Scalable Decomposition Methods for Preventive Security-Constrained Optimal Power Flow

Brian Dandurand
Kibaek Kim
Mathematics and Computer Science Division
Argonne National Laboratory
Lemont, Illinois, United States of America
{bdandurand, kimk}@anl.gov

Abstract—In power system analysis, the alternating current optimal power flow (ACOPF) problem is modeled to minimize the total cost of generation while ensuring balance in the electrical network and addressing security considerations in which scenarios of partial network failure may occur. The ACOPF can be formulated as a nonconvex quadratically constrained quadratic optimization problem, which is well known to be challenging because of nonlinearity and nonconvexity. We present a decomposable reformulation of the ACOPF, in which variables are copied and linear consensus constraints are added. This decomposability is realized once the consensus constraints are relaxed, thus leading to a particular Lagrangian dual of the ACOPF. We apply a recently developed scalable dual solution approach to the ACOPF based on an augmented Lagrangian method (ALM) that integrates the proximal bundle method with the simplicial decomposition method (SDM) and a Gauss-Seidel method, called SDM-ALM, which is used to solve a primal, convexified characterization of the Lagrangian dual of the ACOPF. We provide computational results demonstrating the scalability of our dual solution approach.

Index Terms—parallel algorithm, dual decomposition, augmented Lagrangian, Gauss-Seidel method, nonconvex optimization, optimal power flow

I. INTRODUCTION

The efficient planning, operation, and control of electric power systems are commonly modeled in an alternating current optimal power flow (ACOPF) problem [1], [2]. Such problems are well known for their difficulty due to their nonlinearity and especially due to the presence of nonconvex constraints for power flow equation in the model. Local methods, especially interior-point methods, have been developed and applied to the ACOPF [3], [4] (see the introduction in [5] for more details). Other approaches make use of relaxations, such as convex relaxations derived from the semidefinite programming relaxation [6], that are used within a branch-and-bound framework [5].

Contingencies such as line, generator, or transformer failures need to be incorporated into the modeling of the optimal power flow to ensure that baseline power dispatch can be adjusted to satisfy the engineering and flow constraints as recourse to the realization of contingency event. The incorporation can be formulated as a two-stage decision model, where the first- and second-stage decisions represent the baseline dispatch and the contingency-specific adjustment, respectively. We call the resulting problem security-constrained optimal power flow (SCOPF, see [7] for a survey).

In this paper, we present the Lagrangian-based decomposition of the SCOPF model by introducing the copy of the baseline dispatch variables for each contingency scenario and enforcing the copies to have the same value for all over the contingency scenarios. As a result, the optimal value of the Lagrangian dual problem provides tight lower bounds of the minimum objective function value of the model. Such a decomposition approach has been used for stochastic mixed-integer linear programming problems (see [8] and references therein). Our approach enables the decomposition of the problem for each contingency scenario and is different from the Lagrangian dual approach in [5], where the problem is decomposed into two subproblems (i.e., one with active and reactive power variables; the other with voltage variables). The main limitation of the Lagrangian-based approaches is the requirement that the primal problem be convex; otherwise, an oracle is needed for solving scenario-wise nonconvex optimization problems. Note that the convexity assumption does not hold in our context due to the power flow equations of ACOPF.

We use the recently-developed method in [9], [10] for solving the nonconvex problems. In particular, the method is based on an augmented Lagrangian method (ALM) that integrates the simplicial decomposition method (SDM) and Gauss-Seidel method (abbreviated as SDM-ALM). These approaches are similar to alternating direction method of multipliers (ADMM) [11] and proximal bundle methods (e.g., see the survey in [12]). While the use of ADMM would be apt in a linearized direct current setting, its use in the nonconvex setting is unsupported theoretically and is heuristic. Furthermore, our solution approach is parallelizable and scalable because of its minimal communication bottleneck that consists essentially of
one reduce-sum. Surveys of parallel and distributed computing in optimal power flow problems may be found in [13], [14]. Other related work in solving ACOPF in parallel may be found in [15]. Parallel approaches using the parallel interior point solver PIPS [16] to leverage problem structure for similar problems may be found in [17].

