

Fully Decentralized AC Optimal Power Flow Algorithms

Andy X. Sun

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA
Email: andy.sun@isye.gatech.edu

Dzung T. Phan

Soumyadip Ghosh
Business Analytics and Mathematical Sciences
IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598, USA
Emails: phandu@us.ibm.com, ghoshs@us.ibm.com

Abstract—Motivated by the increasing complexity in the control of distribution level electric power systems especially in a smart grid environment, we propose fully decentralized algorithms to solve alternating current (AC) optimal power flow (OPF) problems. The key feature of the proposed algorithms is a complete decentralization of computation down to nodal level. In this way, no central or sub-area controller is needed, and the OPF problem is solved by individual nodes, which only have local knowledge of the system. Preliminary results show promising performance of the fully decentralized algorithms.

I. INTRODUCTION

In this paper, we propose new decentralized and distributed algorithms to solve the optimal power flow problem in electric power systems. The OPF problem is one of the most important decision problems in power system operation, where generators output levels are decided in order to minimize system-wide production cost subject to power balance constraints, network flow constraints, and other physical constraints. In the current practice, the system operator as a central control entity gathers extensive information about generators' physical characterization and cost structure, load forecast, and electric network parameters, and solves the OPF problem in a centralized manner.

However, the electric industry is experiencing fundamental changes. Especially in the recent decade, an increasing number of renewable energy resources, distributed generators, electric vehicle charging stations, storage devices, and demand-response components are being integrated into the power system. As this trend continues, the number of control objects in a power system can become very large, the nature of control objects will be significantly more diverse, and the interaction between control objects and the central controller can be much more complex and dynamic. This brings new challenges to the traditional centralized control framework. Decentralized and distributed algorithms for solving OPF problems present a desirable alternative control scheme.

The key feature of our proposed distributed algorithms is that the computation is fully localized to *nodal* level in the power network, so is the coordination fully localized to interaction between neighbor nodes. Thus, no central or sub-area controller is needed. The OPF problem is

completely solved by individual nodes, which only have local knowledge of the system. Therefore, we use the name *fully* decentralized OPF algorithms.

Early work in the related areas considered parallelizing certain computation such as matrix factorization in centralized power flow algorithms [1], [2], [3], [4]. Baldick et al. [5], [6] proposed a regional decomposition approach, where a large power system is decomposed into a few overlapping subsystems and each subsystem's problem is solved in parallel. More recent work explored similar ideas of regional decomposition to solve multi-area coordination problem in an interconnected market environment ([7], [8] and references therein). Both [5], [6] and [7] have focused on transmission level bulk power systems such as the ERCOT power system in the United States and the continental European power systems, and concerned with coordination issues between regional markets, where each regional market can be a huge control area by itself and needs a sub-control center to solve its own OPF problem. Our work is different in that no sub-system controller nor global information is needed.

Most recently, distributed OPF algorithms were proposed in [9], where the non-convex OPF problem is first relaxed to semidefinite programming (SDP) problems using convexification techniques proposed in [10], [11], [12], [13], then the resulting SDP problems are decomposed in terms of maximal cliques (i.e. completely connected subgraphs) in a modified network using a sparsity technique first proposed for parallel solution of general SDP problems [14]. Their algorithms require all the nodes in a clique to jointly solve a SDP subproblem, and share information with adjacent cliques. Although maximal cliques of a graph could be computed a priori, it requires centralized computation and the knowledge of the entire power network. Such computation and information gathering can be quite burdensome for large-scale systems, especially when the topology of the network may change frequently as observed in today's practice, due to unexpected loss of transmission lines, maintenance, or network expansion. In comparison, our proposed algorithms can automatically adapt to network topological changes, and no a priori computation or information gathering is needed. In addition, since the decomposition in [9] is decided by the structure of the network, the algorithm designer does not have full control over the structure of the resulted subsystems. In contrast,

our algorithms retain full flexibility to re-group nodes into subsystems to form a hierarchy of coarser grained decompositions.

