

A Current-Voltage Successive Linear Programming Approach to Solving the ACOPF

Anya Castillo, *Student Member, IEEE*, Paula Lipka, *Student Member, IEEE*, Jean-Paul Watson, *Member, IEEE*, Shmuel Oren, *Fellow, IEEE*, Richard P. O'Neill, *Member, IEEE*

Abstract—We propose a successive linear programming (SLP) approach to solve the alternating current optimal power flow (ACOPF) problem, which we refer to as the SLP IV-ACOPF algorithm. Our goal is to develop an ACOPF linearization that can be readily extended and integrated into more complex decision processes, e.g. unit commitment, transmission switching. We demonstrate the computational performance and convergence quality of the SLP IV-ACOPF compared to an interior-point algorithm for solving the nonlinear ACOPF, on publicly available IEEE (14 to 300 buses) and Polish networks (2,383 to 3,375 buses) without and with thermal line limits. While the interior point run-time scales as a polynomial function of network size, run-time of the SLP IV-ACOPF demonstrates linear scaling on these same networks. We also report indicators of convergence quality, specifically the aggregated production bid costs, whether constraint relaxation was required for the solution, and the network power factors.

I. NOMENCLATURE

Sets:

\mathcal{N}	Set of buses $\{1, \dots, N\}$
\mathcal{K}	Set of lines $\{1, \dots, K\}$
$\mathcal{A}(n)$	Set of buses that are adjacent to node n
\mathcal{F}	Set of flows $\{1, \dots, 2K\}$

Indices:

h	Successive linear program iteration; $h \in \mathcal{H}$
n, m	Bus (node) indices; $n, m \in \mathcal{N}$
k	Three-phase transmission element; $k \in \mathcal{K}$
k'	Monitored flowgate; $k' \in \mathcal{K}'^{(h)} \subset \mathcal{K}$
$k(n, m)$	Flow on transmission element k from bus n to m ; $k(n, m) \in \mathcal{F}$ where $k(m, n)$ denotes the flow in the opposite direction along line k , $k(n, \cdot)$ denotes withdrawals from bus n , and $k(\cdot, n)$ denotes injections to bus n
$k(\cdot)$	Bidirectional flows on k ; $k(\cdot) \in \mathcal{F}$
l	Piecewise linear (pwl) segment; $l \in \mathcal{L}$

A. Castillo is with Johns Hopkins University, Baltimore, MD, USA, and Federal Energy Regulatory Commission (FERC), Washington, DC, USA anya.castillo@gmail.com

P. Lipka is with University of California, Berkeley, Berkeley, CA, USA plipka@berkeley.edu

J.-P. Watson is with Sandia National Laboratories, Albuquerque, NM, USA jwatson@sandia.gov

S. Oren is with University of California, Berkeley, Berkeley, CA, USA oren@ieor.berkeley.edu

R. P. O'Neill is with Federal Energy Regulatory Commission (FERC), Washington, DC, USA richard.oneill@ferc.gov

Variables:

p_n^g	Total linearized real power generation at bus n
$p_{n,l}^g$	Linear segment l of generation at bus n
q_n^g	Reactive power generation at bus n
v_n	Voltage magnitude at bus n
v_n^{sq}	Linearization of $(v_n)^2$
v_n^r	Real part of voltage at bus n
v_n^j	Imaginary part of voltage at bus n
i_n^r	Real part of current injection at bus n
i_n^j	Imaginary part of current injection at bus n
$i_{k(n,m)}$	Current magnitude on k from bus n to m
$i_{k(n,m)}^{sq}$	Linearization of $(i_{k(n,m)})^2$
$i_{k(n,m)}^r$	Real part of current on k from bus n to m
$i_{k(n,m)}^j$	Imaginary part of current on k from bus n to m

Parameters:

B^{MVA}	Power base in MVA
R_k	Series resistance of line k
X_k	Series reactance of line k
G_k	Series conductance of line k
B_k	Series susceptance of line k
Y_k	Series admittance of line k ; $Y_k = G_k + jB_k$
G_{kn}^{sh}	Shunt conductance on line k connected to n
B_{kn}^{sh}	Shunt susceptance on line k connected to n
Y_{kn}^{sh}	Shunt admittance on line k connected to n
G_n^{sh}	Shunt conductance at bus n
B_n^{sh}	Shunt susceptance at bus n
Y_n^{sh}	Shunt admittance at bus n
τ_{kn}	Ideal transformer on the n -side of line k
$ \tau_{kn} $	Transformer turns ratio on the n -side of line k
ϕ_{kn}	Phase-shifter on the n -side of line k
P_n^d	Real power demand at bus n
Q_n^d	Reactive power demand at bus n
P_n^{min}	Minimum real power for generation at bus n
P_n^{max}	Maximum real power generation at bus n
Q_n^{min}	Minimum reactive power generation at bus n
Q_n^{max}	Maximum reactive power generation at bus n
V_n^{min}	Minimum voltage magnitude at bus n
V_n^{max}	Maximum voltage magnitude at bus n
I_n^{max}	Maximum current magnitude on line k
$V_n^{(h)}$	Step-size bound on the voltage at bus n in iter h
$C_n^{g,2}$	Quadratic cost coefficient for generation at bus n
$C_n^{g,1}$	Linear cost coefficient for generation at bus n
C_n^q	Linear cost coefficient for reactive support at bus n
$C_{n,l}^g$	Linear segment l of the quadratic cost at bus n
$P_{n,l}^g$	Maximum length of piecewise segment l at bus n

II. INTRODUCTION

THE alternating current optimal power flow (ACOPF) problem, also known as the OPF, is solved by system operators to co-optimize real and reactive power dispatch, to promote reliable operation and implement efficient markets.

The ACOPF originated from Carpentier's reformulation of the economic dispatch problem based on the Karush-Kuhn-Tucker (KKT) conditions [1]. The ACOPF is a nonconvex, nonlinear optimization problem, for which finding globally optimal solutions is known to be non-deterministic polynomial-time (NP) hard. Consequently, this key dispatch problem is not presently solved exactly. Instead, Independent System Operators (ISOs) and other grid operators use approximate solution techniques based on linear programming (LP) and mixed-integer programming (MIP). Leveraging significant recent performance improvements in commercial LP and MIP solvers, dispatches and prices can be obtained within the required time limits. Such approximations are additionally embedded within other optimization problems to address power system restoration, topology control, security-constrained unit commitment, and transmission planning.

In deregulated electricity markets, OPF tractability is critical to support efficient commitment, dispatch and market clearing strategies based on Locational Marginal Prices (LMPs). Presently, the dominant pricing method in deregulated US markets involves computation of the nodal LMP which accounts for network congestion and line losses. Approximation methods that linearly relate MW controls to MW flows on the network are employed for both dispatch and price formation. Independent System Operators (ISOs) commonly use the direct current approximation of the OPF, known as the DCOPF. The DCOPF solely accounts for real power dispatch; reactive power dispatch is determined subsequently through a corrective process. Specifically, full alternating current (AC) feasibility is achieved through an iterative, quasi-optimization process to ensure that a realistic engineering solution is obtained by the DCOPF model, and to identify constraint violations that may require preventive actions including re-dispatch, reactive power compensation, or voltage support. Another common approach is to solve a decoupled ACOPF model, which iterates between $P-\Theta$ and $Q-V$ subproblems. Under normal operating conditions, the DCOPF and the decoupled ACOPF model provide reasonably accurate approximations of the real power dispatch and the associated LMPs observed in AC network models. The models deviate from an acceptable AC feasible solution when the system is stressed, e.g., when there exists a strong physical coupling between real and reactive power (e.g., as specified by generator capability curves) or when voltage limits restrict real power dispatch.

There is over half a century of work on the OPF. In the literature to date, various reformulations, decomposition methods and algorithms have been proposed; we refer to [2], [3], [4] and [5] for thorough surveys. Here, we focus on methods that solve the OPF via approaches based on linear or mixed-integer programming.

Methods that use a MIP to solve the OPF introduce binary variables to model nonconvex, nonlinear functions with a

piecewise linear representation. Zhang et al. [6] linearize power flows by treating the off-nominal bus voltages as variables and then introduce piecewise linear functions with associated binary variables to account for network losses, while simultaneously preventing fictitious losses. The authors then demonstrate the accuracy and computational speed of the proposed algorithm on multiple test cases [7], ranging in size from 14-bus to 3,120-bus networks. Some researchers have used this MIP reformulation of the ACOPF to analyze steady-state operations on radial, as opposed to more general meshed, networks. Borghetti et al. [8] consider the minimum loss reconfiguration problem of distribution networks and approximate the nonlinearities associated with Kirchhoff's voltage law using piecewise linear functions. The authors present results for four networks, with the largest being a 69-bus (74-line) network. Ferreira et al. [9] solve the same minimum loss reconfiguration problem by introducing disjunctive constraints, formulation of piecewise linear functions using specially ordered sets of type 2 (SOS2), and McCormick's convex envelope to model bilinear products. The authors report limited results on 16-bus and 35-bus networks.