The remainder of this paper is as follows. In Section II, we describe the optimization model for the SCOPF. Section III starts with the statement of a Lagrangian dual problem that is posed for a split-variable reformulation of the SCOPF. With that foundation, the last part of Section III provides the remaining developments necessary for the SDM-ALM algorithm, which is then stated at the end of this section along with the main convergence result. In Section IV, we describe the experimental setup, computational environment, and results. Section V gives our conclusion and describes future work.

II. SECURITY-CONSTRAINED ALTERNATING CURRENT OPTIMAL POWER FLOW PROBLEM

We base the following description of the AC power flow network and its topology on the formulation in [18].

A. Nomenclature

We define sets, parameters, and variables that are necessary and used for our optimization models.

Sets

- **G**: set of generators, indexed with \( g \)
- **I**: set of buses, indexed with \( i \) or \( j \)
- **L**: set of lines, indexed with \( l \)
- **C**: set of contingencies (including baseline), indexed with \( s \)
- **S**: set of proper contingencies (excluding baseline), indexed with \( s \)

Parameters

- \( P_{l}^{\max} \): line \( l \) power flow capacity
- \( P_{g}^{\min} \): minimum generator active power
- \( P_{g}^{\max} \): maximum generator active power
- \( Q_{g}^{\min} \): minimum generator reactive power
- \( Q_{g}^{\max} \): maximum generator reactive power
- \( V_{i}^{\min} \): minimum bus voltage magnitude
- \( V_{i}^{\max} \): maximum bus voltage magnitude
- \( Y_{s,i}^{f} \): shunt real power conductance when bus \( i \) is the “from” bus of a given line
- \( Y_{s,i}^{f} \): shunt imaginary power susceptance when bus \( i \) is the “from” bus of a given line
- \( Y_{l}^{f} \): line \( l \) power conductance (from)
- \( Y_{l}^{f} \): line \( l \) power susceptance (from)
- \( Y_{l}^{t} \): line \( l \) power conductance (to)
- \( Y_{l}^{t} \): line \( l \) power susceptance (to)
- \( Y_{s,j}^{f} \): shunt real power conductance when bus \( j \) is the “to” bus of a given line
- \( Y_{s,j}^{f} \): shunt imaginary power susceptance when bus \( j \) is the “to” bus of a given line

Variables

- \( p_{s,g}^{\gen} \): active power produced by each generator \( g \in G \) at contingency \( s \in C \), collected into the vector \( p_{s}^{\gen} := [p_{s,g}^{\gen}]_{g \in G} \)
- \( p_{s}^{\Delta} \): additional power added to (or subtracted from) generator \( g \) in the realization of the contingency associated with contingency \( s \)
- \( p_{s,l}^{f} \): active line \( l \) power flow (from), collected as \( p_{s}^{f} := [p_{s,l}^{f}]_{l \in L} \)
- \( p_{s,l}^{t} \): active line \( l \) power flow (to), collected as \( p_{s}^{t} := [p_{s,l}^{t}]_{l \in L} \)
- \( q_{s,i}^{f} \): reactive power produced by each generator \( g \in G \) at contingency \( s \in C \), collected into the vector \( q_{s}^{f} := [q_{s,g}^{f}]_{g \in G} \)
- \( q_{s,i}^{t} \): reactive line power flow (from), collected as \( q_{s}^{t} := [q_{s,l}^{t}]_{l \in L} \)
- \( q_{s,i}^{f} \): reactive line power flow (to), collected as \( q_{s}^{t} := [q_{s,l}^{t}]_{l \in L} \)
- \( q_{s,i}^{sh} \): shunt reactive power flow at bus \( i \), at contingency \( s \in C \)
- \( v_{s,i} \): voltage magnitude of the bus \( i \) at contingency \( s \in C \), collected as \( v_{s} := [v_{s,i}]_{i \in I} \)
- \( \theta_{s,i} \): voltage angle of the bus \( i \) at contingency \( s \in C \), collected as \( \theta_{s} := [\theta_{s,i}]_{i \in I} \)

Transformers are included in the consideration of the lines. Contingencies are indexed by nonnegative integers \( s \in C \), where \( s = 0 \) corresponds to a trivial baseline contingency where nothing adverse happens to the network, while each \( s \in S := C \setminus \{0\} \) refers to a proper contingencies where a system component does fail. The baseline dispatch of power is governed by the consideration of maintaining baseline bus voltages where any contingency is realized.

