Distributed Methods for Solving the Security-Constrained Optimal Power Flow Problem

Dzung Phan and Jayant Kalagnanam
Department of Business Analytics and Mathematical Sciences, IBM T.J. Watson Research Center
P.O. Box 218, 1101 Kitchawan Rd., Yorktown Heights, New York 10598, USA

***Abstract*—The optimal power flow is the problem of determining the most efficient, low-cost and reliable operation of a power system by dispatching the available electricity generation resources to the load on the system. Unlike the classical optimal power flow problem, the security-constrained optimal power flow (SCOPF) problem takes into account both the pre-contingency (base-case) constraints and post-contingency constraints. In the literature, the problem is formulated as a large-scale non-convex nonlinear programming. We propose two decomposition algorithms based on the Benders cut and the alternating direction method of multipliers for solving this problem. Our algorithms often generate a solution, whose objective function value is smaller than conventional approaches.**

I. Introduction

As the power grids evolve to smart grids in the future, it will become increasingly challenging for independent system operators (ISOs) and/or regional transmission operators (RTOs) to optimally manage such a complex system of systems. The tasks facing ISOs/RTOs are multitudinous, ranging from deciding which generation units (including renewable sources) to be on and off at what specific time, determining the proper pricing signals to achieve desired demand response, configuring the network topologies to ensure power flow without violating power lines physical constraints and voltage collapse, to when to replace an aging asset (such as transformer or breaker) and how to dispatch repair crew for network maintenance. All these tasks need to be optimized to fully realize the potential benefits offered by smart grids.

At the heart of the future smart grid lie two related challenging optimization problems: unit commitment and optimal power flow. Both problems are most relevant to ISOs/RTOs daily operation as they need to be solved on a daily basis, and both are computationally intensive tasks that require significant performance improvement to meet real-time operational requirements. Although these two problems are intermingled with each other, most of the current theoretical and practical efforts treat them separately because of the computational difficulty of solving a single unified problem. The unit commitment problem takes place in a day-ahead market and decides which bulk generation sources (typically thermal, nuclear and hydro sources) are awarded contracts to supply energy in the next day. This base generation capability is augmented by additional smaller capacity peaker thermal generators and external sources of energy (spot markets) connected to a subset of the grids nodes, to hedge against un-planned excess demand. The second planning stage, the optimal power flow problem, i.e., economic dispatch, which is the problem of this paper's interest, is at a smaller time-scale, typically five to fifteen minutes. It decides how the active generators are dispatched (e.g., set the level at which bulk generators produce energy) and how the produced energy is routed through the grid to consumption (or load) nodes. The primary purpose of the problem is to minimize the total cost of generation while ensuring the electrical networks balance [1]. Contingency analysis is performed in current economic dispatch practices, making sure that the load at each node of the network can be satisfied in the case of a failure of one of the generators, transmission lines, or other devices, which is called $N-1$ criterion. This problem is often regarded as a security-constrained optimal power flow problem [2].

Linear direct current (DC) approximation of the (nonlinear, non-convex) AC power flow equation is mostly used in the existing power systems [3], [4] [1, 2]. The main drawback of the DC formulation is that it does not capture the physical power flow more realistically than its AC counterpart, it is desirable to incorporate AC-based formulation (that is, nonlinear models) into existing power system optimization problems. In this work, we focus on the AC formulations. The SCOPF has been widely modeled into two groups: preventive [5] and corrective [6].

The preventive model minimizes some generation cost function by acting only on the base-case (such as, contingency-free) control variables subject to both the normal and abnormal operating constraints. For k contingency scenarios, the problem size of the preventive SCOPF is roughly $k+1$ times larger than the classical (base-case) economic dispatch problem. Solving this problem directly for large-scale power systems with numerous contingencies would lead to prohibitive memory requirements and execution times. In real-world applications, however, many post-contingency constraints are redundant, that is, their absence does not affect the optimal value [7]. Consequently, a class of algorithms based on contingency filtering techniques has been developed [5], [6], [7], [8], [9], [10] that identify and only add those potentially binding contingencies into the formulation. For example, the contingency ranking schemes from [9] are achieved by investigating a re-

This material is based upon work supported by the U.S. Department of Energy under Award Number DE-OE0000190.
Authors contact information: {phandu, jayant}@us.ibm.com

978-1-4577-2159-5/12/$31.00 ©2011 IEEE

laxed preventive SCOPF problem, where a single contingency along with the base-case is considered one at a time. The ranking methods rely on the information of Lagrangian multipliers or the decrease factor of penalized objective function values, and then select contingencies with a severity index above some threshold for further consideration. Other contingency filtering methodologies [7] aim to efficiently identify a minimal subset of contingencies to be added based upon the comparison of post-contingency violations. In addition, an approach using the generalized Benders decomposition to construct the feasibility cut from the Lagrangian multiplier vector of constraints is introduced in [11]. It showed a significant speedup in terms of computation time.