Our work is also different from previous proposals in terms of the algorithmic framework. [5] and [6] form and decompose an augmented Lagrangian using auxiliary problem principle proposed in [15]. [7] and related work decouple the first-order KKT optimality conditions of the original large-scale problem into KKT conditions of sub-area systems. The recent work [9] uses a simple Lagrangian-relaxation algorithm. We form a completely separable augmented Lagrangian and apply the alternating direction method of multipliers (ADMM), which has recently gained considerable popularity due to its simplicity and comparable or better performance compared to various state-of-the-art algorithms for distributed optimization and statistical learning problems [16].

The paper is organized as follows. In Section II, we introduce mathematical formulations of AC OPF problems. Then, we outline the basic framework of ADMM. In Section III, we present fully decentralized algorithms for AC OPF. Section IV presents preliminary computational results for both radial and meshed networks and some discussions. Section V concludes the paper and points out future research directions.

II. BACKGROUND

In this section, we first review the AC optimal power flow formulation. Then, we outline the basic framework of alternating direction method of multipliers.

Let \mathcal{N} denote the set of nodes in a power network. Let P_i^G, Q_i^G be the real and reactive powers produced by generator at node i , and P_i^D, Q_i^D be the real and reactive powers consumed by load at node i . The complex voltage at node i can be represented by its real and imaginary parts as e_i, f_i . Let $\delta(i)$ be the set of nodes connected to i , and g_{ij} and b_{ij} denote the conductance and susceptance of the branch connecting nodes i, j . Define g_{ii} and b_{ii} as the self-conductance and the self-susceptance at node i , respectively.

The rectangular formulation of the AC OPF is given as below:

$$\min_{P^G, Q^G, e, f} \sum_{i \in \mathcal{N}} f_i(P_i^G) \quad (1)$$

$$P_i^G - P_i^D = \sum_{j \in \mathcal{N}} [e_i(g_{ij}e_j - b_{ij}f_j) + f_i(g_{ij}f_j + b_{ij}e_j)], \quad \forall i, \quad (2)$$

$$Q_i^G - Q_i^D = \sum_{j \in \mathcal{N}} [f_i(g_{ij}e_j - b_{ij}f_j) - e_i(g_{ij}f_j + b_{ij}e_j)], \quad \forall i, \quad (3)$$

$$\underline{P}_i \leq P_i^G \leq \bar{P}_i, \quad \forall i \quad (4)$$

$$\underline{Q}_i \leq Q_i^G \leq \bar{Q}_i, \quad \forall i \quad (5)$$

$$\underline{V}_i^2 \leq e_i^2 + f_i^2 \leq \bar{V}_i^2, \quad \forall i \quad (6)$$

where $f_i(P_i^G)$ is the variable production cost of generator i , assuming to be a convex quadratic function. Eq. (2) and (3) are power flow equations for real and reactive power,

respectively. Constraints (4)-(6) specify bounds on the output ranges of generators, and voltage levels at nodes. If node i does not have a generator, we set the lower and upper bounds of P_i^G and Q_i^G to be zero.

AC OPF can also be formulated in the polar form using voltage magnitude and angles [17]. We propose decentralized algorithms for both. In the interest of space, we only present the rectangular formulation. Numerical results for both formulations are shown in Section IV.

In the following, we briefly outline the ADMM algorithm. See [16] for a comprehensive discussion. ADMM inherits the decomposability of dual ascent type algorithms and the good convergence properties of the method of multipliers. It intends to solve the following problem

$$\begin{aligned} \min_{x, y} \quad & f(x) + g(y) \\ \text{s.t.} \quad & Ax + By = c. \end{aligned}$$

Introducing dual variable λ and a positive penalty parameter ρ , the augmented Lagrangian is formed as

$$\begin{aligned} L_\rho(x, y, \lambda) = & f(x) + g(y) + \lambda^T(Ax + By - c) \\ & + \frac{\rho}{2} \|Ax + By - c\|_2^2. \end{aligned}$$

ADMM consists of the following iterations:

$$\begin{aligned} x^{k+1} &:= \arg \min_x L_\rho(x, y^k, \lambda^k), \\ y^{k+1} &:= \arg \min_y L_\rho(x^{k+1}, y, \lambda^k), \\ \lambda^{k+1} &:= \lambda^k + \rho(Ax^{k+1} + By^{k+1} - c), \end{aligned}$$

where the first two steps minimize over primal variables x, y , and the third step updates the dual variable λ with step size $\rho > 0$.

