c}$$

$$d_i^{\min} \le d_i \le d_i^{\max},\ i = 1, 2, \cdots, n \tag{1d}$$

$$0 \le g_i \le g_i^{\max},\ i = 1, 2, \cdots, m \tag{1e}$$

$$-I_i^{\max} \le I_i \le I_i^{\max},\ i = 1, 2, \cdots, L \tag{1f}$$

where $s(i)$ is the set of generators located at node i, $L\_in(i)$ and $L\_out(i)$ are the sets of transmission lines whose currents flow in/out of node i respectively, while $T(i)_+$ and $T(i)_-$ are the sets of transmission lines which belong to loop i and has the same/opposite reference direction as loop i, respectively. In addition, $d_i^{\max}$, $d_i^{\min}$, $g_i^{\max}$ and $I_i^{\max}$ are limits of demand requirement, generation capacity and transmission line capacity. The providers will generate sufficient energy to cover minimum energy requirements of all consumers, i.e. $\sum_{i=1}^{m} g_i^{max} \ge \sum_{i=1}^{n} d_i^{min}$.

In fact, it is a convex optimization problem and can be solved using convex programming techniques in a centralized manner. However, the increasing distributed renewable energy resources is integrated into smart grid. It requires a decentralized DR algorithm. In addition, social and legal barriers of centralized solutions hinder their application in smart grid. These motivate us to solve the problem using a distributed Lagrange-Newton method.

IV. Distributed algorithm for optimization of smart grid

To facilitate utilization of the Lagrange-Newton method, the social-welfare maximization problem is rewritten as the following formula with only equality constraints by using logarithmic barrier functions. Let p be a positive constant coefficient for the logarithmic barrier functions, the solution of *Problem 2* approximately equivalent to that of *Problem 1*, as p approaches zero.

Problem 2:

$$\begin{aligned}\text{minimize} \sum_{i=1}^{m} c_i(g_i) + \sum_{i=1}^{L} w_i(I_i) - \sum_{i=1}^{n} u_i(d_i) \\ - p\sum_{i=1}^{L}\{\log(I_i + I_i^{\max}) + \log(I_i^{\max} - I_i)\} \\ - p\sum_{i=1}^{n}\left\{\log(d_i - d_i^{\min}) + \log(d_i^{\max} - d_i)\right\} \\ - p\sum_{i=1}^{m}\{\log(g_i) + \log(g_i^{\max} - g_i)\}\end{aligned} \tag{2a}$$

subject to constraints

$$\begin{bmatrix} K & G & E \\ 0 & R & 0 \end{bmatrix} \begin{bmatrix} g \\ I \\ d \end{bmatrix} = 0 \tag{2b}$$

where $g = [g_1, \cdots, g_m]^T$ and $I = [I_1, \cdots, I_L]^T$, $d = [d_1, \cdots, d_n]^T$, K is a $n \times m$ matrix representing at which node the generator is located, i.e.,

$$K_{ij} = \begin{cases} 1, & \text{if generator } j \text{ located at node } i \\ 0, & \text{otherwise} \end{cases}$$

G is the $n \times r$ node-line incidence matrix of the smart grid network, i.e.,

$$G_{ij} = \begin{cases} 1, & \text{if the current of line } j \text{ flows into node } i \\ -1, & \text{if the current of line } j \text{ flows out of node } i \\ 0, & \text{otherwise} \end{cases}$$

E is a $n \times n$ diagonal matrix with $E_{ii} = -1$, and R is the $p \times L$ loop-impedance matrix satisfying:

$$R_{ij} = \begin{cases} r_j, & \text{if line } j \text{ is in loop } i \text{ with the same direction} \\ -r_j, & \text{if line } j \text{ is in loop } i \text{ with opposite direction} \\ 0, & \text{otherwise} \end{cases}$$

It should be noted that although demand requirement constraints, generation capacity constraints, and transmission line constraints are removed, the values of d_i, g_i and I_i must always remain in the feasible region, i.e. (1d), (1e) and (1f), in the whole process of Lagrange-Newton algorithm.

For notational simplicity, denote:

$$x = [g; I; d]; \quad A = \begin{bmatrix} K & G & E \\ 0 & R & 0 \end{bmatrix}$$

and the objective function in *Problem 2* is denoted by $f(x)$.

A. Equality Constrained Lagrange-Newton Method

Solve *problem 2* using equality constrained Lagrange-Newton method with infeasible start. In the smart grid system, primal variables are $x = [g; I; d]$, and dual variables are $v = [\lambda_1, \lambda_2, \cdots, \lambda_n, \mu_1, \mu_2, \cdots, \mu_n]^T$, where λ_i is the Lagrange multiplier corresponding to KCL constraint at node i, and μ_i is the Lagrange multiplier corresponding to KVL constraint of loop i. As mentioned previously, the solutions of λ_i, $i = 1, \cdots, n$ are the LMPs. Given an arbitrary initial primal vector x^0 within feasible region and a random initial dual vector v^0. At iteration k, x and v are updated by:

$$x^{k+1} = x^k + s^k \Delta x^k \tag{3a}$$

$$v^{k+1} = v^k + \Delta v^k \tag{3b}$$

where Δx^k and Δv^k are primal and dual Newton steps respectively, and s^k is a positive step size which should guarantee x^{k+1} still fall into the feasible region. Further, Δx^k and Δv^k are solutions of following system [16]:

$$\begin{bmatrix} \nabla^2 f(x^k) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta v^k \end{bmatrix} = - \begin{bmatrix} \nabla f(x^k) + A^T v^k \\ Ax^k \end{bmatrix}$$

Denote $H_k = \nabla^2 f(x^k)$ for notational simplicity, then Δx^k and Δv^k is solved in the following two steps:

$$(AH_k^{-1}A^T)\left(v^k + \Delta v^k\right) = Ax^k - AH_k^{-1}\nabla f(x^k) \tag{4a}$$

$$\Delta x^k = -H_k^{-1}\{\nabla f(x^k) + A^T(v^k + \Delta v^k)\} \quad (4b)$$

Since there are no couplings among d_i, I_i and g_i in the objective function of *Problem 2*, the Hessian matrix H_k is a diagonal matrix:

$$H_k = \begin{bmatrix} C & 0 & 0 \\ 0 & W & 0 \\ 0 & 0 & U \end{bmatrix}$$

where C, W and U are diagonal matrices with:

$$C_{ii} = \frac{\partial^2 c_i(g_i^k)}{\partial g_i^k \partial g_i^k} + \frac{p}{{g_i^k}^2} + \frac{p}{\left((g_i^{\max}) - g_i^k\right)^2} \quad i = 1, \cdots, m \quad (5a)$$

$$W_{ii} = \frac{\partial^2 w_i(I_i^k)}{\partial I_i^k \partial I_i^k} + \frac{p}{\left(I_i^{\max} - I_i^k\right)^2} + \frac{p}{\left(I_i^k + I_i^{\max}\right)^2} \quad i = 1, \cdots, L \quad (5b)$$

$$U_{ii} = -\frac{\partial^2 u_i(d_i^k)}{\partial d_i^k \partial d_i^k} + \frac{p}{\left(d_i^k - d_i^{\min}\right)^2} + \frac{p}{\left(d_i^{\max} - d_i^k\right)^2} \quad i = 1, \cdots, n \quad (5c)$$