B. Formulation

The AC power flow must be subject to the following engineering bounds for each contingency \( s \in C \):

1. \( v_{i}^{\min} \leq v_{s,i} \leq v_{i}^{\max}, \quad \forall i \in I \)  
2. \( P_{g}^{\min} \leq p_{s,g} \leq P_{g}^{\max}, \quad \forall g \in G \)  
3. \(-0.05P_{g}^{\max} \leq p_{s}^{\Delta} \leq 0.05P_{g}^{\max}, \quad \forall g \in G \)  
4. \( q_{g}^{\min} \leq q_{s,g} \leq q_{g}^{\max}, \quad \forall g \in G \)  
5. \( \sqrt{(p_{s,l}^{f})^2 + (q_{s,l}^{f})^2} \leq \ell_{l}^{f}, \quad \forall l \in L \)  
6. \( \sqrt{(p_{s,l}^{t})^2 + (q_{s,l}^{t})^2} \leq \ell_{l}^{t}, \quad \forall l \in L \).
contingency $s \in C$ are given by
\[ p_{s,i}^f = Y_{s,l}^{fR} v_{s,i}^2 + Y_{s,i}^{ftf} u_{s,l}^2 + u_{s,l}^f Y_{s,i}^{ftf}, \quad \forall l \in L, \quad (2a) \]
\[ p_{s,i}^l = Y_{s,l}^{lt} v_{s,i}^2 + Y_{s,i}^{ltf} u_{s,l}^2 + u_{s,l}^l Y_{s,i}^{ltf}, \quad \forall l \in L, \quad (2b) \]
\[ q_{s,i}^f = Y_{s,l}^{fR} u_{s,i}^f - Y_{s,i}^{ftf} u_{s,l}^f - Y_{s,i}^{ftf} v_{s,j}^2, \quad \forall l \in L, \quad (2c) \]
\[ q_{s,i}^l = Y_{s,l}^{lt} u_{s,i}^l - Y_{s,i}^{ltf} u_{s,l}^l - Y_{s,i}^{ltf} v_{s,j}^2, \quad \forall l \in L, \quad (2d) \]
where \( i(l) \in I \) refers to the “from” bus of line \( l \), \( j(l) \in I \) refers to the “to” bus of line \( l \), and
\[ w_{s,l}^{fR} = v_{s,i}^l v_{s,j}^l \cos(\theta_{s,i} - \theta_{s,j}), \quad \forall l \in L, \quad (3a) \]
\[ w_{s,l}^{ftf} = v_{s,i}^l v_{s,j}^l \sin(\theta_{s,i} - \theta_{s,j}), \quad \forall l \in L, \quad (3b) \]
\[ w_{s,l}^{lt} = v_{s,j}^l v_{s,j}^l \cos(\theta_{s,j} - \theta_{s,i}), \quad \forall l \in L, \quad (3c) \]
\[ w_{s,l}^{ltf} = v_{s,j}^l v_{s,j}^l \sin(\theta_{s,j} - \theta_{s,i}), \quad \forall l \in L, \quad (3d) \]
(For a full description of the line conductance/susceptance
\[ Y_{s,l}^{fR}, Y_{s,i}^{ftf}, Y_{s,i}^{ltf}, Y_{s,i}^{ltf} \] and shunt conductance/susceptance
\[ Y_{s,l}^{fR}, Y_{s,i}^{ftf}, Y_{s,i}^{ltf} \] parameters, see, for example, [18].)

