ADMM for Nonconvex AC Optimal Power Flow

Rabi Shankar Kar, Zhixin Miao, Miao Zhang, and Lingling Fan

Abstract— **The objective of the paper is to implement alternative direction method of multipliers (ADMM) to solve a nonconvex alternating current optimal power flow (AC OPF) problem. There is no guarantee of convergence for ADMM when it is applied to a nonconvex optimization problem. In this article, we not only present the procedure of consensus ADMM implementation on an AC OPF problem for IEEE 14-bus system but also present the results related to ADMM parameters and convergence. The solutions from MATPOWER and ADMM implementation are also compared.**

Index Terms— **AC OPF; ADMM**

I. Introduction

AC OPF is a nonlinear and nonconvex optimization problem. The decision variables include generators' real and reactive power outputs and voltage magnitude and angle at each bus. The objective function is usually the generation cost and the constraints include equality constraints that describe power injection relationship with voltage phasor and inequality constraints that describe generator limits, voltage limits, and line flow limits.

Alternating direction method of multipliers (ADMM) is a simple powerful technique suited to solve distributed convex optimization. In ADMM, we divide the main system into different sub-system called as area, and each area is coordinated to find the final optimized solution. It is a attempt to blend the benefits of Dual Decomposition method and Augmented Lagrangian Methods for constraints optimization in ADMM algorithm. The detail description on dual decomposition and Augmented Lagrangian is discussed in [1]. In ADMM, there are different algorithms, e.g., Gauss-Seidal, consensus ADMM, and proximal Jacobian ADMM.

ADMM has been applied to solve AC OPF problems [2]–[5]. In distributed AC OPF, the entire power grid is separated into multiple regions and each region solves its own optimization problem with the inputs from the other region. The output from each region is used for updating until convergence. For nonconvex optimization problems, there is no guaranteed convergence for ADMM. Reference [2] describes ADMM implementation for nonconvex AC OPF problems and it shows that convergence only occurs under certain conditions.

To have guaranteed convergence, AC OPF problems are first relaxed to have convex optimization problems. In [3], second-order cone program (SOCP) relaxation is applied to relax the AC OPF problem for a distribution network. Such relaxation is exact under milder assumptions for distribution tree networks. ADMM is then applied to solve the SOCP relaxation problem. Reference [5] demonstrated consensus ADMM and proximal Jacobian ADMM for DC OPF and SOCP relaxation of AC OPF.

Another type of relaxation is semidefinite programming (SDP) relaxation. In [6], [7], SDP relaxation is applied on a nonlinear state estimation problem. ADMM is then applied to solve the convex relaxation problems. The state estimation problem has a different objective function than the AC OPF problem. It does not have the inequality constraints imposed but has the equality constraints that describe each bus's power injection.

The objective of this paper is to present how to implement ADMM for nonconvex AC OPF. Though this topic has been addressed in [2], the reference paper does not present a tutorial approach of implementation. In addition, information exchange among subsystems is not straightforward.

Our contributions include a tutorial presentation to show how to separate the grid into subsystems and overlapping areas, and how to implement consensus ADMM to find the solution.

The rest of the paper is organized as follows. Section II describes the details about the ADMM algorithm. Section III describes about the implementation of consensus ADMM for an AC OPF problem using IEEE 14-bus system as an example system. In Section IV, case studies are presented. Section V concludes the paper.

II. ADMM Algorithm

A. Standard ADMM

ADMM is an algorithm to solve distributed optimization problem. Consider an optimization problem statement which has an objective function consists of the sum of $G_1(\boldsymbol{x})$ and $G_2(\boldsymbol{z})$:

$$\min_{\boldsymbol{x}\in c_1, \boldsymbol{z}\in c_2} \quad G_1(\boldsymbol{x}) + G_2(\boldsymbol{z}) \qquad (1)$$
$$\text{s.t.} \quad \boldsymbol{Ax} = \boldsymbol{z}$$

The augmented Lagrangian equation is defined as:

$$L_\rho(\boldsymbol{x},\boldsymbol{z},\boldsymbol{\lambda}) = G_1(\boldsymbol{x}) + G_2(\boldsymbol{z}) + \boldsymbol{\lambda}^T(\boldsymbol{Ax}-\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{Ax}-\boldsymbol{z}\|_2^2 \qquad (2)$$

where $\rho > 0$ is called the penalty parameter. The augmented Lagrangian method is developed in part to bring robustness to the dual ascent method, and in particular, to yield convergence without assumptions of strict convexity or finiteness of objective function G [1], [2].