Other linear ACOPF approximations do not introduce binary variables, and thus can be solved efficiently. Franco et al. [10] apply a least-squares regression to obtain a non-iterative linear approximation of the OPF in terms of the real and imaginary voltage components. The approximation in [10] does not consider numerous physical constraints (e.g., phase angles, voltage magnitudes, reactive power, and thermal line limits), and only limited results for a 136-bus network are provided. Mohapatra et al. [11] formulate the OPF in terms of incremental variables and solve the nonlinear formulation by applying Newton's power flow and the primal-dual interior point method. Although the authors report results for the IEEE 14-bus, 118-bus, and 300-bus networks along with a fictitious 1,000-bus network, the proposed formulation and subsequent testing does not consider phase angles and thermal line limits. Coffrin et al. [12] use voltage estimates to derive a piecewise linear representation of apparent power through the introduction of an additional thirteen constraints per line to approximate the associated convex hull, with breakpoints in the range of $\pm\pi/12$ for small angle differences. In further work, Coffrin et al. [13] introduce a piecewise linear approximation of the cosine term in the power flow equations, and use initialization techniques based on Taylor series expansions to solve for the remaining nonlinear terms.

Many recent studies apply convexification techniques to the ACOPF. Bai et al. [14] propose a semidefinite relaxation (SDR) for which Lavaei and Low [15] derive a sufficient condition under which the SDR is exact, i.e., a globally optimal solution is guaranteed. Lavaei and Low further prove that if the network is radial, then this sufficient condition always holds [15]. A shortcomings of this relaxation is that it provides no mechanism to recover a feasible solution when the sufficient condition is not satisfied. There are also practical difficulties in efficiently implementing a semidefinite program in branch-and-bound context, including the lack of an initialization method and limitations in scalability due to the number of linear algebraic iterations required during the

solution process. Furthermore, Lesieutre et al. [16] illustrate practical scenarios where the SDR fails to produce physically meaningful solutions.

In this work, we propose a novel current-voltage (IV) successive linear programming (SLP) approach to solving the ACOPF, which we refer to as the SLP IV-ACOPF. We solve a linear system of equations without decomposition, and represent network flows linearly in terms of complex current instead of apparent power. We apply a combination of linearization and reduction techniques to the problem constraints. We iteratively co-optimize real and reactive power dispatch, and introduce no binary variables in our approach. Our approach can be extended to include discrete controls and embedded within branch-and-bound algorithms to support more complex decision processes. In Sections III and IV, we summarize and detail our SLP IV-ACOPF algorithm. In Section V, we demonstrate the computational performance and convergence quality for our method. We conclude in Section VI with a brief discussion of our results.

III. ALGORITHM OUTLINE

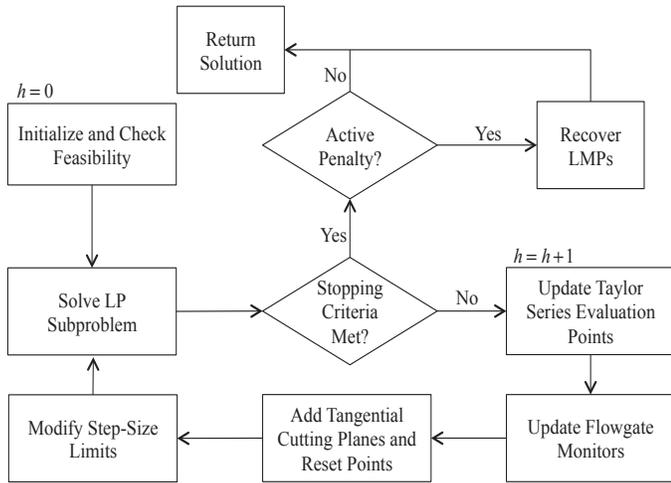


Fig. 1: A successive linear programming approach to solve the IV-ACOPF.