Shunt flow definitions associated with contingency
\[ s \in C \] are given by
\[ p_{s,i}^{I_n} = Y_{s,i}^{I_n} v_{s,i}^2, \quad \forall i \in I, \quad (4a) \]
\[ q_{s,i}^{I_n} = -Y_{s,i}^{I_n} v_{s,i}^2, \quad \forall i \in I. \quad (4b) \]

Flow balance constraints for contingency \( s \in C \) are given by

1) Real power
\[ p_{s,g(i)}^{gen} = p_{s,i}^g + p_{s,i}^{I_n} + \sum_{l \in L_s} p_{s,l}^f + \sum_{l \in L_s} p_{s,l}^l, \quad \forall i \in I, \quad (5a) \]

2) Reactive power
\[ q_{s,g(i)}^{gen} = q_{s,i}^g + q_{s,i}^{I_n} + \sum_{l \in L_s} q_{s,l}^f + \sum_{l \in L_s} q_{s,l}^l, \quad \forall i \in I, \quad (5b) \]
where \( L_s^f \) and \( L_s^l \) are, respectively, the sets of indices of the
“from” lines and the “to” lines of bus \( i \), and \( L_s \) are the lines
that are still operational in the event that the contingency \( s \) is
realized. (So \( L_0 \) contains the indices of all lines that are part
of the baseline network topology.)

Contingency real power generation in terms of base case
real power via power adjustments \( p_s^{\Delta} \) for each contingency
\( s \in S \) is
\[ p_{s,g}^{\Delta} = p_{s,g} - p_{s,g}^0, \quad \forall g \in G. \quad (6) \]

We now introduce the following definitions. For \( s = 0 \),
\[ z := (v_0, p_0^{gen}, q_0^{gen}, \theta_0), \]
For \( s \in S \)
\[ y_s := (v_s, p_s^{gen}, q_s^{gen}, \theta_s, p_s^{\Delta}), \quad y := (y_s)_{s \in S} \]

\[ \mathcal{Y}_s := \left\{ (z, y_s) : \begin{array}{l}
\text{the baseline } (s = 0) \text{ instance of (1)-(5) } \text{ holds for } z, \\
\text{the contingency } s \text{ instance of (1)-(6) } \text{ holds for } y_s 
\end{array} \right\}. \]

Note that the sets \( \mathcal{Y}_s, s \in S \), are closed but may be nonconvex.
The problem of interest is
\[ \min_{y,z} \left\{ \left( \frac{1}{|S|} \sum_{s \in S} f(x, y_s) : (x, y_s) \in \mathcal{Y}_s \forall s \in S \right) \right\} \ . \quad (7) \]
where \( f : \mathbb{R}^{2(|G|+|I|)} \rightarrow \mathbb{R} \) is a convex quadratic function
of the form
\[ f(z) = \sum_{g \in G} f_g(p_{s,g}^{gen}) \ . \quad (8) \]

III. LAGRANGIAN DECOMPOSITION AND STATEMENT OF
THE MAIN ALGORITHM

To realize a decomposable structure, we introduce copies
\( x_s \) of \( z \) for \( s \in S \). Then problem (7) can be reformulated as

\[ \begin{align*}
\min_{x,y,z} & \quad f(x, y_s) \\
\text{s.t.} & \quad (x, y_s) \in \mathcal{Y}_s \forall s \in S, \\
& \quad x_s = z \forall s \in S 
\end{align*} \ . \quad (9) \]

The Lagrangian function due to the relaxation of the
constraints \( x_s = z, s \in S \) is given by
\[ \frac{1}{|S|} \sum_{s \in S} \left[ f_s(x, y_s) + \lambda_s (x_s - z) \right] \ . \quad (10) \]