The corrective SCOPF makes use of the assumption that post-contingency constraint violations can be endured up to several minutes without damaging the equipment [6]. The corrective SCOPF allows post-contingency control variables to be rescheduled, so that it is easier to eliminate violations of contingency constraints than the preventive SCOPF. The optimal value of corrective SCOPF is often smaller than that of preventive SCOPF, but its solution is often harder to obtain, since it introduces additional decision variables and nonlinear constraints. Monticelli et al. [6] tackled the optimization problem by rewriting it in terms of only the contingency-free state variables and control variables, while constraint reductions are represented as implicit functions of these contingency-free state and control variables, which in turn are related to the infeasibility post-contingency operating subproblems. The solution algorithm then becomes an application of the generalized Benders decomposition [12] that iteratively solves a base-case economic dispatch and separate contingency analysis. Moreover, an extension of a contingency filtering technique from [7] was studied in [13], which features an additional optimal power flow module to verify the controllability of post-contingency states.

In this paper, we are concerned with the corrective SCOPF, which is formulated as follows [6]:

$$\begin{aligned} \min\ & f(u_0) \\ \text{s.t.}\ & h_c(x_c, u_c) = 0 \\ & g_c(x_c, u_c) \le 0 \\ & |u_c^i - u_0^i| \le \Delta u_i^{max}, i \in \mathcal{G}, c = 0, 1, \ldots, C, \end{aligned} \tag{1}$$

where $c = 0$ represents the normal case (i.e., no occurrence of contingency constraint), index $c = 1, ..., C$ represents a contingency, x_c is the vector of state variables (e.g., complex voltages) for the c-th configuration, u_c is the vector of control variables (e.g., the active and reactive powers), the generation cost f defined by

$$f(u_0) = \sum_{i \in \mathcal{G}} \left(c_{0i} (u_0^i)^2 + c_{1i} u_0^i + c_{2i} \right),$$

with cost parameters $c_{0i}, c_{1i}, c_{2i} \ge 0$, ($\mathcal{G}$ - the set of generators), C is the number of contingencies. Δu_i^{max} is the predetermined maximal allowed variation of control variables, h_c and g_c are operational constraints including the AC power flow balance equations. Note that the model allows post-contingency control variables rescheduling so as to eliminate contingency constraint violations. Typically, (1) is a very large-scale nonconvex nonlinear optimization problem, which requires to be solved in real-time.

The computational challenge of SCOPF is due to the size of problem caused by a large number of contingencies. In this paper, we handle this issue by decomposition techniques, which treat each contingency separately. We introduce two different algorithms to solve the optimization problem (1). The first one is based on the Benders decomposition, which exploits nicely the special structure of the problem. An adaptive cut is proposed to improve the quality of solution. The second algorithm is an application of the alternating direction method of multipliers. These distributed algorithms are able to run on parallel computers.

II. Benders Decomposition Algorithm

We will show how to use the Benders decomposition to efficiently solve the problem. The main idea of the method is that it decomposes the SCOPF into a master problem and subproblems, where subproblems check the solution feasibility for the master problem. We introduce feasibility cuts that are different from the standard technique in the literature.

A. Cut generation

Let Ω be the feasible set of the SCOPF problem and assume Ω is nonempty. An immediate result can be deduced is that Ω is a compact, nonconvex set. It is easy to see that the problem (1) can be expressed as the general nonlinear nonconvex optimization problem

$$\begin{aligned} \min\ & F(x) \\ \text{s.t.}\ & x \in X \\ & G(y) \le 0 \\ & H(y) = 0 \\ & Ax + By + b \le 0 \\ & y^L \le y \le y^U, \end{aligned} \tag{2}$$

where $X \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, A and B are matrices. In our applications, F is convex, whereas G, H and X are nonconvex. The initial master problem is of the form

$$\begin{aligned} \min\ & F(x) \\ \text{s.t.}\ & x \in X. \end{aligned} \tag{3}$$

Let $\bar{x}$ be a solution of the master problem (3). Then we need to check if the following subproblem is feasible

$$\begin{aligned} & G(y) \le 0 \\ & H(y) = 0 \\ & A\bar{x} + By + b \le 0 \\ & y^L \le y \le y^U. \end{aligned} \tag{4}$$

If (4) is feasible for the given $\bar{x}$, we terminate the algorithm. Otherwise, we will generate a linear cut, which is added to the master problem (3) to enforce the feasibility of the problem (4). Note that each contingency constraint associated with each c in the problem (1) may be written in the form of problem (4). Therefore, it suffices to work with the general form (4).