Our overall solution strategy in the SLP IV-ACOPF is depicted in the process diagram of Figure 1. In the initial iteration ($h = 0$), we initialize variables and the evaluation point for the Taylor series approximations using a flat start, randomized start, or warm start. If the calculated power flows from this initialization are not feasible, we update the initializations to reside within constraint bounds. We then iteratively solve the resulting LP subproblem. Following each iteration h , we check whether the solution meets either of the following stopping criteria: (1) the solution is ACOPF feasible within a specified tolerance or (2) a maximum iteration limit is reached (see Appendix VII-C). If neither of these criteria are met, we update the Taylor series evaluation points $\hat{v}_n^{r(h)}$, $\hat{v}_n^{j(h)}$, $\hat{i}_n^{r(h)}$, $\hat{i}_n^{j(h)}$, $\hat{i}_{k(\cdot)}^{r(h)}$, and $\hat{i}_{k(\cdot)}^{j(h)}$; for an arbitrary evaluation point $\hat{x}^{(h)}$, we have that $\hat{x}^{(h)} = x^*$ from iteration $h - 1$.

Following the Taylor series evaluation point update, we update all flowgate monitors—to identify lines that are near or

at capacities. For any evaluation points that are ACOPF infeasible, we reset these parameters to be within their bounds and accordingly add tangential cutting planes to enforce ACOPF feasibility in the following iteration. The step size bounds, which limit the approximation error, are also modified before resolving the LP subproblem.

Upon termination, the algorithm can yield one of the following outcomes: (1) a KKT optimal solution to the ACOPF is identified, (2) the SLP IV-ACOPF optimal solution is ACOPF feasible but not optimal, (3) the SLP IV-ACOPF optimal solution is ACOPF infeasible, or (4) the SLP IV-ACOPF is infeasible. Results meeting criteria (1) through (3) may be meaningful in practice. An additional step to recover the non-penalized LMPs is required if the criteria (3) is met. An outcome of (4) indicates that either the SLP IV-ACOPF requires a better initialization, the step size used is too small, or that the ACOPF is unbounded or has no solution.

IV. ALGORITHM DETAILS

In this section, we first present the network model in IV-A and the canonical problem formulation in IV-B. In IV-C, we linearize and reduce the nonlinear ACOPF in order to construct the LP subproblem in IV-D.

A. Network Model

We assume balanced three-phase, steady-state conditions; the nomenclature is in Section I. We formulate the nonlinear ACOPF and the subsequent LP subproblem in rectangular coordinates for the voltage phasor $\mathbb{V}_n = v_n^r + jv_n^j$ at each bus $n \in \mathcal{N}$, the current phasor $\hat{i}_n = i_n^r + ji_n^j$ at each bus $n \in \mathcal{N}$, and the current phasor $\hat{i}_{k(\cdot)} = i_{k(\cdot)}^r + ji_{k(\cdot)}^j$ on all network flows $k(\cdot) \in \mathcal{F}$.

Applying the π -model, we determine the series conductance G_k and series susceptance B_k as

$$G_k = R_k / (R_k^2 + X_k^2) \quad (1)$$

$$B_k = -X_k / (R_k^2 + X_k^2). \quad (2)$$

In order to characterize the resistive losses and leakage flux (i.e. self-reactance), we model a practical transformer that is located on the bus n side as an ideal transformer with turns ratio $|\tau_{kn}|$ in series with a series admittance $Y_k = G_k + jB_k$. Depending on if τ_{kn} is real or complex, the transformer is in-phase or phase-shifting. We can similarly represent a phase-shifter as $\tau_{kn} = |\tau_{kn}| e^{j\phi_{kn}}$. For the branch admittance matrix

$$\begin{bmatrix} Y_{1,1}^k & Y_{1,2}^k \\ Y_{2,1}^k & Y_{2,2}^k \end{bmatrix} = \begin{bmatrix} |\tau_{kn}|^2 (Y_k + Y_{kn}^{sh}) & -\tau_{kn}^* \tau_{km} Y_k \\ -\tau_{kn} \tau_{km}^* Y_k & |\tau_{km}|^2 (Y_k + Y_{km}^{sh}) \end{bmatrix}, \quad (3)$$

we model the complex current flows on line k as

$$\begin{bmatrix} \hat{i}_{k(n,m)} \\ \hat{i}_{k(m,n)} \end{bmatrix} = \begin{bmatrix} Y_{1,1}^k & Y_{1,2}^k \\ Y_{2,1}^k & Y_{2,2}^k \end{bmatrix} \times \begin{bmatrix} \mathbb{V}_n \\ \mathbb{V}_m \end{bmatrix}. \quad (4)$$