After imposing the following dual feasibility condition
\[ \lambda \in \Lambda := \left\{ \lambda : \sum_{s \in S} \lambda_s = 0 \right\}, \]
the term in \( z \) vanishes. Dropping the \( z \) term, we get
\[ \mathcal{L}(x, y, \lambda) := f(x, y_s) + \lambda_s^T x_s \text{ for } s \in S \]
\[ \mathcal{L}(x, y, \lambda) := \frac{1}{|S|} \sum_{s \in S} \mathcal{L}(x, y, \lambda_s), \quad (9b) \]
where we see that (9b) is additively separable along \( s \in S \).
Defining
\[ \phi_s(\lambda_s) := \min_{x,y} \left\{ \mathcal{L}(x, y, \lambda_s) : (x, y) \in \mathcal{Y}_s \right\}, \quad (10) \]
we have for dual function
\[ \phi(\lambda) := \min_{x,y} \left\{ \mathcal{L}(x, y, \lambda) : (x, y) \in \mathcal{Y} \right\} \quad (11) \]
\[ = \frac{1}{|S|} \sum_{s \in S} \phi_s(\lambda_s), \quad (12) \]
which is embedded in the following dual maximization problem:
\[ \max_{\lambda \in \Lambda} \{ \phi(\lambda) \}. \quad (13) \]
A. Continuous Master Subproblem

The augmented Lagrangian is defined by
\[ L^s_p(x_s, y_s, z, \lambda_s) := L_s(x_s, y_s, \lambda_s) + \frac{\rho}{2} \| x_s - z \|_2^2 \quad \text{for } s \in S, \]
\[ L^p(x, y, z, \lambda) := \frac{1}{|S|} \sum_{s \in S} L^s_p(x_s, y_s, z, \lambda_s), \]
where \( \rho > 0 \) is a penalty coefficient. Next, we define a convex inner approximation
\[ \hat{\gamma}_s \in \gamma_s, \quad \hat{\gamma} := \prod_{s \in S} \hat{\gamma}_s. \]
Then the continuous master subproblem given fixed \( \lambda \in \Lambda \) has the form
\[ \min_{x, y, z} \{ L^p(x, y, z, \lambda) : (x_s, y_s) \in \hat{\gamma}_s, \ s \in S \}. \tag{15} \]
A feasible solution to (15) (optimal or otherwise) is denoted \((\hat{x}, \hat{y}, \hat{\gamma})\). Typically, the solutions \((\hat{x}, \hat{y}, \hat{\gamma})\) obtained for problem (15) are only approximately optimal. In particular, they may be obtained through a Gauss-Seidel iterative approach, where, starting from an initial \( \hat{\gamma}^0 \), each iteration \( k \geq 1 \) solution \((\hat{x}^k, \hat{y}^k, \hat{\gamma}^k) \) is computed via
\[ (\hat{x}_s^k, \hat{y}_s^k) \in \arg\min_{x, y} \{ L^p(x_s, y_s, \hat{\gamma}^{k-1}, \lambda_s) : (x_s, y_s) \in \hat{\gamma}_s \}, \tag{16a} \]
\[ \hat{\gamma}^k \in \arg\min_{\bar{z}} \left\{ \frac{1}{|S|} \sum_{s \in S} L^p(\hat{x}_s^k, \hat{y}_s^k, z, \lambda_s) \right\}. \tag{16b} \]
Formally, we incorporate the iterative procedure given by (16a) and (16b) into Algorithm 1.

Algorithm 1 Gauss-Seidel iterations
1: function GS(\( \hat{\gamma}, \bar{\gamma}^0, \bar{\lambda}, k_{\max} \))
2: for \( k = 1 \) to \( k = k_{\max} \) do
3: Solve (16a) for \((\hat{x}_s^k, \hat{y}_s^k)_{s \in S}\) with \( \lambda = \bar{\lambda} \)
4: Solve (16b) for \( \bar{\gamma}^k \) with \( \lambda = \bar{\lambda} \)
5: end for
6: return \((\hat{x}_{s_{\max}}^k, \hat{y}_{s_{\max}}^k)_{s \in S}, \bar{\gamma}_{s_{\max}}^k \)
7: end function