III. FULLY DECENTRALIZED OPF ALGORITHM

In this section, we propose a fully decentralized algorithm for AC OPF problem.

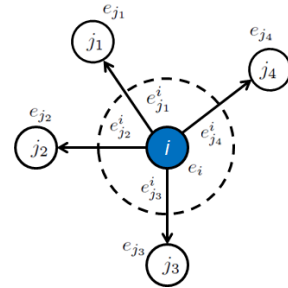


Fig. 1. A set of voltage variables e_j^i, f_j^i are introduced at each node. Local variables controlled by node i are encircled by the dashed line.

The difficulty in developing a decentralized algorithm lies in power flow equations (2) and (3), which induce couplings between nodal voltages. In order to decompose computation down to individual nodes, we need to be able to separate nodal variables and let each node solve its own problem and coordinate through local interactions. To accomplish this, at each node i , we introduce a set of slack variables e_j^i, f_j^i for each neighbor node j to represent

the real and imaginary voltages of j “observed” at node i . Figure 1 illustrates this construction.

The reformulation using this idea is presented below.

$$(P_1) \min \sum_{i \in \mathcal{N}} f_i(P_i^G) \quad (7)$$

$$\text{s.t. } P_i^G - P_i^D = (e_i^2 + f_i^2)g_{ii} + \quad (8)$$

$$\sum_{j \in \delta(i)} [e_i(g_{ij}e_j^i - b_{ij}f_j^i) + f_i(g_{ij}f_j^i + b_{ij}e_j^i)], \quad \forall i$$

$$Q_i^G - Q_i^D = -(e_i^2 + f_i^2)b_{ii} + \quad (9)$$

$$\sum_{j \in \delta(i)} [f_i(g_{ij}e_j^i - b_{ij}f_j^i) - e_i(g_{ij}f_j^i + b_{ij}e_j^i)], \quad \forall i$$

$$\underline{P}_i \leq P_i^G \leq \bar{P}_i, \quad \forall i, \quad (10)$$

$$\underline{Q}_i \leq Q_i^G \leq \bar{Q}_i, \quad \forall i, \quad (11)$$

$$\underline{V}_i^2 \leq e_i^2 + f_i^2 \leq \bar{V}_i^2, \quad \forall i, \quad (12)$$

$$e_j = e_j^i, \quad \forall i, \forall j \in \delta(i), \quad (13)$$

$$f_j = f_j^i, \quad \forall i, \forall j \in \delta(i). \quad (14)$$

Notice that the “observed” voltages e_j^i, f_j^i should be equal to the true values e_j, f_j , respectively, thus we have constraints (13) and (14).

Now we can arrange variables at each node i into two groups, namely $x_i := (P_i^G, Q_i^G, e_i, f_i) \in \mathbb{R}^4$ and $y_i := (e_j^i, f_j^i, \forall j \in \delta(i)) \in \mathbb{R}^{2|\delta(i)|}$. Constraints (8)-(12) define a feasible region Ω_i for (x_i, y_i) of each node i . We introduce an indicator function $\mathbf{1}_{\Omega_i}(x_i, y_i)$, which equals to 1 if $(x_i, y_i) \in \Omega_i$, and $+\infty$ otherwise. Thus, the above formulation (P_1) can be written as

$$(P_2) \min_{x, y} \sum_{i \in \mathcal{N}} [f_i(x_i) + \mathbf{1}_{\Omega_i}(x_i, y_i)],$$

$$\text{s.t. } e_j = e_j^i, \quad \forall i, \forall j \in \delta(i), \quad (15)$$

$$f_j = f_j^i, \quad \forall i, \forall j \in \delta(i). \quad (16)$$

Dualizing constraints (15) and (16), we can form the augmented Lagrangian as

$$L(x, y, \lambda, \mu) = \sum_{i \in \mathcal{N}} [f_i(x_i) + \mathbf{1}_{\Omega_i}(x_i, y_i)] +$$

$$\sum_{i \in \mathcal{N}} \sum_{j \in \delta(i)} [\lambda_j^i (e_j - e_j^i) + \mu_j^i (f_j - f_j^i)] +$$

$$\frac{\rho}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \delta(i)} [(e_j - e_j^i)^2 + (f_j - f_j^i)^2].$$

Now we can see that, using ADMM algorithm to alternate between primal variables x and y and update dual variables λ and μ , the augmented Lagrangian can be completely decomposed to subproblems at each node. Therefore, we have a fully decentralized OPF algorithm. The detail is presented below.