Because the cost function $c_i(x)$ and the wastage function of transmission loss $w_i(x)$ are strictly convex, and the utility function $u_i(x)$ is strictly concave, it is easy to deduce that all diagonal elements are positive. Clearly, the inverse matrix H_k^{-1} is also diagonal and element-wise positive. Examine (4b), Δx^k can be computed in a distributed manner, under a given vector $v^{k+1} = v^k + \Delta v^k$. Further, if the positive step size s^k is also known, the primal vector x^{k+1} is updated locally according to (3a). Each node i executes the following computation locally to update:

(1) the values of g_j^k for the generators located at it

$$\Delta g_j^k = -C_{jj}^{-1}\left(\nabla f\left(g_j^k\right) + \lambda_i^{k+1}\right), \; g_j^{k+1} = g_j^k + s^k \Delta g_j^k \quad (6a)$$

(2) the values of I_l^k for its out-lines

$$\Delta I_l^k = -W_{ll}^{-1}\left(\nabla f\left(I_l^k\right) + q_l\right), \; I_l^{k+1} = I_l^k + s^k \Delta I_l^k \quad (6b)$$

$$q_l = \lambda_{i_l}^{k+1} - \lambda_i^{k+1} + \sum_{t \in m(l)} R_{tl}\mu_t^{k+1} \quad (6c)$$

(3) the value of d_i^k for the consumer connected to it

$$\Delta d_i^k = -U_{ii}^{-1}\left(\nabla f\left(d_i^k\right) - \lambda_i^{k+1}\right), \; d_i^{k+1} = d_i^k + s^k \Delta d_i^k \quad (6d)$$

where i_l denotes the node into which the current of line l flows, $m(l)$ denotes the loops to which line l belongs and a line belongs to at most two loops.

However, there are still two challenges with deriving the values of d_i, g_i and I_i in a distributed fashion:

1) *Distributed computation of dual variables*. It requires global information to compute the inverse matrix $(AH_k^{-1}A^T)^{-1}$, so the computation of the Lagrange multiplier vector $v^k + \Delta v^k$ cannot be implemented in a decentralized manner for a given primal vector x^k.
2) *Distributed computation of the step size*. In the Lagrange-Newton method, the step-size should be equal for all primal variables. In our smart grid case, it also should guarantee x^{k+1} fall into the feasible region. It is difficult to achieve such a step-size.

Authors in [2], [3] have addressed these challenges. However, the KVL constraints in power grid increase the difficulty of finding solution of dual variables. Neither the distributed computation of step size in [2] nor [3] can apply to our problem. Authors in [3] assume that the coefficient for barrier functions is larger than one, in order to prove convergence. This assumption will change the solution of original problem. The computation of step-size in [2] cannot satisfy the requirement that, at each iteration, the primal variables should fall into the feasible region. We propose alternative methods to achieve distributed computation of dual variables and step size in the following section.

B. Distributed Computation of Dual Variables

We first give a lemma about solving a system of linear equations using the matrix splitting technique.

Lemma 1: Let P be a $n \times n$ matrix, and b be a vector of length n. Suppose matrix P can be split into two matrices M and N, i.e. $P = M + N$, such that the spectral radius of $-M^{-1}N$, denoted by $\rho(-M^{-1}N)$, satisfies $\rho(-M^{-1}N) < 1$. Let $y(0)$ be an arbitrary initial vector of length n, then the sequence $\{y(t)\}$ generated by the following iterative procedure converges to the solution of linear equations $Py = b$:

$$y(t+1) = -M^{-1}Ny(t) + M^{-1}b$$

The following theorem proposes a method of splitting matrix $AH_k^{-1}A^T$ so that $v^k + \Delta v^k$ can be estimated through iterative calculation.

Theorem 1: Split $AH_k^{-1}A^T$ into two matrices M_k and N_k, where M_k is a diagonal matrix with $M_{ii} = \frac{1}{2}\sum_{j=1}^{n+p}\left|(AH_k^{-1}A^T)_{ij}\right|$, and $N_k = AH_k^{-1}A^T - M_k$. Then the sequence $\{\vartheta(t)\}$ updated according to (7) converges to $v^k + \Delta v^k$, i.e.the solution of (4a):

$$\vartheta(t+1) = -M_k^{-1}N_k\vartheta(t) + M_k^{-1}b_k \quad (7)$$

where $b_k = \left(Ax^k - AH_k^{-1}\nabla f(x^k)\right)$.

Proof: Let λ be any eigenvalue of $-M_k^{-1}N_k$ and $\mu \neq 0$ be the corresponding eigenvector so that:

$$(-M_k^{-1}N_k)\mu = \lambda\mu$$

Substituting $N_k = AH_k^{-1}A^T - M_k$ and multiplying $\mu^T M_k$ to the two sides, this yields:

$$\mu^T(M_k - AH_k^{-1}A^T)\mu = \lambda\mu^T M_k \mu$$

which implies

$$\lambda = 1 - \frac{\mu^T AH_k^{-1}A^T\mu}{\mu^T M_k \mu} \quad (8)$$

Since A is full row rank and H_k^{-1} is diagonal and element-wise positive, matrix $AH_k^{-1}A^T$ is symmetric and positive definite. By definition, M_k is also positive definite. Using the property of positive definite matrices, we obtain $\mu^T AH_k^{-1}A^T\mu > 0$ and $\mu^T M_k \mu > 0$, which imply $\lambda < 1$.