We now show how to construct a feasibility cut if (4) is infeasible. For the given $\bar{x}$, we consider the feasibility subproblem

$$\begin{array}{l} \min_{y,\alpha} \ \sum_i \alpha_i \\ \text{subject to:} \\ G(y) \le 0 \\ H(y) = 0 \\ A\bar{x} + By + b - \alpha \le 0 \\ y^L \le y \le y^U \\ \alpha \ge 0, \end{array} \tag{5}$$

by adding slack variables α_i's.

Assume that the optimal function value of (5) is strictly positive, i.e. (4) is infeasible, and $(\bar{y}, \bar{\alpha})$ is an optimal solution. Let us substitute $z := y - y^L$ and use the Taylor expansions of $G(y)$ and $H(y)$ at $\bar{y} - y^L$, we obtain the relaxed linear programming

$$\begin{array}{l} \min_{\alpha,z} \ \sum_i \alpha_i \\ \text{subject to:} \\ G(\bar{y}) + \nabla_y G(\bar{y})(z - (\bar{y} - y^L)) \le 0 \\ H(\bar{y}) + \nabla_y H(\bar{y})(z - (\bar{y} - y^L)) = 0 \\ Bz - \alpha + A\bar{x} + b + By^L \le 0 \\ z \le y^U - y^L \\ \alpha, z \ge 0. \end{array} \tag{6}$$

Since the optimal function value of (5) is strictly positive, it follows that the optimal function value of (6) associated with any optimal solution $\hat{\alpha}$ is also strictly positive. We will generate a linear cut based on the Lagrange multipliers arising from the linear programming (6).

The Lagrange dual problem of (6) is as follows:

$$\begin{array}{l} \max\limits_{\pi \ge 0, \mu \ge 0, \eta} \ \min\limits_{\substack{0 \le \alpha \\ 0 \le z \le y^U - y^L}} \\ \sum\limits_i \alpha_i + \pi^\mathsf{T}(Bz - \alpha + A\bar{x} + By^L + b) + \\ \eta^\mathsf{T}\left(H(\bar{y}) + \nabla_y H(\bar{y})(z - (\bar{y} - y^L))\right) + \\ \mu^\mathsf{T}\left(G(\bar{y}) + \nabla_y G(\bar{y})(z - (\bar{y} - y^L))\right) \\ = \max\limits_{\pi \ge 0, \mu \ge 0, \eta} \ \min\limits_{\substack{0 \le \alpha \\ 0 \le z \le y^U - y^L}} \\ \sum\limits_i \alpha_i(1 - \pi_i) + \left(B^\mathsf{T}\pi + \nabla_y^\mathsf{T} H(\bar{y})\eta + \nabla_y^\mathsf{T} G(\bar{y})\mu\right)^\mathsf{T} z + \\ \pi^\mathsf{T}(A\bar{x} + By^L + b) - \eta^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) + \\ \mu^\mathsf{T}(G(\bar{y}) - \nabla_y G(\bar{y})(\bar{y} - y^L)) \\ = \max\limits_{0 \le \pi_i \le 1, \mu \ge 0} \\ \pi^\mathsf{T}(A\bar{x} + By^L + b) - \eta^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) + \\ \mu^\mathsf{T}(G(\bar{y}) - \nabla_y G(\bar{y})(\bar{y} - y^L)) + \sum\limits_i \ell_i(\pi), \end{array}$$

where

$$\ell_i(\pi) = \begin{cases} 0 \text{ if } \left[B^\mathsf{T}\pi + \nabla_y^\mathsf{T} H(\bar{y})\eta + \nabla_y^\mathsf{T} G(\bar{y})\mu\right]_i \ge 0 \\ \left[B^\mathsf{T}\pi + \nabla_y^\mathsf{T} H(\bar{y})\eta + \nabla_y^\mathsf{T} G(\bar{y})\mu\right]_i [y^U - y^L]_i \text{ o.w.} \end{cases}$$

Suppose that $(\hat{\pi}, \hat{\eta}, \hat{\mu})$ is the optimal solution to the dual problem. Since the strong duality holds for (6), we have

$$\begin{array}{l} \hat{\pi}^\mathsf{T}(A\bar{x} + By^L + b) - \hat{\eta}^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) + \\ \hat{\mu}^\mathsf{T}(G(\bar{y}) - \nabla_y G(\bar{y})(\bar{y} - y^L)) + \sum_i \ell_i(\hat{\pi}) = \sum_i \hat{\alpha}_i. \end{array}$$

Note that, because of the definition of $\ell_i(\pi)$, we have $\sum_i \ell_i(\hat{\pi}) \le 0$. Together with the strict positiveness of $\sum_i \hat{\alpha}_i$, it follows that

$$\begin{array}{l} \hat{\pi}^\mathsf{T}(A\bar{x} + By^L + b) - \hat{\eta}^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) + \\ \hat{\mu}^\mathsf{T}(G(\bar{y}) - \nabla_y G(\bar{y})(\bar{y} - y^L)) > 0. \end{array}$$

Hence, we should impose the inequality (7) to the master problem

$$\begin{array}{l} \hat{\pi}^\mathsf{T}(Ax + By^L + b) - \hat{\eta}^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) + \\ \hat{\mu}^\mathsf{T}(G(\bar{y}) - \nabla_y G(\bar{y})(\bar{y} - y^L)) \le 0, \end{array} \tag{7}$$

which is a linear function of x acting as a cut. Notice that the construction of the cut (7) is achieved by solving both optimization problems (5) and (6). However, (6) is a linear program, the existing solvers, e.g. CPLEX [14], can handle this problem very efficiently.