To search the optimal point, it is performed via an alternating procedure which starts from initial values $\boldsymbol{z}_0$ and $\boldsymbol{\lambda}_0$, and

R. Kar, Z. Miao, M. Zhang, and L. Fan are with Dept. of Electrical Engineering, University of South Florida, Tampa FL 33620. Email: zmiao@usf.edu

iteratively updated according to eq. (3) until convergence.

$$\boldsymbol{x}^{k+1} = \underset{\boldsymbol{x}\in c_1}{\text{argmin}} \quad \Big(G_1(\boldsymbol{x}) + G_2(\boldsymbol{z}^k) + (\boldsymbol{\lambda}^k)^T(\boldsymbol{Ax} - \boldsymbol{z}^k) + \frac{\rho}{2}\|\boldsymbol{Ax} - \boldsymbol{z}^k\|_2^2\Big) \tag{3a}$$

$$\boldsymbol{z}^{k+1} = \underset{\boldsymbol{z}\in c_2}{\text{argmin}} \quad \Big(G_1(\boldsymbol{x}^{k+1}) + G_2(\boldsymbol{z}) + (\boldsymbol{\lambda}^k)^T(\boldsymbol{Ax}^{k+1} - \boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{Ax}^{k+1} - \boldsymbol{z}\|_2^2\Big) \tag{3b}$$

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \rho(\boldsymbol{Ax}^{k+1} - \boldsymbol{z}^{k+1}) \tag{3c}$$

Finally, the ADMM iteration should converge to the following results.

$$\lim_{k\to\infty} \begin{bmatrix} \boldsymbol{Ax}^k - \boldsymbol{z}^k \\ G_1(\boldsymbol{x}^k) + G_2(\boldsymbol{z}^k) \\ \boldsymbol{\lambda}^k \end{bmatrix} = \begin{bmatrix} 0 \\ v^\star \\ \boldsymbol{\lambda}^\star \end{bmatrix} \tag{4}$$

where $v^\star$ is the optimal value of the objective function, and $\boldsymbol{\lambda}^\star$ is the optimal dual variable.

B. Consensus ADMM for OPF

Consensus ADMM is a special form of ADMM. Consider an optimization problem given below:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} f_i(\boldsymbol{x}_i) \\ \text{s.t.} \quad & \boldsymbol{x_i} = \boldsymbol{z}, \quad i = 1, \cdots, m \end{aligned} \tag{5}$$

where $_i$ notates variables in Area i. It is called global consensus algorithm, since at convergence all the local variables should be equal to global variable. The augmented Lagrangian function can be derived for the problem (5) as below,

$$L_\rho(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m, \boldsymbol{z}, \boldsymbol{\lambda}) = \sum_{i=1}^{m} \underbrace{\Big(f_i(\boldsymbol{x}_i) + \boldsymbol{\lambda}_i^T(\boldsymbol{x}_i - \boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{x}_i - \boldsymbol{z}\|_2^2\Big)}_{L_{\rho,i}} \tag{6}$$

Thus, the common global variable $\boldsymbol{z}$ is solved by collaborative filtering in (7).

$$\boldsymbol{x}_i^{k+1} = \underset{\boldsymbol{x}_i}{\text{argmin}} \quad L_{\rho,i}(\boldsymbol{x}_i, \boldsymbol{\lambda}_i^k, \boldsymbol{z}^k), \quad i = 1, \cdots, m \tag{7a}$$

$$\boldsymbol{z}^{k+1} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i^{k+1} \tag{7b}$$

$$\boldsymbol{\lambda}_i^{k+1} = \boldsymbol{\lambda}_i^k + \rho(\boldsymbol{x}_i^{k+1} - \boldsymbol{z}^{k+1}), \qquad i = 1, \cdots, m \tag{7c}$$

For details refer to [1], [5].