Next, substitute the value of matrix M_k, and we obtain:

$$\begin{aligned}\mu^T M\mu &= \sum_{i=1}^{n+p} M_{ii}\mu_i^2 = \tfrac{1}{2}\sum_{i=1}^{n+p}\sum_{j=1}^{n+p}\left|\left(AH_k^{-1}A^T\right)_{ij}\right|\mu_i^2 \\ &= \tfrac{1}{4}\sum_{i=1}^{n+p}\sum_{j=1}^{n+p}\left|\left(AH_k^{-1}A^T\right)_{ij}\right|\left(\mu_i^2+\mu_j^2\right) \\ &\geq \tfrac{1}{2}\sum_{i=1}^{n+p}\sum_{j=1}^{n+p}\left|\left(AH_k^{-1}A^T\right)_{ij}\right|\left(\mu_i\mu_j\right) \\ &> \tfrac{1}{2}\sum_{i=1}^{n+p}\sum_{j=1}^{n+p}\left(AH_k^{-1}A^T\right)_{ij}\left(\mu_i\mu_j\right) \\ &= \tfrac{1}{2}\mu^T AH_k^{-1}A^T\mu > 0\end{aligned} \tag{9}$$

By combining (8) and (9), it indicates $\lambda > -1$.

In conclusion, $|\lambda| < 1$, i.e. $\rho(-M_k^{-1}N_k) < 1$. According to *Lemma 1*, the proposition is proved. ■

Next we analyze that $v^k + \Delta v^k$ in (4a) can be solved in a decentralized fashion. The matrix $AH_k^{-1}A^T$ is calculated,

$$AH_k^{-1}A^T = \begin{bmatrix} KCK^T + GWG^T + EUE^T & GWR^T \\ RWG^T & RWR^T \end{bmatrix}$$

The details of $AH_k^{-1}A^T$ are shown in Fig. 2.

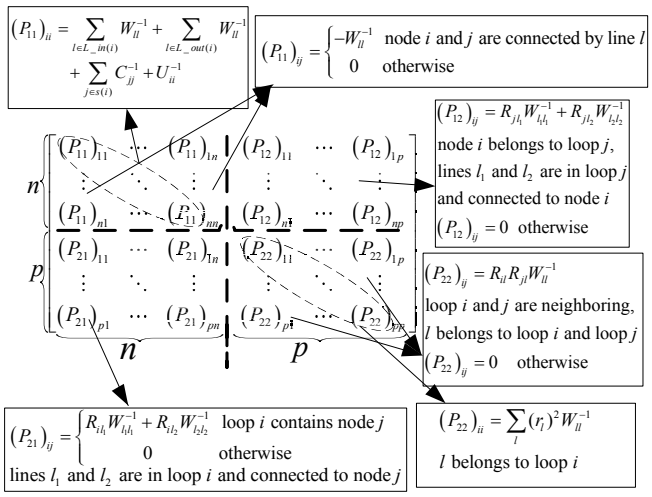

Figure 2. Details of each element in $AH_k^{-1}A^T$

It is observed that in the first n rows, the nonzero elements of the ith row are related to consumer and generators located at node i as well as its adjacent transmission lines, while in the last p rows, the nonzero elements of row $n+j$ are related to the transmission lines in loop j and its neighboring loops. Each node or master-node[1] can obtain all of this information locally. Further, based on the definition of M_k and N_k in *Theorem 1*, the elements in matrix $M_k^{-1}N_k$ have similar properties to those in $AH_k^{-1}A^T$. In addition, we find that the first n components and the last p components in vector b_k have the same formula as (1b) and (1c) respectively, replacing g_i, I_i and d_i with $g_i^k - \left(C_{ii}^k\right)^{-1}\nabla f\left(g_i^k\right)$, $I_i^k - \left(W_{ii}^k\right)^{-1}\nabla f\left(I_i^k\right)$, and $d_i^k - \left(U_{ii}^k\right)^{-1}\nabla f\left(d_i^k\right)$. These imply that given the values of x^k, λ_i^{k+1} $(i = 1, \cdots, n)$ can be estimated at node i, and master-node j is responsible for computing μ_j^{k+1} $(j = 1, \cdots, p)$, as shown in **Algorithm 1**.

Algorithm 1 distributed computation of $v^{k+1} = v^k + \Delta v^k$

1: **Pre-computation**.
Step 1: node i computes $\nabla f(g_j^k)$ and C_{jj}^{-1} for the generators j located at it; $\nabla f(I_l^k)$ and ${W_{ll}}^{-1}$ for its outlines l; $\nabla f(d_i^k)$ and $U_{ii}^{-1}$for the consumer i connected to it.
Step 2: node i communicates these computed values along with d_i^k, g_j^k and I_l^k to its neighboring nodes and the master-nodes of loops to which it belongs.
Step 3:
Node i computes $(M_k)_{ii}^{-1}(b_k)_i$, which is related to the generators, consumer and transmission lines connected to it;
Master-node j estimates $(M_k)_{tt}^{-1}(b_k)_t$, $t = n + j$. This value is influenced by the transmission lines in loop j.
2: **Initialization**. Node i initializes an arbitrary value for λ_i^{k+1}; Master-node j initializes μ_j^{k+1} randomly.
3: **repeat**
4: Node i communicates λ_i^{k+1} to its neighboring nodes and master-nodes of the loops to which it belongs;
Master-node j delivers μ_j^{k+1} to nodes belongs to its loop and the master-nodes of its neighboring loops.
5: Node i and Master-node j update λ_i^{k+1} and μ_j^{k+1} according to (7), respectively.
6: **until** predefined precision is achieved

C. Distributed Computation of Step-size

Backtracking line search is a general method for selecting step-size in Lagrange-Newton algorithm. Define

$$r(x, v) = (\nabla f(x) + A^T v; Ax)$$

and term it residual function which is used to measure the progress of the algorithm. If x^k, Δx^k, v^k and v^{k+1} are given, the step-size s^k is computed by following backtracking line search process:

step 1: Initialize $s^k = 1$

[1]Suppose each node knows to which meshes it belongs, and for each loop, a node termed master-node is selected to manage it when the smart grid is built. We also assume that master-node can communicate with nodes in the local loop and other master-nodes of neighboring loop conveniently

step 2: Update Repeatedly $s^k := \beta s^k$, if following inequality is true:

$$\left\|r(x^k + s^k\Delta x^k, v^{k+1})\right\| > \left(1 - \partial s^k\right)\left\|r(x^k, v^k)\right\|$$