Remark 1. *For a general optimization problem of the form (5), it can be still infeasible if we only introduce the auxiliary variable α for constraint $A\bar{x} + By + b - \alpha \le 0$. To construct a new optimization problem whose feasible set is definitely nonempty, we also have to deal with the constraints $G(y) \le 0$ and $H(y) = 0$. However, when designing an electric power grid, the architecture needs to guarantee that the system will be able to be adjusted to a new state that is stable and meets the demand if one contingency occurs. Mathematically speaking, for α is large enough, (5) is feasible in our application.*

Motivated by the previous analysis, we present here another valid cut provided that the optimal value of (5) is strictly positive, i.e., (4) is infeasible. Denote $(\bar{\pi}, \bar{\eta}, \bar{\mu})$ by the Lagrange multipliers corresponding to the constraints $A\bar{x}+By+b-\alpha \le 0$, $H(y) = 0$ and $G(y) \le 0$, respectively, in the nonlinear subproblem (5). From the complementary slackness condition, it implies $\bar{\mu}^\mathsf{T} G(\bar{y}) = 0$. From (7), we claim that

Theorem 1. *For a given $\bar{x}$, if (4) is infeasible, then the following is a cut for $\bar{x}$*

$$\begin{array}{l} \bar{\pi}^\mathsf{T}(Ax + By^L + b) - \bar{\eta}^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) - \\ \bar{\mu}^\mathsf{T}\nabla_y G(\bar{y})(\bar{y} - y^L) \le 0. \end{array} \tag{8}$$

Proof: We argue by contradiction. Assume that

$$\begin{array}{l} \bar{\pi}^\mathsf{T}(A\bar{x} + By^L + b) - \bar{\eta}^\mathsf{T}\nabla_y H(\bar{y})(\bar{y} - y^L) - \\ \bar{\mu}^\mathsf{T}\nabla_y G(\bar{y})(\bar{y} - y^L) \le 0. \end{array} \tag{9}$$

Denote $\Omega = \{(y, \alpha) : y^L \le y \le y^U, \alpha \ge 0\}$. We apply the

Karush-Kuhn-Tucker conditions for (5):

$$\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \sum_j \bar{\pi}_j \begin{pmatrix} B_{j:} \\ 0 \\ \vdots \\ -1 \\ \vdots \\ 0 \end{pmatrix} + \sum_j \bar{\eta}_j \begin{pmatrix} \nabla_y H_j(\bar{y}) \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \sum_j \bar{\mu}_j \begin{pmatrix} \nabla_y G_j(\bar{y}) \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} u_y \\ u_\alpha \end{pmatrix} = 0, \quad (10)$$

where $B_{j:}$ is the j-th row of matrix B, (u_y, u_α) is a vector in the normal cone $\mathcal{N}_{(\bar{y},\bar{\alpha})}(\Omega)$ of Ω at $(\bar{y}, \bar{\alpha})$, that is,

$$\begin{pmatrix} u_y \\ u_\alpha \end{pmatrix}^\mathsf{T} \begin{pmatrix} \bar{y} - y \\ \bar{\alpha} - \alpha \end{pmatrix} \geq 0,$$

for all $(y, \alpha) \in \Omega$. Now we select $(y, \alpha) = (y^L, \mathbf{0})$ and then multiply (10) by $(\bar{y} - y^L, \bar{\alpha})$, which yields

$$\sum_i \bar{\alpha}_i + \bar{\pi}^\mathsf{T} B(\bar{y} - y^L) - \bar{\pi}^\mathsf{T} \bar{\alpha} + \bar{\eta}^\mathsf{T} \nabla_y H(\bar{y})(\bar{y} - y^L) + \bar{\mu}^\mathsf{T} \nabla_y G(\bar{y})(\bar{y} - y^L) \leq 0.$$

Note that $\bar{\pi}^\mathsf{T} \bar{\alpha} = \bar{\pi}^\mathsf{T}(A\bar{x} + B\bar{y} + b)$, then we have

$$\sum_i \bar{\alpha}_i - \bar{\pi}^\mathsf{T}(By^L + A\bar{x} + b) + \bar{\eta}^\mathsf{T} \nabla_y H(\bar{y})(\bar{y} - y^L) + \bar{\mu}^\mathsf{T} \nabla_y G(\bar{y})(\bar{y} - y^L) \leq 0. \quad (11)$$

Combining $\sum_i \bar{\alpha}_i > 0$ with (9) and (11) we obtain the contradiction. ∎

Notice that our feasibility cut (8) is different from the cut in [6]. Our cut makes use of not only the information of the Lagrange multipliers of the linear constraints but also that of nonlinear ones as well as the bounds on the variables.