The above representation is for a two-winding transformer, and if $\tau_{kn} = \tau_{km} = 1$, then the equivalent π -model is of a transmission line. For an N -winding transformer, we would

have a Y^k matrix of size $N \times N$ for the unified branch model. We can represent the linear relationship between the real and imaginary parts of the complex current flows, $\hat{i}_{k(n,m)}$ and $\hat{i}_{k(m,n)}$, and the complex nodal voltages, \mathbb{V}_n and \mathbb{V}_m , as

$$i_{k(n,m)}^r = \text{Re}(Y_{1,1}^k \mathbb{V}_n + Y_{1,2}^k \mathbb{V}_m) \quad (5)$$

$$i_{k(n,m)}^j = \text{Im}(Y_{1,1}^k \mathbb{V}_n + Y_{1,2}^k \mathbb{V}_m) \quad (6)$$

$$i_{k(m,n)}^r = \text{Re}(Y_{2,1}^k \mathbb{V}_n + Y_{2,2}^k \mathbb{V}_m) \quad (7)$$

$$i_{k(m,n)}^j = \text{Im}(Y_{2,1}^k \mathbb{V}_n + Y_{2,2}^k \mathbb{V}_m) \quad (8)$$

for all flows $k(\cdot) \in F$. The nodal current balance for each bus $n \in \mathcal{N}$ is

$$i_n^r - \left(\sum_{k(n,\cdot)} i_{k(n,m)}^r + G_n^{sh} v_n^r - B_n^{sh} v_n^j \right) = 0 \quad (9)$$

$$i_n^j - \left(\sum_{k(n,\cdot)} i_{k(n,m)}^j + G_n^{sh} v_n^j + B_n^{sh} v_n^r \right) = 0. \quad (10)$$

B. Canonical Formulation

We formulate the nonlinear ACOPTF in rectangular coordinates where we balance on the above network current flows. Instead of introducing the square root ($\sqrt{\cdot}$) operator, we compute the squared voltage magnitude $(v_n)^2 = (v_n^r)^2 + (v_n^j)^2$ for all buses $n \in \mathcal{N}$ and the squared current magnitude $(i_{k(\cdot)})^2 = (i_{k(\cdot)}^r)^2 + (i_{k(\cdot)}^j)^2$ for all flows $k(\cdot) \in \mathcal{F}$. The canonical formulation is

$$\min \sum_{n \in \mathcal{N}} C_n^{g,2} (p_n^g)^2 + C_n^{g,1} p_n^g \quad (11)$$

subject to

$$(5) - (10) \quad (12)$$

$$p_n^g - (v_n^r i_n^r + v_n^j i_n^j) = P_n^d \quad (13)$$

$$q_n^g - (v_n^j i_n^r - v_n^r i_n^j) = Q_n^d \quad (14)$$

$$P_n^{min} \leq p_n^g \leq P_n^{max} \quad (15)$$

$$Q_n^{min} \leq q_n^g \leq Q_n^{max} \quad (16)$$

$$(V_n^{min})^2 \leq (v_n)^2 \leq (V_n^{max})^2 \quad (17)$$

$$(i_{k(\cdot)})^2 \leq (I_k^{max})^2 \quad (18)$$

for all $n \in \mathcal{N}$ in (13) – (17) and all $k(\cdot) \in \mathcal{F}$ in (18). The generator bid curves in the objective function (11) are convex quadratic. The upper bounds in (17) and (18) are nonlinear and convex; equations (13), (14), and the lower bound in (17) are nonlinear and nonconvex.

C. Linearization and Reduction Methods

We apply approximations, relaxations, and penalty variables in reformulating the nonlinearities in the canonical formulation of (11) – (18). We also reduce the constraint set to monitor flowgate limits which are near or at the bound.

1) *Piecewise Linear Interpolations*: A piecewise linear interpolation can be applied to approximate the quadratic generator bid curve in (11) where the generators bid at marginal cost. Typically, generator bid curves are monotonically increasing, and by partitioning the interval into more linear segments, this approach results in a tighter upper bound on the quadratic cost function.

To construct the piecewise linear function, we partition the interval $[P_n^{min}, P_n^{max}]$ into $|\mathcal{L}|$ linear segments with length $P_{n,l}^g = (P_n^{max} - P_n^{min}) / |\mathcal{L}|$. There are $|\mathcal{L}|+1$ points where the l -th segment is associated with points

$$[x_l, x_{l+1}] := \left[P_n^{min} + l P_{n,l}^g, P_n^{min} + (l+1) P_{n,l+1}^g \right]. \quad (19)$$

For $l = 0, \dots, |\mathcal{L}|$, we have that $x_0 < x_1 < \dots < x_{|\mathcal{L}|}$.