where $\partial \in (0, 1/2)$, $\beta \in (0, 1)$. However, in our situation,

Algorithm 2 distributed computation of s^k at node i

1: Initialize $s^k = 1$, $\partial \in (0, 1/2)$, $\beta \in (0, 1)$, a positive constant η, and a positive scalar ψ that is large sufficient.
2: Initialize $\gamma_i^k(0)$, then $\overline{\|r\left(x^k, v^k\right)\|}$ is computed according to (10).
3: **while** (1) **do**
4: Initialize $\gamma_i^{k+1}(0)$
5: **if** Updated energy consumption of the consumer, generation capacity of the installed energy provider or current flow of the out-lines exceed feasible region **then**
6: Replace $\gamma_i^{k+1}(0)$ with $\overline{\|r\left(x^k, v^k\right)\|} + 3\eta$
7: **end if**
8: Compute $\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|}$ according to (10)
9: **if** $\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|} \approx \psi$ or $\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|} > \psi$ **then**
10: $s^k = \frac{s^k}{\beta}$
11: **Break**
12: **else if** $\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|} > (1 - \partial s^k)\overline{\|r\left(x^k, v^k\right)\|} + \eta$ **then**
13: $s^k = \beta s^k$
14: **else**
15: Replace $\gamma_i^{k+1}(0)$ with ψ and engage in the average consensus process.
16: **Break**
17: **end if**
18: **end while**

this backtracking line search process could not be applied directly. There are three critical considerations: 1) a distributed technique to compute $\|r\left(x, v\right)\|$ at each node; 2) a scheme to insure that the energy consumption amount of each consumer, generation capacities of energy providers, current flow in the transmission lines always satisfy (1d), (1e) and (1f); 3) a strategy to guarantee that the values d_i^k, g_i^k and I_i^k are updated by a same step-size. Thus, we propose an alternative distributed computation of step-size at each node based on average consensus method, shown as in **Algorithm 2**. Average consensus is a simple distributed and iterative scheme to estimate the value of $\|r\left(x, v\right)\|$ at each node [17]:

$$\overline{\|r\left(x, v\right)\|} = sqrt(n * \gamma_i(t)) \tag{10a}$$

$$\gamma_i(t+1) = \omega_i\gamma_i(t) + \sum_{j\in\chi(i)} \omega_j\gamma_j(t), \quad i = 1, \cdots, n \tag{10b}$$

where $\chi(i)$ is the set of neighbors of node i, and $\omega_j = \frac{1}{n}$, $\omega_i = 1 - \frac{\pi_i}{n}$, π_i denotes the number of neighbors of node i.

At $t = 0$, each node i initializes the value of $\gamma_i(0)$ as

$$\begin{aligned}\gamma_i(0) = {} & \nabla f(d_i) - \lambda_i + \sum_{j\in s(i)} \{\nabla f(g_j) + \lambda_i\} \\ & + \sum_{l\in L\_out(i)} (\nabla f(I_l) + \lambda_{l_{in}} - \lambda_i + \sum_{\ell\in m(l)} R_{\ell l}\mu_\ell) \\ & + \sum_{j\in s(i)} g_j + \sum_{j\in L\_in(i)} I_j - \sum_{j\in L\_out(i)} I_j - d_i\end{aligned} \tag{11}$$

If node i is selected as master-node, another component, i.e. $\sum_{l\in T(i)_+} r_l I_l - \sum_{l\in T(i)_-} r_l I_l$ should be added to $\gamma_i(0)$.

Since this distributed solution involves an iterative method, it has unavoidable error. Suppose the estimated value $\overline{\|r\left(x^k, v^k\right)\|}$ at node i satisfies

$$\left|\left\|r\left(x^k, v^k\right)\right\| - \overline{\|r\left(x^k, v^k\right)\|}\right| \le \varepsilon \tag{12}$$

where ε is a positive constant, and it assumes $2\varepsilon \le \eta$. Then the last two considerations above are elaborated. The convergence will be proved in the next section, with consideration of error in estimating $v^k + \Delta v^k$.

If the following inequality is true

$$\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| > \left(1 - \partial s^k\right)\left\|r\left(x^k, v^k\right)\right\| + 2\eta \tag{13}$$

Then, it is not difficult to deduce from (12) that the following inequality also holds at each node

$$\begin{aligned}\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|} &\ge \left(1 - \partial s^k\right)\overline{\|r\left(x^k, v^k\right)\|} - 2\varepsilon + 2\eta \\ &> \left(1 - \partial s^k\right)\overline{\|r\left(x^k, v^k\right)\|} + \eta\end{aligned}$$

This implies that when current step-size satisfies (13), each node will update the step-size simultaneously (line 13). According to line 5 and line 6 in **Algorithm 2**, if node i finds that current step-size could not guarantee x^{k+1} fall into the feasible region, it replaces $\gamma_i^{k+1}(0)$ with $\overline{\|r\left(x^k, v^k\right)\|} + 3\eta$. After such component replacement, we have

$$\begin{aligned}\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| &> \overline{\|r\left(x^k, v^k\right)\|} + 3\eta \\ &\ge \left\|r\left(x^k, v^k\right)\right\| - \varepsilon + 3\eta \\ &> \left\|r\left(x^k, v^k\right)\right\| + 2\eta \\ &> \left(1 - \partial s^k\right)\left\|r\left(x^k, v^k\right)\right\| + 2\eta\end{aligned}$$

Thereby, the step-size will be updated at each node. This explains the second consideration.

On the other hand, when

$$\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| \le \left(1 - \partial s^k\right)\left\|r\left(x^k, v^k\right)\right\| + \eta \tag{14}$$

we wish that all nodes would stop the backtracking line search. However, using (12), it yields

$$\overline{\|r\left(x^{k+1}, v^{k+1}\right)\|} \le \left(1 - \partial s^k\right)\overline{\|r\left(x^k, v^k\right)\|} + 2\varepsilon + \eta$$

which means some nodes would not stop searching the step-size, according to line 12 in **Algorithm 2**. To solve this issue, an extra step, i.e. line 15, is introduced. The value of ψ is large enough, e.g. much larger than $\max\|r\left(x, v\right)\|$, so that other nodes will realize they should stop searching the

step-size in the previous step and then take actions on the step-size (line 9 and line 10). There may be cases where no node stops searching the step-size when (14) holds, although it is very rare. Indeed, under our strategy, all nodes will achieve the same step-size once there is one node that stops searching by

$$\overline{\|r\left(x^{k+1},v^{k+1}\right)\|} \leq \left(1-\partial s^k\right)\overline{\|r\left(x^k,v^k\right)\|}+\eta$$

According to (12), in this situation, the searched step-size satisfies

$$\begin{aligned}\left\|r\left(x^{k+1},v^{k+1}\right)\right\| &\leq \left(1-\partial s^k\right)\left\|r\left(x^k,v^k\right)\right\|+2\varepsilon+\eta\\ &\leq \left(1-\partial s^k\right)\left\|r\left(x^k,v^k\right)\right\|+2\eta\end{aligned}$$

This completes the third consideration.