B. Description of Benders decomposition algorithm

We have introduced the feasibility cut (8). The utilization of the cut leads to the following Benders decomposition algorithm. The master problem is of the form:

$$\begin{aligned} \min\ & f(u_0) \\ \text{s.t.}\ & g_0(x_0, u_0) \leq 0 \\ & h_0(x_0, u_0) = 0 \\ & Pu_0 + q \leq 0, \end{aligned} \quad (12)$$

where $Pu_0 + q \leq 0$ consists of the feasibility cuts, P is a matrix and q is a vector.

BENDERS DECOMPOSITION ALGORITHM

1. Solve the following to get $(x_0^{(1)}, u_0^{(1)})$:

$$\begin{aligned} \min\ & f(u_0) \\ & g_0(x_0, u_0) \leq 0 \\ & h_0(x_0, u_0) = 0 \end{aligned}$$

2. For $k = 1, 2, \ldots$
   (a) For $c = 1, \ldots, C$:
   Check the feasibility of the subproblem
   $$\begin{aligned} & h_c(x_c, u_c) = 0 \\ & g_c(x_c, u_c) \leq 0 \\ & |u_c^i - u_0^{i\,(k)}| \leq \Delta u_i^{max}, i \in \mathcal{G}. \end{aligned}$$
   This can be done by solving a problem of form (5), where $(x_0^{(k)}, u_0^{(k)})$ is in place of $\bar{x}$.
   * If $\sum_i \alpha_i > 0$, then add the cut (7) or (8) into the master problem (12).
   * If $\sum_i \alpha_i = 0$ for all $c = 1, \ldots, C$, terminate the algorithm.
   (b) Solve the master problem (12) to obtain $(x_0^{(k+1)}, u_0^{(k+1)})$.

Remark 2 (Adaptive cuts). *It is known that SCOPF is a nonconvex optimization problem, if we use the conventional Benders cut without care, the optimal solutions will likely be pruned. Furthermore, it also deletes subregions of the feasible domain containing good feasible points, which follows that the generated solution is often far from the optimum. To ease this phenomena, we propose to utilize the cut in an adaptive manner. Suppose that $c(x) \leq 0$ is a valid cut for $\bar{x}$, it is plain to see that*

$$c(x) - \tau c(\bar{x}) \leq 0, \text{ for any } \tau < 1 \quad (13)$$

is also a cut. By choosing τ adaptively based on a constraint infeasibility measure for $\bar{x}$, we might improve the quality of the solution. For example, the measurement can be the optimal function value of the problem (5). The idea is that if the constraint infeasibility is large, we select τ close to 1, otherwise τ is close to 0. We will show later in the numerical experiments, this approach greatly improves the solution quality.

III. ALTERNATING DIRECTION METHOD OF MULTIPLIERS

This section investigates an approach that also helps to decompose the large-scale problem (1) into smaller subproblems. By reformulating the original problem, we can apply the alternating direction method of multipliers (ADMM) [15], [16], [17] to derive our algorithm. ADMM belongs to the class of first-order primal dual algorithms, that updates both primal and dual variables at each iteration. The method has been successfully applied to solve for various real-world applications, including image and signal processing [18], [19], [20], statistics and machine learning [21], and analytical target cascading in system design [22]. In power system analysis, to solve the classical optimal power flow problem without the security constraints, in [23], [24], Kim and Baldick split the power grid into a number of separate regions. By duplicating the variables in overlap regions, they were able to solve the distributed OPF problem by the ADMM. We now show how we can apply the method to solve for SCOPF.

Consider the general optimization problem with block separable structure

$$\min_{x,y} \{F(x) + G(y) : Mx + Ny = d, x \in X, y \in Y\}, \quad (14)$$

where $X \subset \mathbb{R}^n, Y \subset \mathbb{R}^m, M \in \mathbb{R}^{p\times n}, N \in \mathbb{R}^{p\times m}$ and $d \in \mathbb{R}^p$. We assume that F and G are closed, proper, convex and differentiable.

We form the augmented Lagrangian

$$\begin{aligned} L_\beta(x, y, \lambda) =& F(x) + G(y) + \lambda^T(Mx + Ny - d)+ \\ & \frac{1}{2}\beta\|Mx + Ny - d\|^2, \end{aligned} \quad (15)$$

where $\beta > 0$.