B. Column Generation Subproblem

To prepare, we define two more problems. For \( \lambda \in \Lambda \), the linearized Lagrangian at \((\hat{x}_s, \hat{y}_s) \in \gamma_s\) for the scenario corresponding to contingency \( s \in S \) is given by
\[ \tilde{L}_s(x_s, y_s, \lambda_s, \hat{x}_s, \hat{y}_s) := L_s(\hat{x}_s, \hat{y}_s, \lambda_s) + \nabla_{x,y} L(\hat{x}_s, \hat{y}_s, \lambda_s)((x_s, y_s) - (\hat{x}_s, \hat{y}_s)), \tag{17a} \]
and the total linearized Lagrangian at \((\hat{x}, \hat{y}) \in \gamma\) then is
\[ \tilde{L}(x, y, \lambda, \hat{x}, \hat{y}) := \frac{1}{|S|} \sum_{s \in S} \tilde{L}_s(x_s, y_s, \lambda_s, \hat{x}_s, \hat{y}_s). \tag{17c} \]
Next, we define the linearized dual function at \((\hat{x}_s, \hat{y}_s) \in \gamma_s\)
\[ \tilde{\phi}_s(\lambda_s, \hat{x}_s, \hat{y}_s) := \min_{x_s, y_s} \{ \tilde{L}_s(x_s, y_s, \lambda_s, \hat{x}_s, \hat{y}_s) : (x_s, y_s) \in \gamma_s \} \tag{18a} \]
\[ \tilde{\phi}(\lambda, \hat{x}, \hat{y}) := \min_{x, y} \{ \tilde{L}(x, y, \lambda, \hat{x}, \hat{y}) : (x, y) \in \gamma \} \tag{18b} \]
\[ = \frac{1}{|S|} \sum_{s \in S} \tilde{\phi}_s(\lambda_s, \hat{x}_s, \hat{y}_s). \tag{18c} \]
Let the primal solutions realizing the minimizations of (18a) and (18b) be denoted \((\hat{x}_s, \hat{y}_s), s \in S\), and \((\hat{x}, \hat{y})\), respectively. As we collect these solutions \((\hat{x}, \hat{y})\), we refer to sets of these solutions using notation \( \hat{\gamma} := (\hat{\gamma}_s)_{s \in S} \). (In fact, we typically set \( \hat{\gamma} = \text{conv}(\hat{\gamma}) \).)

C. Dual Solution Update

At the beginning of each iteration of SDM-ALM, we take the null step \( \lambda^k = \lambda^{k-1} \). After having computed \( \hat{x}^k, s \in S \), and \( \hat{\gamma}^k \) during the iteration \( k \) solution of the continuous master problem, we may form the following conditional update for each \( s \in S \):
\[ \hat{x}_s \leftarrow \lambda^k_s + \rho(\hat{\gamma}^k_s - \hat{\gamma}^{k-1}_s) \quad \text{for } s \in S. \]

The condition for accepting \( \hat{\lambda} := (\hat{\lambda}_s)_{s \in S} \) as the new value of \( \lambda^k \) is given as follows. Set
\[ \gamma^k \leftarrow \frac{\tilde{\phi}(\lambda, \hat{x}^k, \hat{\gamma}^k) - \tilde{\phi}(\lambda, \hat{x}^{k-1}, \hat{\gamma}^{k-1})}{\tilde{L}(\hat{x}^k, \hat{\gamma}^k) - \tilde{L}(\hat{x}^{k-1}, \hat{\gamma}^{k-1})}, \]
and if the serious step condition \( \gamma^k \geq \gamma \) is satisfied for some fixed \( \gamma \in (0, 1) \), then we set \( \lambda^k = \hat{\lambda} \) (a serious step). Otherwise \( \lambda^k = \lambda^{k-1} \) remains true (a null step). While setting the conditional update \( \hat{\lambda} \) requires only solutions of the continuous master problem, the testing of the serious step condition requires additionally solutions of the column generation subproblem.

D. SDM-ALM Algorithm

Proposition 1. Let \( \{(\hat{x}_s^k, \hat{\gamma}_s^k, \lambda^k_s, \hat{\gamma}^k_s)\}_{k=1}^\infty \) be generated with Algorithm 2 applied to problem (8). The following hold:
1) \( \lim_{k \to \infty} \hat{\gamma}^k = \hat{\gamma}_s^0 \) exists and is the optimal value to the following problem:
\[ \min_{x, y, z} \left\{ \frac{1}{|S|} \sum_{s \in S} f_s(x_s, y_s) : \begin{array}{c}
(x_s, y_s) \in \text{conv}(\gamma_s) \forall s \in S, \quad x_s = z \forall s \in S
\end{array} \right\}, \tag{19} \]
where \( \text{conv}(\gamma_s) \) is the convex hull of \( \gamma_s \) for each \( s \in S \).
2) Each limit point \((\hat{x}^*, \hat{\gamma}^*, \lambda^*)\) is optimal for (19).