D. Distributed Algorithm for Optimizing Social-Welfare

In light of the above, the energy consumption of each consumer, and the generation capacities of energy providers in each time period could be decided in a distributed manner:

Preliminary. Before the next time slot starts, each consumer informs the connected node of its minimum and maximum demand requirements for this time slot, as well as the utility function. Likewise, the energy provider reports the maximum production and generation cost function to the node at which it is installed. Regarding the information on transmission line, it is fixed and known by its adjacent nodes.

Step 1: Node i initializes g_j, $j \in s(i)$, I_l, $l \in L\_out(i)$ and d_i randomly within the feasible region, as well as an arbitrary λ_i. It also initializes a random value for μ_i if it is a master-node.

Step 2: Node i updates λ_i according to **Algorithm 1**. It also updates μ_i if it is a master-node. Then the updated values are communicated to its neighboring nodes and master-nodes or master-nodes of neighboring loops.

Step 3: The step-size is estimated at each node according to **Algorithm 2**.

Step 4: The values of g_j, $j \in s(i)$, I_l, $l \in L\_out(i)$ and d_i are updated at node i, according to (6a), (6b) and (6d) respectively.

Step 5: Go to **Step 6** if predefined precision is achieved, otherwise go to **Step 2**.

Step 6: Node i informs the located consumer of the amount of energy it can use as well as the energy price i.e. λ_i for the next time slot, and requires the generator j which is installed there to provide g_j units of energy. Once the time slot starts, the ECC unit will control the consumer consuming d_i units energy, and the energy generation is controlled by the EGC.

V. Convergence analysis

This section shows that the proposed distributed Demand and Response algorithm for optimizing social-welfare is convergent even though there is error induced in the processes of **Algorithm 1** and **Algorithm 2**. First, a lemma that establishes the relation between $\left\|r(x^k,v^k)\right\|$ and $\left\|r(x^{k+1},v^{k+1})\right\|$ is presented [2].

Lemma 2: Denote the gradient matrix of $r(x,v)$ by $D(x,v)$, i.e.,

$$D(x,v)=\begin{bmatrix}\nabla^2 f(x) & A^T\\ A & 0\end{bmatrix}$$

Let the following two assumptions hold:
(a) (Lipschitz Condition) There exists some constant $Q>0$ such that

$$\|D(x,v)-D(\bar{x},\bar{v})\| \leq Q\,\|(x,v)-(\bar{x},\bar{v})\|\,\forall (x,v),(\bar{x},\bar{v})$$

(b) There exists some constant $M>0$ such that

$$\left\|D(x,v)^{-1}\right\| \leq M$$

Let (x^k,v^k) be the primal-dual vector at iteration k, then for any step-size rule θ^k, it has

$$\begin{aligned}\left\|r(x^{k+1},v^{k+1})\right\| \leq{}& (1-\theta^k)\left\|r(x^k,v^k)\right\|\\ &+M^2Q(\theta^k)^2\left\|r(x^k,v^k)\right\|^2\\ &+\theta^k\left\|\xi^k\right\|+M^2Q(\theta^k)^2\left\|\xi^k\right\|^2\end{aligned} \tag{15}$$

where ξ^k is the error vector at iteration k and it assumes that there exists a scalar $\xi>0$ such that $\left\|\xi^k\right\|\leq\xi$ for all k.

The next two subsections analyze convergence for the damped Newton phase and the quadratically convergent phase, respectively.

A. Convergence for Damped Newton Phase

In this subsection, we will show that when $\left\|r(x^k,v^k)\right\| \geq 1/2M^2Q$, one iteration process reduces $\|r\|$ by at least a certain minimum amount if the error scalars ξ is small enough, as quantified in the following:

$$\xi+M^2Q\xi^2\leq\eta \tag{16}$$

Further, it supposes that η in **Algorithm 3** is so small that $\eta \leq \frac{\partial\beta}{8M^2Q}$.

Define a step-size

$$\overline{s^k}=\frac{1}{2M^2Q\left\|r(x^k,v^k)\right\|}\leq 1$$

According to *Lemma 2*, (15) holds for any step-size rule, so does for $\overline{s^k}$. Substituting $\theta^k=\overline{s^k}$ in (15), it yields

$$\begin{aligned}\left\|r(x^{k+1},v^{k+1})\right\| &\leq \left\|r(x^k,v^k)\right\|-\tfrac{1}{4M^2Q}\\ &\quad+\overline{s^k}\left\|\xi^k\right\|+M^2Q(\overline{s^k})^2\left\|\xi^k\right\|^2\\ &\leq \left\|r(x^k,v^k)\right\|-\tfrac{1}{4M^2Q}+\xi+M^2Q\xi^2\\ &=\left(1-\tfrac{\overline{s^k}}{2}\right)\left\|r(x^k,v^k)\right\|+\xi+M^2Q\xi^2\\ &\leq\left(1-\partial\overline{s^k}\right)\left\|r(x^k,v^k)\right\|+\eta\end{aligned}$$

where the second inequality follows by the facts $\overline{s^k} < 1$, $\left\|\xi^k\right\| \leq \xi$, for all k, while the last one follows by (16) and the fact $\partial \in (0, 0.5)$. This result shows that $\overline{s^k}$ satisfies the line search exit condition of our algorithm. Therefore, we have $s^k \geq \beta\overline{s^k}$, where s^k is the searched step-size. From the previous section, when the step-size is searched, the relationship between $\left\|r(x^k, v^k)\right\|$ and $\left\|r(x^{k+1}, v^{k+1})\right\|$ satisfies:

$$\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| \leq \left(1 - \partial s^k\right)\left\|r\left(x^k, v^k\right)\right\| + 2\eta$$

It yields

$$\begin{aligned}\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| &\leq \left(1 - \partial\beta\overline{s^k}\right)\left\|r\left(x^k, v^k\right)\right\| + 2\eta \\ &= \left\|r\left(x^k, v^k\right)\right\| - \tfrac{\partial\beta}{2M^2Q} + 2\eta \\ &\leq \left\|r\left(x^k, v^k\right)\right\| - \tfrac{\partial\beta}{4M^2Q}\end{aligned}$$

The result shows that we obtain a minimum decrease of $\frac{\partial\beta}{4M^2Q}$ in the norm of residual function per iteration, as long as $\left\|r\left(x^k, v^k\right)\right\| \geq \frac{1}{2M^2Q}$. This indicates that it takes at most

$$\frac{4\left\|r\left(x^0, \omega^0\right)\right\| M^2Q}{\partial\beta}$$

iterations before $\left\|r\left(x^k, v^k\right)\right\| \leq \frac{1}{2M^2Q}$ is reached.