In this paper, we will adopt consensus ADMM to solve a nonconvex AC OPF problem. Each area only need to handle its own objective function and constraints for a given global $\boldsymbol{z}^k$. All the areas update their decision variables in parallel till convergence.

III. ADMM Formulation for AC OPF

In this section, we will present the ADMM formulation for an AC OPF problem related to the IEEE 14-bus system. The distribution topology of IEEE 14-bus is shown in Fig. 1. Each bus has their own load. There are three transformers between bus 5 and 6, bus 4 and 9, bus 4 and 7, respectively. Five generators are connected on bus 1, 2, 3, 6 and 8, respectively.

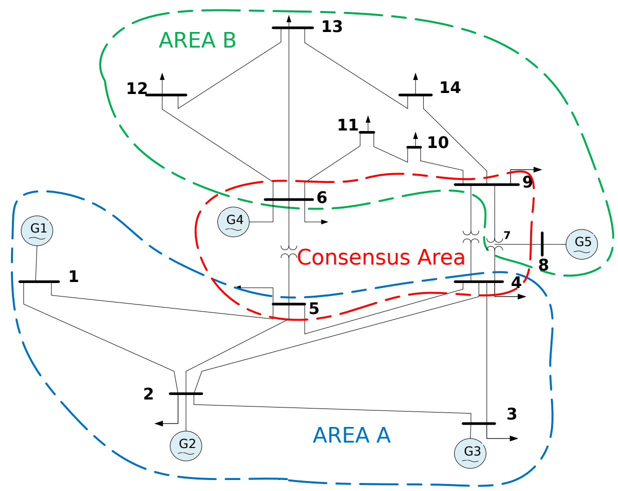


Fig. 1: IEEE 14 bus partition into 2 areas. The dash lines denotes the boundary of each areas and the buses enclosed in each area.

A. AC OPF Statement

Each bus number is marked as i, where $i = 1, 2, \ldots 14$. Let $\boldsymbol{Y}$ be the branch admittance matrix, $\overline{V_i}$ and $\overline{I_i}$ represent, respectively, voltage and current injection at bus i. The net power injection at the bus i can be expressed as follows:

$$S_i = \overline{V_i} \cdot \overline{I_i}^* \tag{8a}$$

$$P_i(V,\theta) = \sum_{j=1}^{N} V_i V_j (G_{ij}\cos(\theta_i - \theta_j) + B_{ij}\sin(\theta_i - \theta_j)) \tag{8b}$$

$$Q_i(V,\theta) = \sum_{j=1}^{N} V_i V_j (G_{ij}\sin(\theta_i - \theta_j) - B_{ij}\cos(\theta_i - \theta_j)) \tag{8c}$$

where $\overline{I_i}$ is derived by $Y_{i,j}$ and $\overline{V_j}$ in (9), $\boldsymbol{G}$ is the bus conductance matrix, $G_{ij} = \text{Re}(Y_{ij})$ and, $\boldsymbol{B}$ is the bus susceptance matrix, $B_{ij} = \text{Im}(Y_{ij})$ and $\boldsymbol{Y}$ is the bus admittance matrix.

$$\overline{I_i} = \sum_j Y_{i,j} \cdot \overline{V_j} \tag{9}$$

The AC OPF problem can be formulated as a minimization of the summation of individual polynomial cost functions $f_{P_{gi}}$ and $f_{Q_{gi}}$ of real and reactive power generation, respectively, for each generator on the following buses $(1,2,3,6,8)$, shown in (10).

The optimization variables $\boldsymbol{x}$ in (11) consists of 5×1 vectors of generator real and reactive power generations P_g

and Q_g, and 14×1 vectors of voltage magnitudes V and angles θ.

$$\min_{\boldsymbol{x}} \sum f_P(P_{g_i}) + f_Q(Q_{gi}), \tag{10}$$

$$\boldsymbol{x} = \begin{bmatrix} P_g \\ Q_g \\ V \\ \theta \end{bmatrix} \tag{11}$$

$$\text{s.t.} \quad \underline{P_{gi}} \le P_{gi} \le \overline{P_{gi}} \tag{12a}$$
$$\underline{Q_{gi}} \le Q_{gi} \le \overline{Q_{gi}} \tag{12b}$$
$$\underline{V_i} \le V_i \le \overline{V_i} \tag{12c}$$
$$\underline{\theta_i} \le \theta_i \le \overline{\theta_i} \tag{12d}$$
$$P_{gi} - P_{di} - P_i(V,\theta) = 0 \tag{12e}$$
$$Q_{gi} - Q_{di} - Q_i(V,\theta) = 0 \tag{12f}$$

The inequality constrains eqs. (12a) to (12d) give the upper and lower bounds for generator real power, reactive power, all bus's voltage magnitudes and angles.