For each segment $l \in \mathcal{L}$, we calculate the midpoint of the linear interpolation between points $(x_l, f_n^r(x_l))$ and $(x_{l+1}, f_n^r(x_{l+1}))$. Applying the slope of the bid curve, the resulting cost coefficient is

$$C_{n,l}^g = C_n^{g,1} + (B^{MVA})^2 C_n^{g,2} (x_l + x_{l+1}) \quad (20)$$

for all segments $l \in \mathcal{L}$ and buses $n \in \mathcal{N}$; the $(B^{MVA})^2$ accounts for any per-unit scaling of the power variables. We approximate the aggregate bid curve in (11) as

$$\text{bids}(\cdot)^{(h)} = \sum_{n \in \mathcal{N}} \sum_{l \in \mathcal{L}} C_{n,l}^g p_{n,l}^g + C_n^0 \quad (21)$$

for iteration h , where $C_n^0 = C_n^{g,2} (P_n^{min})^2 + C_n^{g,1} P_n^{min}$ when $P_n^{min} > 0$.

Furthermore, each segment $p_{n,l}^g$ of the piecewise linear function is limited by $P_{n,l}^g$, that is

$$p_{n,l}^g \leq P_{n,l}^g \quad (22)$$

for all $n \in \mathcal{N}$ and $l \in \mathcal{L}$. The aggregate of the segments at bus n must equal the real power generation

$$p_n^g = \sum_{l \in \mathcal{L}} p_{n,l}^g + P_n^{min} \quad (23)$$

for all $n \in \mathcal{N}$.

2) *Taylor Series Approximations*: We apply first order Taylor series approximations to address the nonlinear terms $p_n^g, q_n^g, (v_n)^2$ and $(i_{k(\cdot)})^2$ in constraints (13), (14), (17) and (18), respectively. For iteration h , we use the Taylor series evaluation points (denoted with a caret) to approximate the first order linearizations

$$v_n^{sq} = 2\hat{v}_n^{r(h)} v_n^r + 2\hat{v}_n^{j(h)} v_n^j - \left(\hat{v}_n^{r(h)} \right)^2 - \left(\hat{v}_n^{j(h)} \right)^2 \quad (24)$$

$$p_n^g = \hat{v}_n^{r(h)} \hat{i}_n^r + \hat{v}_n^{j(h)} \hat{i}_n^j + v_n^r \hat{i}_n^r + v_n^j \hat{i}_n^j - \hat{v}_n^{r(h)} \hat{i}_n^r - \hat{v}_n^{j(h)} \hat{i}_n^j + P_n^d \quad (25)$$

$$q_n^g = \hat{v}_n^{j(h)} \hat{i}_n^r - \hat{v}_n^{r(h)} \hat{i}_n^j + v_n^r \hat{i}_n^r - v_n^j \hat{i}_n^j - \hat{v}_n^{j(h)} \hat{i}_n^r + \hat{v}_n^{r(h)} \hat{i}_n^j + Q_n^d \quad (26)$$

for all buses $\forall n \in \mathcal{N}$, and

$$i_{k(\cdot)}^{sq} = 2\hat{i}_{k(\cdot)}^{r(h)} i_{k(\cdot)}^r + 2\hat{i}_{k(\cdot)}^{j(h)} i_{k(\cdot)}^j - \left(\hat{i}_{k(\cdot)}^{r(h)} \right)^2 - \left(\hat{i}_{k(\cdot)}^{j(h)} \right)^2 \quad (27)$$

for all flows $\forall k(\cdot) \in \mathcal{F}$. Since a first order method is used, larger step sizes result in larger approximation error. Therefore, we must require step sizes that are small enough to gain higher accuracy (i.e. lower truncation error) but large enough to minimize the required number of iterations. Depending on the penalty and real power mismatch costs, the step size is restricted at an accelerated or decelerated rate; see Appendix VII-A. At each iteration h , we update the tunable parameter $V_n^{(h)}$ and introduce the step size limits