B. Convergence for Quadratical Phase

When $\left\|r\left(x^k, v^k\right)\right\| < \frac{1}{2M^2Q}$, substituting $\theta^k = 1$ in (15), it yields

$$\begin{aligned}\left\|r\left(x^{k+1}, v^{k+1}\right)\right\| &\leq \tfrac{1}{2}\left\|r\left(x^k, v^k\right)\right\| + \xi + M^2Q\xi^2 \\ &\leq (1 - \partial)\left\|r\left(x^k, v^k\right)\right\| + \eta\end{aligned}$$

which implies that the searched step-size is $s^k = 1$. In the situation where $\left\|r\left(x^k, v^k\right)\right\| < \frac{1}{2M^2Q}$ and $s^k = 1$, literature [2] has proved

$$\lim_{k\to\infty}\left\|r\left(x^k, v^k\right)\right\| \leq B + \frac{\delta}{2M^2Q}$$
$$B = \xi + M^2Q\xi^2$$

with further assumption

$$B + M^2QB^2 \leq \frac{\delta}{4M^2Q}$$

where δ is a constant satisfies $\delta \in (0, 0.5)$.

VI. Performance Evaluation

In this section, we present simulation results and analyze the performance of the proposed distributed DR algorithm for smart grid. The simulator is developed using R software [18] which is open source. In the simulation model, we consider quadratic utility function for consumer and quadratic cost function for energy generator [9], i.e.:

$$u_i(d_i) = \begin{cases}\varphi_i d_i - \dfrac{\alpha}{2}d_i^2, & 0 \leq d_i \leq \frac{\varphi_i}{\alpha} \\ \dfrac{{\varphi_i}^2}{2\alpha}, & d_i \geq \frac{\varphi_i}{\alpha}\end{cases} \qquad i = 1, \cdots, n \tag{17a}$$

Table I
Parameters for Proposed Problem

Consumer	Generator	Transmission line
$d_i^{\max} = rnd\,[25, 30]$ [1]	$g_i^{\max} = rnd\,[40, 50]$	$I_i^{\max} = rnd\,[20, 25]$
$d_i^{\min} = rnd\,[2, 6]$	$a_i = rnd\,[0.01, 0.1]$	$c = 0.01$
$\varphi_i = rnd\,[1, 4]$		
$\alpha = 0.25$		

[1] $x = rnd\,[x_1, x_2]$ denotes that the value of x is selected uniformly from the interval$[x_1, x_2]$

and

$$c_i(g_i) = a_i g_i^2 \qquad i = 1, \cdots, m \tag{17b}$$

where α is a pre-defined parameter, and φ_i is a parameter reflecting the consumer preference of energy consumption, so it may vary among consumers and also at different time slots during the day. Similarly, the parameter a_i that describes the performance of energy providers vary among different energy generators and some factors, e.g. weather conditions. The parameters corresponding to *Problem 1*, (17a) and (17b) are given in Table I. In the simulation below, the initial values of all dual variables are one, and the initial values of primal variables are defined as follows:

$$g_i = 0.5g_i^{max} \qquad i = 1, \cdots, m$$
$$I_i = 0.5I_i^{max} \qquad i = 1, \cdots, L$$
$$d_i = 0.5(d_i^{min} + d_i^{max}) \qquad i = 1, \cdots, n$$

We analyze the proposed algorithm through multi-aspects:

1. Verify the correctness of the distributed Demand and Response algorithm;

2. Study the computation accuracy of dual variables influences the social-welfare; The computation accuracy in the form of residual function is also considered;

3. Analyze the communication overhead;

4. Analyze how the smart grid scale influences the performance of the distributed Demand Response algorithm.

In the first three situations, the smart grid system consists of 20 nodes, 32 transmission lines and 13 independent loops. There are 20 consumers and 12 energy generators.

A. Correctness Verification

To verify the correctness of the proposed distributed algorithm, the iterations of computing dual variables and the form of residual function are large enough. It is compared with the Rdonlp2 solution [19] which is an R package for solving nonlinear programming problems. Simulation results are shown in Fig. 3 and Fig. 4. As illustrated in Fig. 3, after about 35 Lagrange-Newton iterations, the maximum social-welfare is close to the optimal value given by the Rdonlp2 solution. Figure 4 shows the distributed results, i.e. energy provided by each generator(variables 1-12), the current flow through each transmission line (variables 13-44) and the energy consumed by each consumer(variables

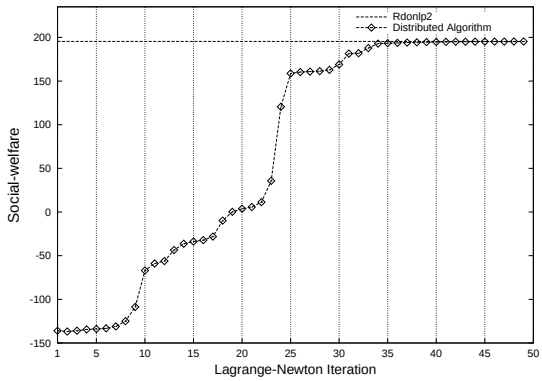

Figure 3. Social-welfare comparison (distributed vs centralized).

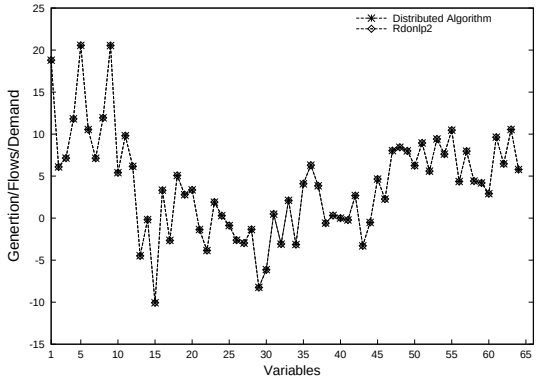

Figure 4. Generation/flows/demand results comparison (distributed vs centralized).

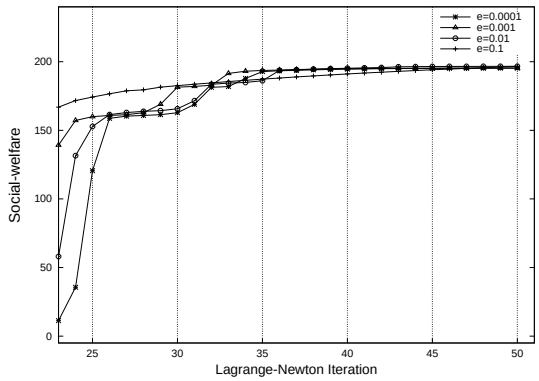

Figure 5. The impact of computation accuracy of dual variables on social-welfare.