The classical augmented Lagrangian multiplier method [25], [26] involves a joint optimization and multiplier update step:

$$\begin{aligned} (x^{k+1}, y^{k+1}) &= \mathtt{argmin}_{x\in X, y\in Y}\ L_\beta(x, y, \lambda^k) \\ \lambda^{k+1} &= \lambda^k + \beta(Mx^{k+1} + Ny^{k+1} - d). \end{aligned}$$

In many applications, the first optimization problem is difficult to solve. To deal with the issue by using a Gauss-Seidel step, the alternating direction method of multipliers [15], [16] consists of the iterations

$$\begin{aligned} x^{k+1} &= \mathtt{argmin}_{x\in X}\ L_\beta(x, y^k, \lambda^k) \\ y^{k+1} &= \mathtt{argmin}_{y\in Y}\ L_\beta(x^{k+1}, y, \lambda^k) \\ \lambda^{k+1} &= \lambda^k + \beta(Mx^{k+1} + Ny^{k+1} - d). \end{aligned} \quad (16)$$

We observe that the optimization problem (1) can be reformulated by introducing auxiliary variables u_{0c} in a form suitable for ADMM as

$$\begin{aligned} \min\ & f(u_0) \\ s.t.\ & h_0(x_0, u_0) = 0 \\ & g_0(x_0, u_0) \le 0 \\ & h_c(x_c, u_c) = 0 \\ & g_c(x_c, u_c) \le 0 \\ & |u_c^i - u_{0c}^i| \le \Delta u_i^{max}, i \in \mathcal{G}, \\ & u_0^i - u_{0c}^i = 0, c = 1, \ldots, C. \end{aligned} \quad (17)$$

Let us consider

$$\begin{aligned} & x \triangleq (x_0, u_0), \\ & y \triangleq (x_1, u_1, \ldots, x_C, u_C), \\ & F(x) \triangleq f(u_0),\ G(y) \triangleq 0, \\ & X \triangleq \{(x_0, u_0) : h_0(x_0, u_0) = 0, g_0(x_0, u_0) \le 0\}, \\ & Y \triangleq (Y_1, \ldots, Y_C), \\ & \text{where} \\ & Y_c \triangleq \{(x_c, u_c, u_{0c}) : h_c(x_c, u_c) = 0, g_c(x_c, u_c) \le 0, \\ & |u_c^i - u_{0c}^i| \le \Delta u_i^{max}, i \in \mathcal{G}\}, c = 1, \ldots, C, \\ & Mx + Ny = d \triangleq u_0 - u_{0c} = 0,\ c = 1, \ldots, C, \end{aligned}$$

then the Lagrangian becomes

$$L_\beta(x, y, \lambda) = f(u_0) + \sum_{c=1}^{C} \lambda_c(u_0 - u_{0c}) + \frac{\beta}{2}\sum_{c=1}^{C} \|u_0 - u_{0c}\|^2.$$

The resulting ADMM algorithm is the following:

ADMM ALGORITHM

Initialize $u_{0c}^{(1)}, \lambda_c^{(1)}, c = 1, \ldots, C$

For $k = 1, 2, \ldots$

(a) $(x_0^{(k+1)}, u_0^{(k+1)}) = \mathtt{argmin}_{(x_0,u_0)\in X}\ f(u_0) + \sum_c \lambda_c^{(k)}(u_0 - u_{0c}^{(k)}) + \frac{\beta}{2}\sum_c \|u_0 - u_{0c}^{(k)}\|^2$

(b) For $c = 1, \ldots, C$:

$(x_c^{(k+1)}, u_c^{(k+1)}, u_{0c}^{(k+1)}) = \mathtt{argmin}_{(x_c,u_c,u_{0c})\in Y_c} \sum_c \lambda_c^{(k)}(u_0^{(k+1)} - u_{0c}) + \frac{\beta}{2}\sum_c \|u_0^{(k+1)} - u_{0c}\|^2,$

$\lambda_c^{(k+1)} = \lambda_c^{(k)} + \beta(u_0^{(k+1)} - u_{0c}^{(k+1)}).$

The algorithm is terminated either the number of iterations exceeds a pre-specified limit, or some stopping criterion is met. A widely used such stopping criterion [21] is

$$\begin{aligned} & \|u_0^{(k+1)} - u_{0c}^{(k+1)}\| \le \epsilon_1, \\ & \|u_{0c}^{(k+1)} - u_{0c}^{(k)}\| \le \epsilon_2,\ c = 1, \ldots, C. \end{aligned}$$

At each iteration, we break the SCOPF into $C+1$ subproblems with roughly the same size of the conventional optimal power flow problem, which, in turn, is well-suited for parallel computers. Note that the feasible sets of subproblems in the ADMM algorithm have not been modified over iterations, a significant computation effort can be saved by using warm-start techniques for the subproblems.

IV. Numerical Experiments

We use the following test electric power systems to demonstrate the efficiency of our proposed algorithms. Their characteristics are described in Table I.