At the k -th iteration:

- 1) Fix $(e_j^j)^k, (f_j^j)^k$ and the multipliers $(\lambda_j^j)^k, (\mu_j^j)^k$ for all i, j . We have the following system-wide

problem:

$$\min_{P_i^G, Q_i^G, e_i, f_i} L(P_i^G, Q_i^G, e_i, f_i, y^k, \lambda^k, \mu^k),$$

which can be completely decentralized to nodes. Each node i solves the following problem:

$$\min_{P_i^G, Q_i^G, e_i, f_i} f_i(P_i^G) +$$

$$\sum_{j \in \delta(i)} [(\lambda_j^j)^k (e_i - (e_j^j)^k) + (\mu_j^j)^k (f_i - (f_j^j)^k)] +$$

$$\sum_{j \in \delta(i)} \frac{\rho}{2} [(e_i - (e_j^j)^k)^2 + (f_i - (f_j^j)^k)^2]$$

$$\text{s.t. } P_i^G - P_i^D = (e_i^2 + f_i^2)g_{ii} +$$

$$\sum_{j \in \delta(i)} [e_i (g_{ij}(e_j^j)^k - b_{ij}(f_j^j)^k) + f_i (g_{ij}(f_j^j)^k + b_{ij}(e_j^j)^k)],$$

$$Q_i^G - Q_i^D = -(e_i^2 + f_i^2)b_{ii} +$$

$$\sum_{j \in \delta(i)} [f_i (g_{ij}(e_j^j)^k - b_{ij}(f_j^j)^k) - e_i (g_{ij}(f_j^j)^k + b_{ij}(e_j^j)^k)],$$

$$\underline{P}_i \leq P_i^G \leq \bar{P}_i, \underline{Q}_i \leq Q_i^G \leq \bar{Q}_i, \underline{V}_i^2 \leq e_i^2 + f_i^2 \leq \bar{V}_i^2.$$

Notice that this nodal problem is a quadratic optimization problem in only 4 variables. Even though it is non-convex due to the power flow constraints, it still can be solved very fast. To solve this problem, each node i needs its own estimation of its neighbor nodes voltages $((e_j^j)^k, (f_j^j)^k)$, as well as its neighbor nodes' estimation of i 's voltages $((e_i^i)^k, (f_i^i)^k)$ and local multipliers $((\lambda_j^j)^k, (\mu_j^j)^k)$ for all $j \in \delta(i)$. The first set of voltage information is available at node i , while the second set of voltage information and the Lagrangian multipliers require a local message passing between node i and its neighbors $j \in \delta(i)$. Denote the solution of this problem as $(x_i)^{k+1} = ((P_i^G)^{k+1}, (Q_i^G)^{k+1}, (e_i)^{k+1}, (f_i)^{k+1})$. Upon solving this problem, node i shares this solution $(x_i)^{k+1}$ with its neighbors.

- 2) Fix x^{k+1} and (λ^k, μ^k) . We have the following system-wide problem:

$$\min_y L(x^{k+1}, y, \lambda^k, \mu^k),$$

which can be fully decomposed to subproblems at each node i as

$$\min_{e_j^i, f_j^i, \forall j \in \delta(i)} \sum_{j \in \delta(i)} [(\lambda_j^j)^k (\alpha_j - e_j^i) + (\mu_j^j)^k (\beta_j - f_j^i)]$$

$$+ \sum_{j \in \delta(i)} \frac{\rho}{2} [(\alpha_j - e_j^i)^2 + (\beta_j - f_j^i)^2]$$