The equality constrains eqs. (12e) and (12f) are from power injection equations (8b) and (8c), which including real and reactive components, expressed as functions of V, θ and generator generations P_g and Q_g. The load demand P_d and Q_d at each bus is assumed constant and are given in the MATPOWER [8] input file. The AC OPF is a nonconvex optimization problem due to the nonlinear equality constraints related to power injections.

B. ADMM Implementation

The IEEE 14-bus network is shown in Fig. 1. It has 5 generator buses and 9 load buses. The network is partitioned into 2 areas: Area A and Area B. In the figure, the area enclosed with red dotted line is the intersection area called as consensus area or overlapping area. Area A includes buses 1-5, the branches inside this area as well as the branches in the consensus area (4-7, 4-9, and 5-6). Area B includes buses 6-14, branches inside Area B as well as the branches 6-5, 7-4, and 9-4. Each area decides its own buses' voltage magnitudes, phase angles as well as real and reactive power of its generators inside the area. Area A treats the buses belonging to Area B in the overlapping area as voltage sources and decides their voltage magnitudes and phase angles. Power injection equality of those buses (bus 6, 7, 9) will not be considered. Area B treats the buses inside the overlapping area but belonging to Area A as the boundary buses (bus 4 and bus 5). For its own buses, power injection equations will be imposed as equality constraints while considering bus 4 and 5 as two voltage sources.

Both areas will decide the voltage magnitudes and phase angles for the buses in the overlapping area. Thus, voltage phasors of those buses (4, 5, 6, 7, 9) will be treated as the local variables to achieve consensus. Below is a detailed list of each area's information and consensus area's information.

- Area A:
  - Its own buses: 1-5.
  - Boundary buses: 6, 7 and 9.
  - Branches: 1-2, 1-5, 2-3, 2-4, 2-5, 3-4, 4-5, 4-7. 4-9, 5-6.
- Area B:
  - Its own buses: 6-14.
  - Boundary buses: 4 and 5.
  - Branches: 6-11, 6-12, 6-13, 7-8, 7-9, 9-10, 9-14, 10-11, 12-13, 13-14, 5-6, 4-7, 4-9.
- Consensus Area:
  - Consensus buses: 4, 5, 6, 7, 9.

Based on the partition of the area, the decision variables for each area will be defined. We define the variables which belong to area A as $(.)_A$ and the variables which belong to area B as $(.)_B$. And we define the bus variables as $(.)_i^k$ where i is the bus number and k denotes the number of iterations.

Let $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$ be the decision variables for area A and B respectively, where,

$$\boldsymbol{x}_A = \begin{bmatrix} P_{gA} & Q_{gA} & V_{1_A} & \cdots & V_{7_A} & V_{9_A} & \theta_{1_A} & \cdots & \theta_{7_A} & \theta_{9_A} \end{bmatrix}^T \tag{13a}$$
$$\boldsymbol{x}_B = \begin{bmatrix} P_{gB} & Q_{gB} & V_{4_B} & \cdots & V_{14_B} & \theta_{4_B} & \cdots & \theta_{14_B} \end{bmatrix}^T \tag{13b}$$

where $P_{gA} = [P_{g1}, P_{g2}, P_{g3}]$, $Q_{gA} = [Q_{g1}, Q_{g2}, Q_{g3}]$, P_{gB} includes $[P_{g6}, P_{g8}]$, and Q_{gB} includes $[Q_{g6}, Q_{g8}]$.