- CH9: the 9 bus example from [27, p.70]
- NE39: the New England system [28]
- IEEE14, IEEE30, IEEE57, IEEE118 and IEEE300: the five IEEE systems, they can be found at http://www.ee.washington.edu/research/pstca/

Test system	Buses	Generators	Lines	# Contingencies
CH9	9	3	9	3
IEEE14	14	5	20	5
IEEE30	30	6	41	10
NE39	39	10	46	14
IEEE57	57	7	80	18
IEEE118	118	54	186	25
IEEE300	300	69	411	30

TABLE I: Test systems characteristics

The first column shows the abbreviations of the systems, while the second and third columns show the total number of buses and the number of generators in each system. The fourth column reports the number of lines interconnecting the buses. We artificially generated the list of contingencies, whose numbers of scenarios are presented in the remaining column. The contingency corresponds to the failure of a transmission line in these experiments; every active generator is able to reschedule up to 5% of its maximum power output.

The code was written in Matlab and all experiments were carried out on a PC using Matlab 7.10 with an Intel Xeon X5570 2.93 GHz under the Linux operating system. We terminated the ADMM algorithm when

$$\|u_0^{(k+1)} - u_{0c}^{(k+1)}\| \leq 10^{-3},$$
$$\|u_{0c}^{(k+1)} - u_{0c}^{(k)}\| \leq 10^{-3}.$$

We use a fixed parameter of $\beta = 1$ for the iterations.

First, we report the performance of the Benders decomposition using the cut (8) and the adaptive cut (8) with the following rule for selecting τ in (13):

if the optimal function value of (5) is larger than 1, then we choose $\tau = 0.8$; otherwise $\tau = 0.4$.

We omit to present the results for the cut (7) since it essentially gave a similarity with those of (8).

TABLE II: The performance of the Benders decomposition algorithms. CPU time in seconds

		Benders			Adaptive Benders		
Cases	Base	Cost	It	Time	Cost	It	Time
CH9	42.1	56.8	4	0.39	42.2	6	0.66
IEEE14	80.8	84.1	3	0.57	81.1	4	1.39
IEEE30	89.0	97.2	5	1.34	92.5	20	3.52
NE39	361.5	418.8	5	1.55	391.5	18	3.51
IEEE57	417.3	432.6	3	2.01	427.9	9	4.38
IEEE118	1296.6	1402.7	7	5.81	1332.9	18	13.97
IEEE300	7197.2	7325.2	5	11.38	7197.3	17	32.46

In Tables II and III, the base OPF cost "Base" represents the cost of the OPF problem without contingencies. The total generation cost from each solution method is denoted by "Cost", "Time" is the CPU time in seconds, and "It" is the total number of the required iterations. From Table II, we see that the regular Benders decomposition algorithm achieves a fast convergence rate (up to 7 iterations and 11.38 (s)); however it generates low quality solutions. Theoretically, the difficulty with this sub-optimality is understandable since hyperplanes from the feasibility cuts often prune the optimal solution from the non-convex feasible set. The adaptive scheme produces a much better solution; however it requires more iterations to converge. Its complexity grows linearly with the nodes on the system and the number of contingencies.

In Table III, the generated solutions from the adaptive Benders decomposition and ADMM are comparative; however ADMM is more robust, i.e., often obtains a lower generation cost. The execution time of the adaptive Benders decomposition is faster than ADMM for small-sized instances, but slower for large-sized ones. Finally, it shows that both algorithms are scalable, they are suitable for solving large-scale power systems with large numbers of contingencies.

TABLE III: The comparison of the adaptive Benders decomposition and ADMM. CPU time in seconds

		ADMM			Adaptive Benders		
Cases	Base	Cost	It	Time	Cost	It	Time
CH9	42.1	42.2	19	1.02	42.2	6	0.66
IEEE14	80.8	81.1	35	2.17	81.1	4	1.39
IEEE30	89.0	89.3	71	7.16	92.5	20	3.52
NE39	361.5	391.5	31	4.35	391.5	18	3.51
IEEE57	417.3	419.5	12	4.53	427.9	9	4.38
IEEE118	1296.6	1333.0	29	10.69	1332.9	18	13.97
IEEE300	7197.2	7197.4	16	23.86	7197.3	17	32.46

V. Conclusions

We have presented distributed algorithms based on the Benders decomposition and the alternating direction method of multipliers for solving the security-constrained optimal power flow problem. We have shown that if we use the conventional Benders cut without care, the generated solution will be rather far from the optimal operating point. By taking the non-convexity into account properly, our adaptive Benders decomposition often produces a solution with high quality. Furthermore, the ADMM algorithm is also able to yield a robust solution. These proposed techniques tackle each contingency individually, therefore they are able to solve for large-scale problems and can be done in parallel.

References

[1] J. Carpentier, "Contribution to the economic dispatch problem," *Bull Soc. France Elect.*, vol. 8, pp. 431–447, August 1962.