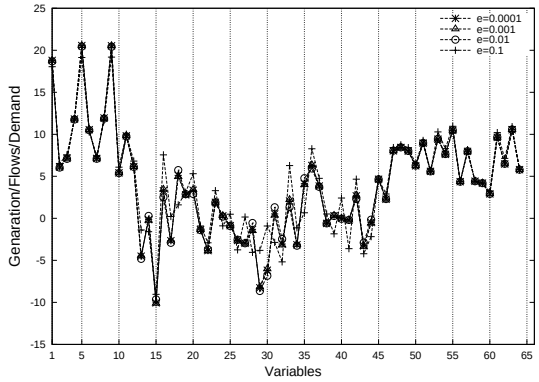

Figure 6. The impact of computation accuracy of dual variables on generation/flows/demand.

45-64). These results are also close to Rdonlp2 solution. Hence, using the proposed distributed algorithm, the energy consumption, generation and transmission can be determined locally so that the social-welfare is maximized. Further, the LMPs are also estimated during this distributed algorithm. As a result, the proposed algorithm provides a potential scheme for energy transactions among users in the future smart grid.

B. Impact of Computation Error

As analyzed previously, estimating dual variables and step-size involves unavoidable errors. The error e is formulated as: $e = \left|\frac{(z-\overline{z})}{z}\right|$, where $\overline{z}$ is the estimated value, and z is true value.

The results under different computation errors of dual variables are shown in Fig. 5-Fig. 6. The computation error in the form of residual function is set at 0.001. Both the energy generation/transmission flow/demand values and social-welfare are almost equal when the computation error is less than 0.01. Shown in Fig. 5, the convergency speed of Lagrange-Newton method is also close if the computation error is less than 0.01. However, the computation results deviate from normal values if the error achieves 0.1.

Figure 7-Fig. 8 show the impact of computation accuracy of the form of residual function. In this case, the computation error of dual variables is fixed at 0.0001. From these figures, the values of generation/transmission flow/demand and social-welfare are not affected by the computation error of the form of residual function. This indicates that the proposed algorithm is robust when the computation error in the form of residual function is within a certain region.

C. Communication Traffic Analysis

According to the results above, the communication traffic of each node is determined by: the convergence rate of the Lagrange-Newton method, the convergence speed of dual variables and the form of residual function. Figure 9 and Fig. 10 show average iteration times for computing dual variables and the form of residual function, respectively. In addition, we should point out that it may execute more than one computation of the form of residual function during each Lagrange-Newton iteration (see **Algorithm 2**). In this simulation case, it executes an average ten computations of the form of residual function in each Lagrange-Newton iteration. Overall, each node would exchange several thousands of messages with its neighbors. It seems that these results are

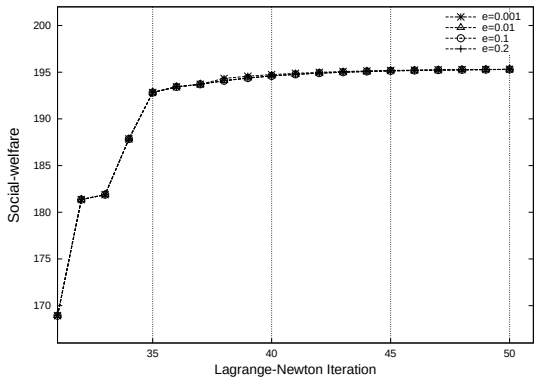

Figure 7. The impact of computation error in the form of residual function on social-welfare. The curves of the four iteration processes almost overlap.

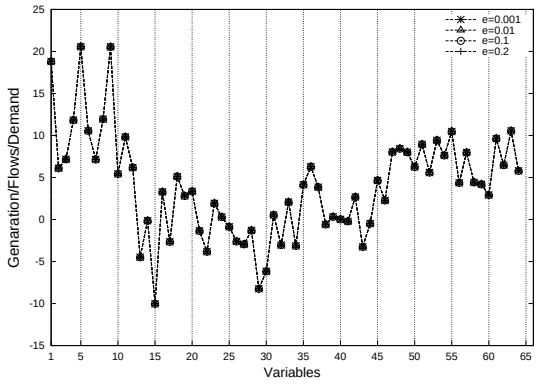

Figure 8. The impact of computation error in the form of residual function on generation/flows/demand.

unsatisfactory, since it is desirable to determine the values of d_i, g_i and I_i quickly before the next time slot starts so that the smart grid can always run in an optimum state. However, the proposed algorithm can be further improved as follows:

- Indeed, the convergence rate of dual variables is determined by the spectral radius of $-M_k^{-1}N_k$ in *Theorem 1*, while the coefficient ω in (10b) controls the computation of step-size. Therefore, it is critical to find a favorable split method for matrix $AH_k^{-1}A^T$ and coefficients ω to improve the whole algorithm rate in smart grid.
- We have mentioned that it executes an average ten computations of the form of residual function each Lagrange-Newton iteration. However, most computations are used to guarantee that the next updating results fall into the feasible region, shown as in Fig. 11. The algorithm rate would be improved a lot if we can find a method to initialize a step-size that is feasible.

D. Algorithm Scalability Analysis

Figure 12 is the results of different smart grid scales. In the simulation, the distributed Lagrange-Newton process

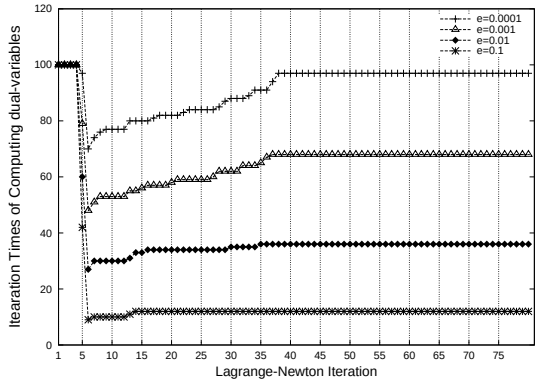

Figure 9. The iteration times of computing dual variables in different computation errors. The maximum iteration times is fixed at 100.

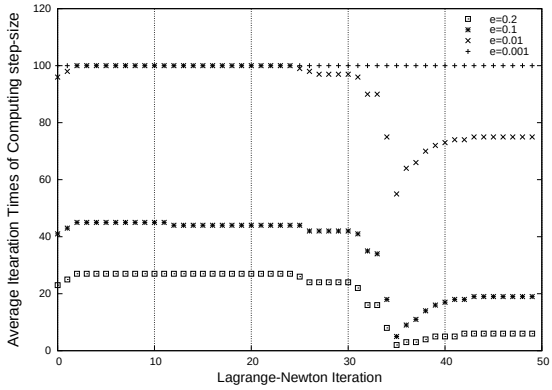

Figure 10. The average iteration times of computing the residual function's form in different computation errors. The maximum iteration times is fixed at 100.