$$\left| v_n^r - \hat{v}_n^{r(h)} \right| \leq V_n^{(h)} \quad (28)$$

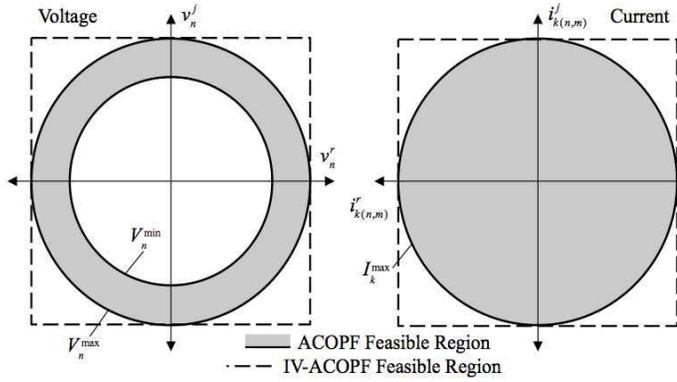


Fig. 2: Outer approximation of the voltage and current phasor bounds with box constraints.

$$\left| v_n^j - \hat{v}_n^{j(h)} \right| \leq V_n^{(h)} \quad (29)$$

on the real and imaginary parts of the nodal voltage, v_n^r and v_n^j , for all buses $n \in \mathcal{N}$. By controlling the step size for the real and imaginary parts of the nodal voltages, we limit the approximation error in the real and reactive power.

3) *Relaxations and Penalty Factors*: In conjunction with (24) and (27), we introduce the following box constraints on the real and imaginary parts of the nodal voltage as

$$-V_n^{max} \leq v_n^r \leq V_n^{max} \quad (30)$$

$$-V_n^{max} \leq v_n^j \leq V_n^{max} \quad (31)$$

for all buses $\forall n \in \mathcal{N}$, and also on the real and imaginary parts of the current flow as

$$-I_k^{max} \leq i_{k(\cdot)}^r \leq I_k^{max} \quad (32)$$

$$-I_k^{max} \leq i_{k(\cdot)}^j \leq I_k^{max} \quad (33)$$

for all flows $\forall k(\cdot) \in \mathcal{F}$, in order to bound our approximation, as illustrated in Figure 2.

However, the relaxations along with the first order Taylor series approximations can result in ACOPF infeasible solutions. When this occurs due to violations of the voltage (17) or current (18) upper bound, we reset the evaluation points to the Taylor series approximation that resulted in an ACOPF infeasible solution. We also include a tangential cutting plane to the constraint set for the subsequent iteration, as illustrated in Figure 3. We impose constraint satisfaction of the tangential cutting plane by introducing a slack variable, which is penalized in the cost function; see Appendix VII-B. This approach only applies for the outer approximation on the upper bounds.

When the lower bound constraint in (17) is violated, we do not introduce a tangential cutting plane, which would eliminate parts of the ACOPF feasible region. Instead, we only impose constraint satisfaction by introducing slack variables that are penalized in the cost function. We treat the bounds on the first order Taylor approximations similarly. As a result we reformulate (15) – (18) as

$$P_n^{min} - p_n^{viol,-} \leq p_n^g \leq P_n^{max} + p_n^{viol,+} \quad (34)$$

$$Q_n^{min} - q_n^{viol,-} \leq q_n^g \leq Q_n^{max} + q_n^{viol,+} \quad (35)$$

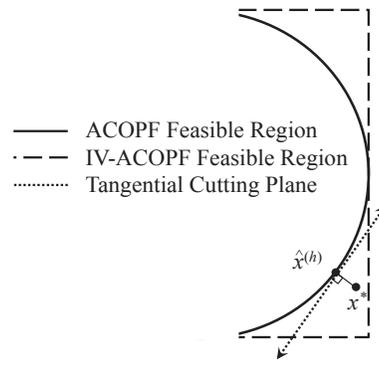


Fig. 3: The infeasible solution x^* from iteration $h - 1$, the updated evaluation point $\hat{x}^{(h)}$, and the tangential cutting plane included to the constraint set for iteration h .

$$(V_n^{min})^2 - v_n^{viol,-} \leq v_n^{sq} \leq (V_n^{max})^2 + v_n^{viol,+} \quad (36)$$

$$i_{k(\cdot)}^{sq} \leq (I_k^{max})^2 + i_{k(\cdot)}^{viol,+} \quad (37)$$

where the slack variables $p_n^{viol,+}$, $q_n^{viol,+}$, $v_n^{viol,-}$, $v_n^{viol,+}$, and $i_{k(\cdot)}^{viol,+}$ are penalized in the objective function.