$\boldsymbol{z}$ denotes the vector of the consensus variables:

$$\boldsymbol{z} = \begin{bmatrix} V_4 & V_5 & V_6 & V_7 & V_9 & \theta_4 & \theta_5 & \theta_6 & \theta_7 & \theta_9 \end{bmatrix}^T. \tag{14}$$

The global variable $\boldsymbol{z}$ is initialized with the default value of V and θ. The parameter ρ is considered as 20 initially. The lower bounds and upper bounds of the objective variables, as in (12a)-(12d), are defined based on MATPOWER's input file. The dual variable vectors are defined as $\boldsymbol{\lambda}_A$ and $\boldsymbol{\lambda}_B$ for area A and area B respectively.

The global variable vector $\boldsymbol{z}$ is related with consensus local variable $\boldsymbol{x}_{cA}$ and $\boldsymbol{x}_{cB}$: $\boldsymbol{x}_{cA} = \boldsymbol{z}$, $\boldsymbol{x}_{cB} = \boldsymbol{z}$.

1) Area A: The objective function for Area A is as follows

$$L_{\rho A}(\boldsymbol{x}_A, \boldsymbol{z}^k, \boldsymbol{\lambda}_A^k) = \sum_{i=1,2,3} f_{pi}(P_{gi}) + (\boldsymbol{\lambda}_A^k)^T(\boldsymbol{x}_{cA} - \boldsymbol{z}^k) + \frac{\rho}{2}\left\|\boldsymbol{x}_{cA} - \boldsymbol{z}^k\right\|_2^2 \tag{15}$$

Inequality constraints include the following.

$$\text{s.t.} \quad \underline{P_{gi}} \le P_{gi} \le \overline{P_{gi}}, \quad i = 1, \ldots, 3 \tag{16a}$$
$$\underline{Q_{gi}} \le Q_{gi} \le \overline{Q_{gi}}, \quad i = 1, \ldots, 3 \tag{16b}$$
$$\underline{V_i} \le V_i \le \overline{V_i}, \quad i = 1, \ldots, 7, 9 \tag{16c}$$
$$\underline{\theta_i} \le \theta_i \le \overline{\theta_i}, \quad i = 1, \ldots, 7, 9 \tag{16d}$$

Equality constraints including the following.

$$P_{g_i} - P_{D_i} = \sum V_i V_j (G_{ij} \cos(\theta_i - \theta_j) + B_{ij} \sin(\theta_i - \theta_j)) \tag{17a}$$
$$Q_{g_i} - Q_{D_i} = \sum V_i V_j (G_{ij} \sin(\theta_i - \theta_j) - B_{ij} \cos(\theta_i - \theta_j)) \tag{17b}$$
$$i = 1, \ldots, 5$$

2) Area B: The objective function for Area B is as follows

$$L_{\rho B}(\boldsymbol{x}_B, \boldsymbol{z}^k, \boldsymbol{\lambda}_B^k) = \sum_{i=6,8} f_{pi}(P_{gi}) + (\boldsymbol{\lambda}_B^k)^T(\boldsymbol{x}_{cB} - \boldsymbol{z}^k) + \frac{\rho}{2}\left\|\boldsymbol{x}_{cB} - \boldsymbol{z}^k\right\|_2^2 \tag{18}$$

Inequality constraints include the following.

$$\text{s.t.} \quad \underline{P_{gi}} \leq P_{gi} \leq \overline{P_{gi}}, \quad i = 6, 8 \tag{19a}$$

$$\underline{Q_{gi}} \leq Q_{gi} \leq \overline{Q_{gi}}, \quad i = 6, 8 \tag{19b}$$

$$\underline{V_i} \leq V_i \leq \overline{V_i}, \quad i = 4, \ldots, 14 \tag{19c}$$

$$\underline{\theta_i} \leq \theta_i \leq \overline{\theta_i}, \quad i = 4, \ldots, 14 \tag{19d}$$

Equality constraints including the following.