$$\text{s.t. } (P_i^G)^{k+1} - P_i^D = (\alpha_i^2 + \beta_i^2)g_{ii} +$$

$$\sum_{j \in \delta(i)} [\alpha_i (g_{ij}e_j^i - b_{ij}f_j^i) + \beta_i (g_{ij}f_j^i + b_{ij}e_j^i)],$$

$$(Q_i^G)^{k+1} - Q_i^D = -(\alpha_i^2 + \beta_i^2)b_{ii} +$$

$$\sum_{j \in \delta(i)} [\beta_i (g_{ij}e_j^i - b_{ij}f_j^i) - \alpha_i (g_{ij}f_j^i + b_{ij}e_j^i)],$$

where $\alpha_i := (e_i)^{k+1}, \beta_i := (f_i)^{k+1}$ and $\alpha_j := (e_j)^{k+1}, \beta_j := (f_j)^{k+1}$ for all $j \in \delta(i)$. Notice that this is a convex quadratic optimization problem in $2|\delta(i)|$ variables, which can be solved very efficiently. To solve this problem, node i needs its neighbor nodes' updated voltages $((e_j)^{k+1}, (f_j)^{k+1})$ and multipliers $((\lambda_j^i)^k, (\mu_j^i)^k)$. The information about $((e_j)^{k+1}, (f_j)^{k+1})$ can be shared through a local message passing between node i and its neighbors, while the multipliers information can be processed and stored at i .

- 3) Update multipliers (λ_j^i, μ_j^i) at each node i ,

$$\begin{aligned} (\lambda_j^i)^{k+1} &= (\lambda_j^i)^k + \rho((e_j)^{k+1} - (e_j^i)^{k+1}), \forall j \in \delta(i), \\ (\mu_j^i)^{k+1} &= (\mu_j^i)^k + \rho((f_j)^{k+1} - (f_j^i)^{k+1}), \forall j \in \delta(i). \end{aligned}$$

In each iteration of the above decentralized algorithm, each node solves two optimization problems, shares information with its neighbors, and updates multipliers. We can explicitly write out the local communication in each step of the decentralized algorithm:

- 1) Each node i receives voltage estimates $((e_j^i)^k, (f_j^i)^k)$ and multipliers $((\lambda_j^i)^k, (\mu_j^i)^k)$ from its neighbors $j \in \delta(i)$, solves the first optimization problem, and passes the result $((P_i^G)^{k+1}, (Q_i^G)^{k+1}, (e_i)^{k+1}, (f_i)^{k+1})$ to its neighbors.
- 2) Each node i uses its own updated information $((P_i^G)^{k+1}, (Q_i^G)^{k+1}, (e_i)^{k+1}, (f_i)^{k+1})$, multipliers $((\lambda_j^i)^k, (\mu_j^i)^k)$, and neighbors updated voltages $((e_j)^{k+1}, (f_j)^{k+1})$ to solve the second optimization problem and pass solution $((e_j^i)^{k+1}, (f_j^i)^{k+1})$ to its neighbors.
- 3) Each node i updates its multipliers using its neighbors true voltages $((e_j)^{k+1}, (f_j)^{k+1})$ and its own estimate $((e_j^i)^{k+1}, (f_j^i)^{k+1})$.

IV. COMPUTATIONAL RESULTS AND DISCUSSIONS

In this section, we present preliminary testing results on a radial network and discuss the performance of the proposed algorithms for both rectangular and polar formulations. Then, we show additional results for two small meshed networks and have further discussions on algorithm's performance for meshed networks.

The proposed algorithms are prototyped in MATLAB 7.10, where the convex quadratic subproblems are solved by the MATLAB solver `quadprog`. The non-convex subproblems are solved by the MATLAB nonlinear solver `fmincon`. We acknowledge that the non-convex problems could be solved by more powerful solvers such as IPOPT [18]. However, `fmincon` seems to be sufficient for the subproblems have a simple structure and small size. All tests were performed on a 64-bit Windows 7 ThinkPad W520 with Intel i7-2720QM 2.2 GHz CPU and 8GB RAM.

The structure of the radial network is illustrated in Figure 2.