[2] A. J. Wood and B. F. Wollenburg, *Power Generation Operation and Control*. New York: John Wiley and Sons, 1996.

[3] M. Shahidehpour, H. Yamin, and Z. Li, *Market Operations in Electric Power Systems: Forecasting, Scheduling, and Risk Management*. New York: IEEE/Wiley-Interscience, 2002.

[4] P. Biskas and A. Bakirtzis, "A decentralized solution to the security constrained DC-OPF problem of multi-area power systems," in *Power Tech, 2005 IEEE Russia*, June 2005, pp. 1–7.

[5] O. Alsac and B. Stott, "Optimal load flow with steady state security," *IEEE Trans. Power App. Syst.*, vol. PAS-93, no. 3, pp. 745–751, 1974.

[6] A. Monticelli, M. V. F. Pereira, and S. Granville, "Security-constrained optimal power flow with post-contingency corrective rescheduling," *IEEE Trans. Power Syst.*, vol. 2, no. 1, pp. 175–180, 1987.

[7] F. Capitanescu, M. Glavic, D. Ernst, and L. Wehenkel, "Contingency filtering techniques for preventive security-constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 1690–1697, November 2007.

[8] O. Alsac, J. Bright, M. Prais, and B. Stott, "Further developments in LP-based optimal power flow," *IEEE Trans. Power Syst.*, vol. 5, no. 3, pp. 697–711, 1990.

[9] F. Bouffard, F. Galiana, and J. M. Arroyo, "Umbrella contingencies in security constrained optimal power flow," in *15th Power Systems Computation Conference (PSCC 05)*, Liège, Belgium, Aug 2005.

[10] D. Ernst, D. Ruiz-Vega, M. Pavella, P. M. Hirsch, and D. Sobajic, "A unified approach to transient stability contingency filtering, ranking and assessment," *IEEE Trans. Power Syst.*, vol. 16, no. 3, pp. 435–443, 2001.

[11] Y. Li and J. D. McCalley, "Decomposed SCOPF for improving efficiency," *IEEE Trans. Power Syst.*, vol. 24, 2009.

[12] A. M. Geoffrion, "Generalized Benders decomposition," *J. Optim. Theory Appl.*, vol. 10, no. 4, pp. 237–260, 1972.

[13] F. Capitanescu and L. Wehenkel, "A new iterative approach to the corrective security-constrained optimal power flow problem," *IEEE Trans. Power Syst.*, vol. 23, no. 4, pp. 1533–1541, November 2008.

[14] "IBM ILOG CPLEX Optimizer," http://www.ibm.com/software/integration/optimization/cplex-optimizer/.

[15] R. Glowinski and A. Marrocco, “Sur l’approximation par lments finis d’ordre un, et la rsolution, par pnalisation-dualit, d’une classe de problmes de Dirichlet non lineaires,” *RAIRO Anal. Numr.*, no. 2, pp. 41–76, 1975.

[16] D. Gabay and B. Mercier, “A dual algorithm for the solution of nonlinear variational problems via finite-element approximations,” *Computers and Mathematics with Applications*, vol. 2, pp. 17–40, 1976.

[17] J. Eckstein and D. P. Bertsekas, “On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, pp. 293–318, 1992.

[18] Y. Wang, J. Yang, W. Yin, and Y. Zhang, “A new alternating minimization algorithm for total variation image reconstruction,” *SIAM J. Imaging Sci*, pp. 248–272, 2008.

[19] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, “Fast image recovery using variable splitting and constrained optimization,” *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2345–2356, 2010.

[20] E. Esser, “Applications of Lagrangian-based alternating direction methods and connections to split Bregman,” UCLA, Los Angeles, Tech. Rep. CAM Rep. 0931, 2009.

[21] 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*, no. 3, pp. 1–124, 2011.

[22] S. Tosserams, L. F. P. Etman, P. Y. Papalambros, and J. E. Rooda, “An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers,” *Structural and Multidisciplinary Optimization*, vol. 31, no. 3, pp. 176–189, 2006.

[23] B. Kim and R. Baldick, “Coarse-grained distributed optimal power flow,” *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 932–939, 1997.

[24] ——, “A comparison of distributed optimal power flow algorithms,” *IEEE Transactions on Power Systems*, vol. 15, no. 2, pp. 599–604, 2000.

[25] M. R. Hestenes, “Multiplier and gradient methods,” *Journal of Optimization Theory and Applications*, vol. 4, pp. 303–320, 1969.

[26] M. J. D. Powell, “A method for nonlinear constraints in minimization problems,” *Optimization*, pp. 283–298, 1969.

[27] J. Chow, D. Frederick, and N. Chbat, *Discrete-Time Control Problems Using MATLAB*. CL Engineering, 2002.

[28] M. Pai, *Energy Function Analysis for Power System Stability*. Boston, MA: Kluwer Academic Publishers, 1989.