$$P_{g_i} - P_{D_i} = \sum V_i V_j (G_{ij}\cos(\theta_i - \theta_j) + B_{ij}\sin(\theta_i - \theta_j)) \tag{20a}$$

$$Q_{g_i} - Q_{D_i} = \sum V_i V_j (G_{ij}\sin(\theta_i - \theta_j) - B_{ij}\cos(\theta_i - \theta_j)) \tag{20b}$$

$$i = 6, \ldots, 14$$

At each iteration step k, for a given $\boldsymbol{z}^k$, $\boldsymbol{\lambda}_A^k$ and $\boldsymbol{\lambda}_B^k$, Area A finds the optimal solution $\boldsymbol{x}_A^{k+1}$ and Area B finds the optimal solution $\boldsymbol{x}_B^{k+1}$. The update process for the decision variables based in the consensus ADMM are as follows:

$$\boldsymbol{z}^{k+1} = \frac{1}{2}(\boldsymbol{x}_{cA}^{k+1} + \boldsymbol{x}_{cB}^{k+1}) \tag{21a}$$

$$\boldsymbol{\lambda}_A^{k+1} = \boldsymbol{\lambda}_A^k + \rho(\boldsymbol{x}_{cA}^{k+1} - \boldsymbol{z}^{k+1}) \tag{21b}$$

$$\boldsymbol{\lambda}_B^{k+1} = \boldsymbol{\lambda}_B^k + \rho(\boldsymbol{x}_{cB}^{k+1} - \boldsymbol{z}^{k+1}) \tag{21c}$$

IV. Case Study

The pseudocode is presented in Algorithm 1. Matlab function $fmincon$ is used to solve each area's optimization problem.

Algorithm 1 Pseudocode for ADMM

procedure Main Program
- Initialize ρ.
- Initialize x_{A0} and x_{B0} for area A and area B.
- Initialize the lower bound and upper bound for the decision variables.
- Initialize the threshold error value
- **while** $err \leq$ threshold error **do**
  - ***Area A***:
  - $Obj_A \leftarrow$ Objective function of area A
  - $\boldsymbol{x}_{A_{out}} \leftarrow$ Output from $fmincon$ ▷ solves Obj_A
  - ***Area B***:
  - $Obj_B \leftarrow$ Objective function of area B
  - $\boldsymbol{x}_{B_{out}} \leftarrow$ Output from $fmincon$ ▷ solves Obj_B
  - Update $\boldsymbol{x}_A^{k+1} \leftarrow \boldsymbol{x}_{A_{out}}$
  - Update $\boldsymbol{x}_B^{k+1} \leftarrow \boldsymbol{x}_{B_{out}}$
  - Update $\boldsymbol{z}^{k+1} \leftarrow Avg(\boldsymbol{x}_A^{k+1}, \boldsymbol{x}_B^{k+1})$
  - Update $\boldsymbol{\lambda}_A^{k+1} \leftarrow \boldsymbol{\lambda}_A^k + \rho * (\boldsymbol{x}_A^{k+1} - \boldsymbol{z}^{k+1})$
  - Update $\boldsymbol{\lambda}_B^{k+1} \leftarrow \boldsymbol{\lambda}_B^k + \rho * (\boldsymbol{x}_B^{k+1} - \boldsymbol{z}^{k+1})$
- **end while**

end procedure

To illustrate convergence of the consensus ADMM, the norm of the error $\|x_i^k - z^k\|$ was calculated and the dual variables λ_A and λ_B are plotted versus the iteration steps. Along with dual variable, the decision variables P_g, Q_g, V and θ are also plotted.

We did test for three different scenarios to check the robustness of the consensus ADMM optimization algorithm.

- Scenario 1: We assume that the costs of generation are all zero.
- Scenario 2: The cost of each generator is assumed to be linear cost.
- Scenario 3: The cost of each generator is quadratic using the same cost function as the MATPOWER input file *case*14.*m*.

A. Scenario 1: Zero generation cost

Fig. 2 resents the outputs for Scenario 1. The figure shows convergence for all variables. This case is tested for ρ = 20.

B. Scenario 2: Linear cost

The cost function equation of a power system network is stated as (22).

$$f(P_{g_i}) = \sum C_{1_i} P_{g_i} + C_{2_i} P_{g_i}^2 \tag{22}$$

where C_{1_i} is the linear cost coefficient and C_{2_i} is the quadratic cost Coefficient. In Scenario 2, C_{2_i} is changed to zero. Therefore,

$$f(P_{g_i}) = \sum C_{1_i} P_{g_i} \tag{23}$$

Fig. 3 presents the outputs with a linear cost function. In this case, the penalty parameter ρ is 20 and the system is converging for this ρ. We calculated the minimum generation cost at $\rho = 20$, and it was found to be 5383 \$/hr.