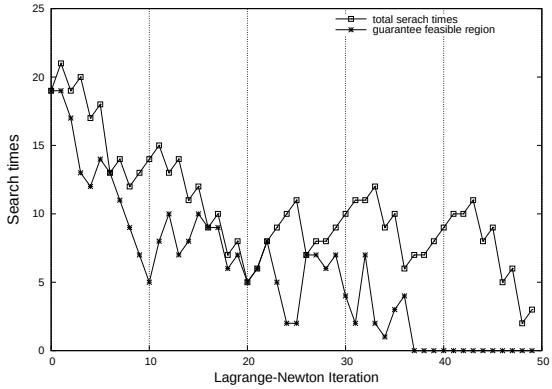

Figure 11. Step-size search times during each Lagrange-Newton iteration.

stops when the relative error between distributed Lagrange-Newton result and the value obtained by the Rdonlp2 solution is less than 0.005. In addition, the relative error between two consecutive iterations should also be less than 0.001. The required relative errors in estimating dual variables and step-size are 0.01, while the allowed maximum iteration

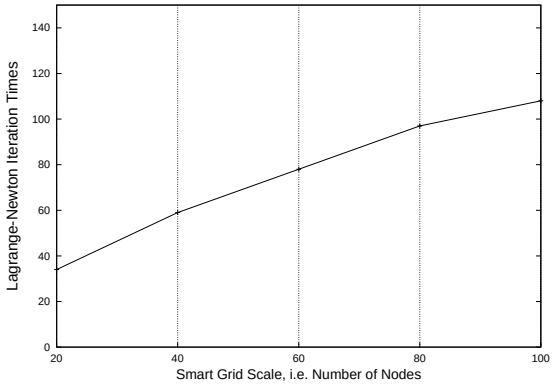

Figure 12. The results of different smart grid scales.

times of computing dual variables and the form of residual function are fixed at 100 and 200, respectively. We observed that the relative errors in estimating dual variables and step-size could not achieve 0.01 as the number of nodes increases. However, the values of generation/transmission flow/demand and social-welfare still approximately converge to the values obtained by the Rdonlp2 solution. This indicates that the computation and communication traffic at each node are mainly influenced by the convergence rate of the Lagrange-Newton method within a certain smart grid scale.

VII. Conclusions

In this paper, we propose a distributed Demand and Response algorithm for smart grid with the objective of maximizing social-welfare. The proposed algorithm is run periodically. Before the next time slot starts, the energy consumption amount of each consumer, and the generation of energy providers are determined locally, through information exchange with neighbors. The simulation verified the correctness of the proposed distributed algorithm. However, the computation rate and communication traffic is still high from a system's viewpoint, although the convergence speed of Lagrange-Newton algorithm is quadratic. How to significantly reduce communication costs in real systems remains a challenge and an area for future investigation.

References

[1] M. H. Albadi, and E. F. El-Saadany, "Demand Response in Electricity Markets: An Overview," in *Proc.2007 IEEE Power Engineering Society General Meeting*, Tampa, FL, pp.1-5.

[2] A. Jadbabaie, A. Ozdaglar, and M. Zargham, "A Distributed Newton Method for Network Optimization," in *Proc. of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conf.*, Shanghai, pp. 2736 - 2741, Dec. 2009.

[3] E. Wei, A. Ozdaglar, and A. Jadbabaie. "A distributed newton method for network utility maximization," Lab. for Information and Decision Systems, MIT, Tech. Rep. LIDS-2832, 2010.

[4] T. Zheng, and E. Litvinov, "Ex-post pricing in the co-optimized energy and reserve market," *IEEE Trans. Power Systems*, vol. 21, no. 4, pp. 15281538, 2006.

[5] S. Kishore, and L. V. Snyder, "Control Mechanisms for Residential Electricity Demand in SmartGrids," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 443-448, Oct. 2010.

[6] S. Hatami, and M. Pedram, "Minimizing the Electricity Bill of Cooperative Users under a Quasi-Dynamic Pricing Model," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 421-426, Oct. 2010.

[7] D. O'Neill, M. Levorato, A. Goldsmith, and U. Mitra, "Residential Demand Response Using Reinforcement Learning," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 409-414, Oct. 2010.

[8] S. Caron, and G. Kesisdis, "Incentive-based Energy Consumption Scheduling Algorithms for the Smart Grid," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 391-396, Oct. 2010.

[9] P. Samadi, A. H. Mohseian-Rad, R. Schober, V. W. S. Wong, and J. Jatskevich, "Optimal Real-time Pricing Algorithm Based on Utility Maximization for Smart Grid," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 415-420, Oct. 2010.

[10] M. Rooabehani, M. Dahleh, and S. Mitter, "Dynamic Pricing and Stabilization of Supply and Demand in Modern Electric Power Grids," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 543-548, Oct. 2010.

[11] A. Kiani, A. Annaswamy, "Perturbation analysis of market equilibrium in the presence of renewable energy resources and demand response," *IEEE Innovative Smart Grid Technologies Conference Europe*, Gothenburg, pp.1-8, Oct. 2010.

[12] C. Ibars, M. Navarro, and L. Giupponi "Distributed Demand Management in Smart Grid with a Congestion Game," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 495-500, Oct. 2010.

[13] M. A. A. Pedrasa, T. D. Spooner, and I. F. MacGill, "Coordinated Scheduling of Residential Distributed Energy Resources to Optimize Smart Home Energy Services," *IEEE Trans. Smart Grid*, vol. 1, no. 2, pp. 134-143, Sept. 2010.

[14] L. Chen, N. Li, S. H. Low, and J. C. Doyle, "Two Market Models for Demand Response in Power Networks," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 397-402, Oct. 2010.

[15] V. Bakker, M. G. C. Bosman, A. Molderink, J. L. Hurink, and G. J. M. Smit, "Demand side load management using a three step optimization methodology," in *Proc. of the First IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersbur, MD, pp. 431-436, Oct. 2010.

[16] S. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004, pp. 531-533.

[17] X. Lin, S. Boyd, and S. Lall, "A Scheme for Robust Distributed Sensor Fusion Based on Average Consensus," in *Proc. of the Fourth International Conference on Information Processing in Sensor Networks*, Los Angeles, California, USA, 2005, pp. 63-70.

[18] The R Project for Statistical Computing, [Online]. Avaiable: http://www.r-project.org/

[19] Rdonlp2-an R interface to DONLP2, [Online]. Avaiable: http://arumat.net/Rdonlp2/