The decentralized algorithms are terminated if the objective value obtained has relative error less than 10^{-2} with

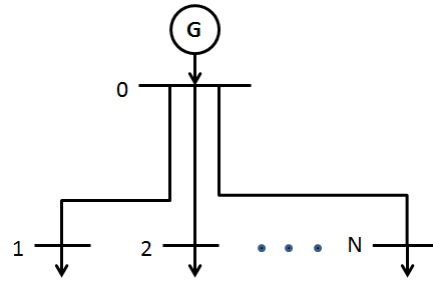


Fig. 2. The radial network consists of a generator node 0 and load nodes $1, 2, \dots, N$.

respect to the global optimum, which is pre-computed by a global OPF solver [19]. We observe that the decentralized algorithms converge to the global optimum upon termination with infeasibility residual less than 10^{-3} when the algorithms are started from a point close to the optimum. We emphasize that since the problem is non-convex and only local optimization techniques are employed, it may happen that both the subproblems and the entire problem only find local optimum. To the authors' best knowledge, there is no general convergence proof for ADMM type algorithms applied to nonconvex problems. To enhance convergence, one could argue that, since the OPF problems are usually solved repeatedly every few minutes, earlier solutions could be used as warm start for later re-solving. However, the actual performance of such a warm-start strategy needs to be evaluated numerically, which is an interesting future research direction.

TABLE I. COMPUTATIONAL RESULTS

N	Rectangular AC OPF			Polar AC OPF		
	iter	T_S (s)	T_P (s)	iter	T_S (s)	T_P (s)
5	33	7.66	1.28	54	13.45	2.24
10	44	15.26	1.39	95	42.09	3.83
20	63	37.71	1.80	157	125.91	6.00
30	85	76.89	2.48	225	252.50	8.15
40	102	121.87	2.97	290	443.95	10.82
50	128	198.40	3.89	367	807.34	15.83

Table I shows the computational results for the radial network shown in Figure 2 with the number of load nodes N up to 50. For each algorithm, an accumulated CPU time (T_S) of the serial implementation and the number of iterations (iter) are recorded. From this, we can have a rough estimation of the CPU time in a parallel implementation (T_P) where $T_P \approx T_S/(N+1)$. We can see that, for AC OPF, the algorithm using rectangular form has a faster convergence than the one using polar form. This may be explained by the fact that the rectangular AC OPF algorithm has a convex quadratic subproblem, whereas subproblems in the polar AC OPF algorithm are non-convex problems involving trigonometric functions. We also observe that the computation times (T_P) in parallel implementation of both algorithms grow linearly with respect the size of the network (see Figure 3). This shows a clear advantage of the fully decentralized algorithms: Even though the entire network can become very large and complex, as long as the local structure of the network remains relatively simple with sparse connections between neighbor nodes, the fully decentralized OPF algorithm will have a scalable computation and communication complexity.

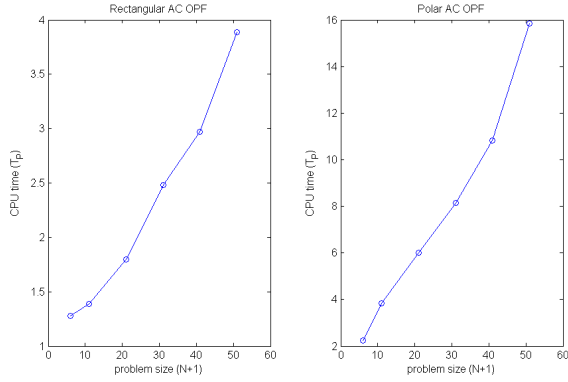


Fig. 3. Plots of computation time (T_P) vs. problem size.

The decomposition only depends on the local connectivity of the network and is independent of the global network structure. Therefore, the resulting distributed algorithms should in principle be applicable to both radial and meshed networks. We have also conducted preliminary tests on small meshed networks from MATPOWER's library [20] (a 4-bus system with 2 generators and 4 branches connecting 4 buses in a cycle, and a 6-bus system with 3 generators and 11 branches). For both meshed networks, we observed global convergence under the same termination criterion for the radial network. In particular, for the 4-bus system the distributed algorithm in rectangular form converges $T_s = 3.87$ sec and 31 iterations; For the 6-bus system, the algorithm converges in $T_s = 5.67$ sec and 32 iterations. However, we did observe that the convergence is more sensitive to the initial starting point of the algorithm, which indicates that the meshed network may be much more difficult to solve than the radial case.