C. Scenario 3: Quadratic cost

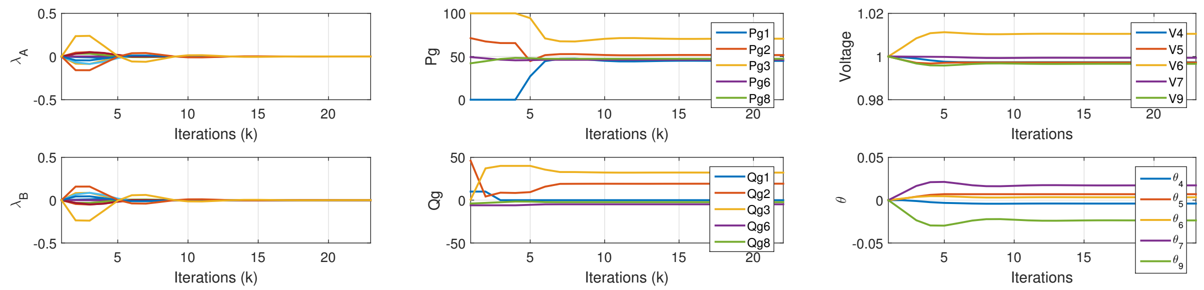
A quadratic cost function for a power system network is shown as in (22).

In this case, tests were conducted with different ρs. The outputs in Fig. 4 shows that when ρ is small, ADMM is not converging.

TABLE I: MATPOWER and ADMM Comparison

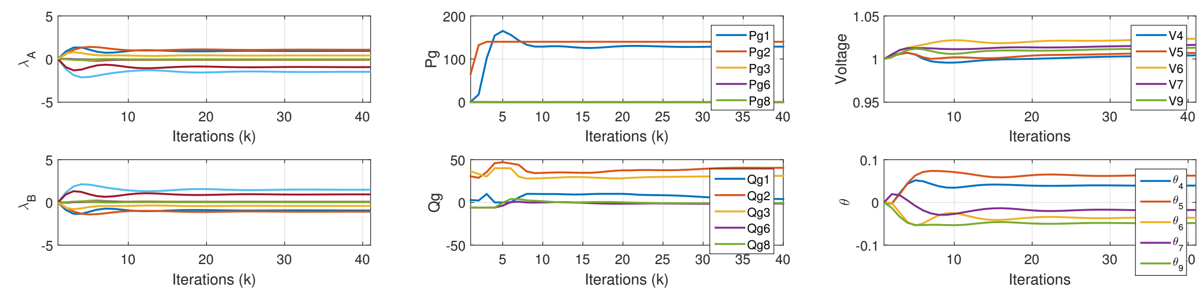
Generations	MATPOWER	ADMM	Error
Pg_1(MW)	194.33	194.30	-0.03
Qg_1(MVar)	0	0	0
Pg_2(MW)	36.72	36.71	-0.01
Qg_2(MVar)	23.69	23.52	-0.17
Pg_3(MW)	28.74	28.74	0
Qg_4(MVar)	24.13	24.27	0.14
Pg_6(MW)	0	0	0
Qg_6(MVar)	11.55	11.49	-0.06
Pg_8(MW)	8.49	8.47	-0.02
Qg_8(MVar)	8.27	8.23	-0.04
Minimum Cost (\$/hr)	8081.5	8079.3	-2.2

Next ρ is increased to $100,000$ and ADMM is converging. The outputs are plotted in Fig. 5. The outputs show that all variables are converging. The converged values are compared with MATPOWER's solution as shown in Table I. The results from ADMM are similar as those from MATPOWER.



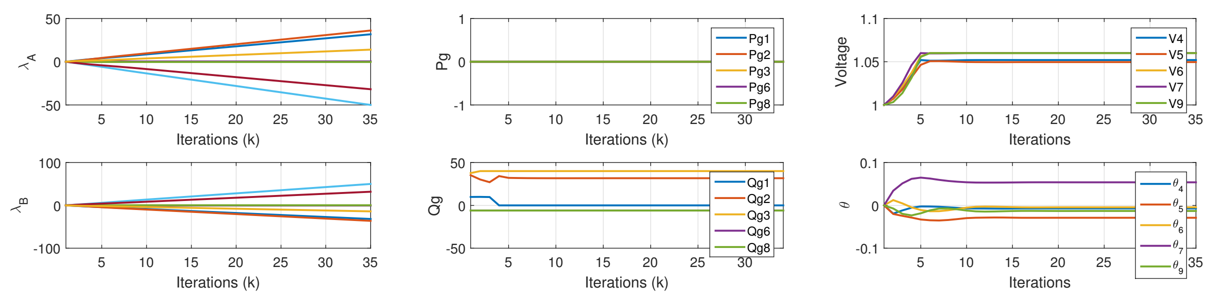
(a) λ_A and λ_B. (b) P_g and Q_g. (c) The voltages in consensus area.