Proof. See Propositions 2 and 4 of [10].

Because problem (19) is a relaxation of problem (8), we note that each \( \hat{\gamma}^k \) is a lower bound on the optimal value of problem (8), as is \( \hat{\gamma}^0 \). However, as pointed out in [10], \( \hat{\gamma}^* \) is not as strong a lower bound in general as is the optimal
value of the dual problem (13) when nonlinearity is present in problem (8).

We envision SDM-ALM being applied to generate lower bounds within a branch-and-bound framework. Therefore, a brief comment on the need to generate upper bounds is warranted. Such upper bounds are typically generated heuristically by finding feasible solutions to problem (8). One such approach can be obtained by setting \( x_s = \bar{x}_{k_{max}} \) for each \( s \in S \), and solving the resulting decomposed \(|S|\) subproblems in terms of \( y \). When these subproblems are all solvable, we have constructed a feasible solution for problem (8), which is then used to provide an upper bound on the optimal value of (8).

IV. COMPUTATIONAL EXPERIMENTS

We test Algorithm 2 on the IEEE 118-bus AC network [19]–[21], where the system consists of 118 buses, 186 lines, 54 generators, and 99 loads. Furthermore, we consider 64 contingency scenarios generated in [17], where each contingency corresponds to the removal of one of the 186 lines from the baseline network. Hence, each scenario subproblem is modeled for recourse operations to a particular contingency relative to the baseline network topology. The algorithm is implemented in Julia, using the JuMP [22] interface with IPOPT [23] as the solver. The computational experiments are carried out on the Argonne National Laboratory Blues cluster, with parallel communication interfaced through the MPI wrapper of Julia. The computing environment consists of Intel® Xeon® CPU E7-4820 v2 @ 2.0GHz processors each with 16 cores. Because of the extensive memory requirements for this test problem, only one node per process is specified. Thus, for example, 32 processes require 32 nodes.

To illustrate the effect of the variation of the penalty parameter \( \rho \) on the trend of the dual bound, we let \( \rho \) vary, taking values \( \rho = 100, 1000, 5000, \) and 10000. For the serious step parameter \( \gamma \), we fix \( \gamma = 0.1 \). The maximum number of outer iterations is \( k_{max} = 20 \). And lastly, the initial \( \bar{\lambda}^0 \) and \((\bar{x}^0, \bar{y}^0)\) are obtained in the initialization phase of the code by solving the scenario-specific subproblems (10) with \( \lambda = 0 \).

Because of the nonconvexity of the constraint set \( \mathcal{Y} \) in the instances subproblem (18b), solutions to the subproblem in Line 8 of Algorithm 2 returned by a solver such as IPOPT may be only locally optimal. To improve the quality of the solution computed under this reality, we solve each instance of these subproblems ten times from randomized starting points input into the IPOPT solver, taking the solution corresponding to the best value. This repetitive process itself may be incorporated into the parallelization to improve scalability. Otherwise, another approach to solving the subproblems may be considered using, say, a branch-and-bound approach.

We apply Algorithm 2 to this problem for \( k_{max} \) iterations with varying numbers of processors. The wall time and speedup ratio due to varying numbers of processors are reported in Table I.

<table>
<thead>
<tr>
<th>Number of Processors</th>
<th>Wall Time (sec)</th>
<th>Speedup Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13965</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>7831</td>
<td>1.78</td>
</tr>
<tr>
<td>4</td>
<td>3610</td>
<td>3.87</td>
</tr>
<tr>
<td>8</td>
<td>1964</td>
<td>7.11</td>
</tr>
<tr>
<td>16</td>
<td>1098</td>
<td>12.72</td>
</tr>
<tr>
<td>32</td>
<td>834</td>
<td>16.74</td>
</tr>
<tr>
<td>64</td>
<td>695</td>
<td>20.09</td>
</tr>
</tbody>
</table>

The best bound obtained in the experiment is \( \bar{\lambda}^0 = 132178.4 \) using \( \rho = 10000 \), which is a lower bound on the optimal value of the SCOPF problem (8). For reference, the optimal value of the baseline 118 bus problem (with no contingencies) is 129660.7, which itself should also be a lower bound on the optimal value of problem (8) since the baseline problem is obtained from (8) by relaxing any constraints having to do with the contingency specific security constraints.