V. CONCLUSIONS

We proposed fully decentralized algorithms for solving the important problem of optimal power flow, where each node in the power network solves simple optimization problems using only local information. Such algorithms can be a desirable alternative to the current centralized control framework, especially when the size and complexity of power systems increase significantly.

We conducted preliminary computational experiments on radial networks. Test results demonstrated good convergence performance of the proposed algorithms for radial networks, i.e. the global optimum is obtained and linear growth computation complexity is observed. We have also shown convergence results for 4-bus and 6-bus meshed network. For future research, we will conduct extensive experiments on larger meshed networks and will study issues related to warm start and convergence acceleration.

ACKNOWLEDGEMENTS.

This work is partially supported by the Department of Energy Award DE-OE0000190.

REFERENCES

- [1] G. Huang and W. Ongsakul, "Speedup and synchronization overhead analysis of Gauss-Seidel type algorithms on a sequent balance machine," *IEEE Proceedings, Part C*, vol. 141, no. 5, pp. 437–444, September 1994.
- [2] S. Lin and J. E. V. Ness, "Parallel solution of sparse algebraic equations," *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 743–749, May 1994.
- [3] T. Oyama, T. Kitahara, and Y. Serizawa, "Parallel processing for power system analysis using band matrix," *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 1010–1016, August 1990.
- [4] Task Force of the Computer and Analytical Methods Subcommittee of the Power Systems Engineering Committee, "Parallel processing in power systems computation," *IEEE Transactions on Power Systems*, vol. 7, no. 2, pp. 629–637, May 1992.
- [5] B. Kim and R. Baldick, "Coarse-grained distributed optimal power flow," *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 932–939, May 1997.
- [6] R. Baldick, B. Kim, C. Chase, and Y. Luo, "A fast distributed implementation of optimal power flow," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 858–864, August 1999.
- [7] P.N.Biskas and A. Bakirtzis, "Decentralized opf of large multi-area power systems," *IEE Proceedings Generation, Transmission & Distribution*, vol. 153, no. 1, pp. 99–105, January 2006.
- [8] F. Nogales, F. Prieto, and A. Conejo, "A decomposition methodology applied to the multi-area optimal power flow problem," *Annals of Operations Research*, vol. 120, no. 1-4, pp. 99–116, Apr. 2003.
- [9] A. Lam, B. Zhang, and D. Tse, "Distributed algorithms for optimal power flow problem," *arXiv:1109.5229v1*, September 2011.
- [10] R. Jabr, "Radial distribution load flow using conic programming," *IEEE Transactions on Power Systems*, vol. 21, no. 3, pp. 1458–1459, August 2006.
- [11] J. Lavaei and S. Low, "Zero duality gap in optimal power flow problem," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 92–107, February 2012.
- [12] S. Sojoudi and J. Lavaei, "Network topologies guaranteeing zero duality gap for optimal power flow problem," *Submitted to IEEE Transactions on Power Systems*, 2011.
- [13] B. Zhang and D. Tse, "Geometry of feasible injection region of power networks," *Proceeding of Allerton*, 2011.
- [14] M. Fukuda, M. Kojima, K. Murota, and K. Nakata, "Exploiting sparsity in semidefinite programming via matrix completion i: General framework," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 647–674, May 2000.
- [15] G. Cohen, "Auxiliary problem principle and decomposition of optimization problems," *Journal of Optimization Theory and Applications*, vol. 32, no. 3, pp. 277–305, November 1980.
- [16] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.
- [17] H. Glavitsch and R. Bacher, *Optimal power flow algorithms*. ACADEMIC Press Inc., 1991, vol. 41 of Analysis and Control System Techniques for Electric Power Systems.
- [18] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.
- [19] D. T. Phan, "Lagrangian duality and branch-and-bound algorithms for optimal power flow," *Operations Research*, vol. 60, no. 2, pp. 275–285, 2012.
- [20] R. Zimmerman, C. Murillo-Sánchez, and R. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, feb. 2011.