Fig. 2: Scenario 1: Zero generation cost. $\rho = 20$.



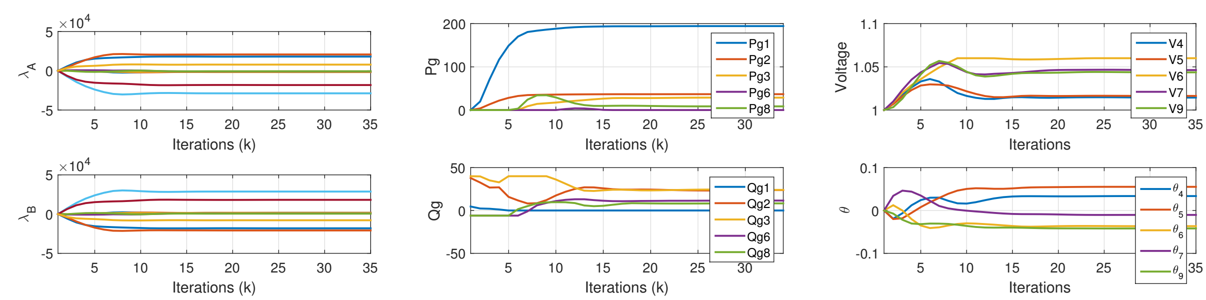
(a) λ_A and λ_B. (b) P_g and Q_g. (c) The voltages in consensus area.

Fig. 3: Scenario 2: Linear generation cost. $\rho = 20$.



(a) λ_A and λ_B. (b) P_g and Q_g. (c) The voltages in consensus area.

Fig. 4: Scenario 3: Quadratic cost. $\rho = 20$



(a) λ_A and λ_B (b) P_g and Q_g (c) The voltages in consensus area.

Fig. 5: Scenario 3: Quadratic cost. $\rho = 100,000$.

Remarks: We have demonstrated the performance of ADMM for nonconvex AC OPF. Depending on the penalty parameter ρ, the algorithm may or may not converge. Carefully choosing ρ results in a solution same as that from MATPOWER.

V. Conclusion

ADMM is implemented in this paper for a nonconvex AC OPF. The consensus ADMM algorithm is applied to IEEE 14-bus system to solve an AC OPF problem. Different scenarios are carried out to test our methodology. Based on the output of the case studies, we can conclude that a nonconvex optimization problem will not always present a stable converged output. Ability to achieve a converged solution depends on the tuning penalty parameter ρ.

References

[1] 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, 2011.

[2] T. Erseghe, "Distributed optimal power flow using admm," *IEEE transactions on power systems*, vol. 29, no. 5, pp. 2370–2380, 2014.

[3] Q. Peng and S. H. Low, "Distributed algorithm for optimal power flow on a radial network," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 167–172.

[4] A. X. Sun, D. T. Phan, and S. Ghosh, "Fully decentralized ac optimal power flow algorithms," in *Power and Energy Society General Meeting (PES), 2013 IEEE*. IEEE, 2013, pp. 1–5.

[5] M. Ma, L. Fan, and Z. Miao, "Consensus admm and proximal admm for economic dispatch and ac opf with socp relaxation," in *North American Power Symposium (NAPS), 2016*. IEEE, 2016, pp. 1–6.

[6] H. Zhu and G. B. Giannakis, "Multi-area state estimation using distributed sdp for nonlinear power systems," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*. IEEE, 2012, pp. 623–628.

[7] ——, "Power system nonlinear state estimation using distributed semidefinite programming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 6, pp. 1039–1050, 2014.

[8] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. 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, 2011.