Taking the iteration-specific output data from the 64-processor experiment, we show the progression of the best bound on the optimal value of (8) with iterations \( k = 1, \ldots, 20 \) in Fig. 1 for varying values of \( \rho \). We see that the most rapid increase in the lower bounds are seen in the plots for \( \rho = 1000 \) and \( \rho = 5000.0 \). As noted in [10], the trend of the lower bounds is sensitive to the choice of \( \rho \), and the most favorable choices for \( \rho \) leading to the most rapid increase in lower bounds are not known beforehand. When the bound does not increase from one iteration to the next, we see the application of a null step in \( \lambda \) due to the serious step condition of Line 12 of Algorithm 2 not being met. ("Serious step" and "null step" are terms commonly used in the proximal bundle method literature.) The testing of this serious step condition guarantees
optimal convergence and can improve the increasing trend of the lower bounds, especially as the penalty parameter \( \rho \) grows larger (see the experimental results plotted in Fig. 1 of [10]).

V. CONCLUSION

Motivated by the development of the dual solution approach SDM-ALM for problems with nonconvexity and decomposability in their structure [10], we apply SDM-ALM to security constrained ACOPF problems (SCOPF). The security constraints introduce a large-scale decomposable structure, while the power flow constraints introduce nonconvexity. SDM-ALM was originally developed for stochastic mixed-integer optimization, and its application within SCOPF is new, where the nonconvexity arises from the AC power flow equations rather than mixed-integer restrictions. For stochastic mixed-integer optimization, well-established methods in mixed-integer optimization provide the necessary oracle solutions, while in the context of SCOPF, the required oracle solution approaches must use other solvers suited for non-linear and nonconvex optimization. Even within this context, the results shown in this paper are promising because they demonstrate the ability to obtain good quality lower bounds (assuming minimization) on the optimal value of the overall large-scale security constrained optimal power flow problem. Also from the computational experiments, we see promising trends in the parallel speedup. Note that there were only 64 contingencies, so any number of processors beyond 64 is not warranted in this experiment. We expect that for increased numbers of contingencies beyond 64, the speedup trend should improve. Additionally, improvement in the trend of the dual bounds may be realized by incorporating simple heuristics for dynamically tuning the penalty coefficient parameter similar to that found in [24].

Consequently, such an approach shows promise for application within a branch-and-bound or branch-and-cut framework, where dual bounds can be associated with the node lower bounds (assuming minimization).

The main bottleneck is in the solution to the linearized (column-generating) subproblem, where the problem objective function is linearized but the constraint set remains nonconvex. The convergence analysis of SDM-ALM assumes that this problem is solved to global optimality. This was a reasonable assumption in the original setting of large-scale linear mixed-integer problems motivating the development of SDM-ALM. To mitigate this problem arising from the nonconvex structure of the ACOPF power flow constraints, we have each subproblem instance of the linearized subproblem solved multiple times from random initial points. (This repeated solving provides another source to improve the scalability of the parallelization.) But improvements in this part of the algorithm may be made by introducing better-tailored approaches that may use, for example, a branch-and-bound framework with the use of appropriate convex relaxations. As the application of SDM-ALM to SCOPFs is improved, the next stage is to use the bounds produced therefrom within a branch-and-bound framework analogous to dual decomposition [25]. The use of such a framework also requires the generation of upper bounds on the optimal value of the SCOPFs. This is typically done heuristically; and, as noted in Section III, the solutions generated by SDM-ALM may be used to construct feasible solutions providing these bounds.

REFERENCES

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan. http://energy.gov/downloads/doe-public-access-plan.