4) *Constraint Reduction*: We apply the concept of flowgate monitors to solve the linearized formulation with a reduced constraint set. We compute and monitor the flows for a subset of lines $k' \in \mathcal{K}'^{(h)} \subset \mathcal{K}$ where $k'(\cdot) \in \mathcal{F}$ are near or at $(I_k^{max})^2$. The subset of lines $\mathcal{K}'^{(h)} \subset \mathcal{K}$ is updated at each iteration $h > 0$. The constraint set is therefore reduced to only include (27), (32), (33), and (37) for all $k(\cdot) = k'(\cdot) \in \mathcal{F}$.

D. LP Subproblem Formulation

For each iteration h , the SLP IV-ACOPF solves the following LP subproblem:

$$\text{cost}(\cdot)^{(h)} = \min \left(\text{bids}(\cdot)^{(h)} + \text{penalty}(\cdot)^{(h)} \right) \quad (38)$$

subject to

$$(5) - (10), (22) - (37), \text{Appendix VII-B line (4)} \quad (39)$$

where $\text{bids}(\cdot)^{(h)}$ is defined in (21) and $\text{penalty}(\cdot)^{(h)}$ is defined as

$$\begin{aligned} \text{penalty}(\cdot)^{(h)} = & \sum_{n \in \mathcal{N}} \left[P_n^\epsilon (p_n^{viol,-} + p_n^{viol,+}) \right. \\ & \left. + Q_n^\epsilon (q_n^{viol,-} + q_n^{viol,+}) + V_n^\epsilon (v_n^{viol,-} + v_n^{viol,+}) \right] \\ & + \sum_{k' \in \mathcal{K}'^{(h)}} I_{k'}^\epsilon \left(i_{k'(n,m)}^{viol,+} + i_{k'(m,n)}^{viol,+} \right). \end{aligned} \quad (40)$$

The tangential cutting planes in line (4) of the process in Appendix VII-B are included in order to modify any ACOPF infeasibilities from the prior solution, for iterations $h > 0$.

V. RESULTS

We now test the computational performance and convergence quality of our SLP IV-ACOPF algorithm compared to a widely used interior point algorithm for solving a modified version of the nonlinear ACOPF described in Section IV-B.

TABLE I: Parameter defaults. Note that the flat start requires a slower rate a in step-size decrease due to poor initial start quality.

Parameter	Description	Value
P_n^ϵ	Real Power Penalty	$2.5 \cdot \max_n C_n^{g,1}$
Q_n^ϵ	Reactive Power Penalty	$12.5 \cdot \max_n C_n^{g,1}$
V_n^ϵ	Voltage Penalty	$15 \cdot \max_n C_n^{g,1}$
$I_{k'}^\epsilon$	Line Current Penalty	$25 \cdot \max_n C_n^{g,1}$
$r^2 (I_k^{max})^2$	Flowgate Monitor Rate	$r = 0.9$
I_k^{max}	Thermal Line Limits	$S_{k(\cdot)}^{max*} / \min_{i=n, m \in k(\cdot)} V_i^{max}$
$ \mathcal{L} $	Piecewise Segments	10
LIM	Iteration Limit	20
a	Step-Size Parameter	0.25 (0.1 for flat start)
b	Step-Size Parameter	1.5
Δ^{P-tol}	Mismatch Tolerance	$50e-4$
Δ_n^{P-tol}	Mismatch Tolerance	$10e-4$
Δ^{Q-tol}	Mismatch Tolerance	$50e-3$
Δ_n^{Q-tol}	Mismatch Tolerance	$50e-4$

The modified version includes penalty factors on inequality constraints and the piecewise linear objective function given by (38). Both approaches were implemented in Python 2.7 with Pyomo 3.5 [17] and executed on a workstation with four quad-core Intel Xeon 2.7 GHz processors with hyper-threading and 512GB RAM. We solve the LP subproblems of the SLP IV-ACOPF with either Gurobi 5.6.2 [18] or CPLEX 12.5.1 [19] limited to two threads, and the nonlinear ACOPF with IPOPT 3.11.4 configured with the MA27 linear sub-solver (no multi-threading support) [20]. We compare the two approaches on various IEEE and Polish networks [7]. For each network, we consider a baseline case and a thermally constrained case. The thermal line limits are systematically computed as described in Lipka et. al. [21] from the optimal solution for the apparent power flow $S_{k(\cdot)}^{max*}$ using MATPOWER [7]; the defaults for this and other parameters are specified in Table I.

We consider four types of initializations: (1) flat start, (2) DC warm start, (3) AC warm start, and (4) uniform start. The flat start assumes unit voltage and half-max output for all generation. The DC and AC warm starts are constructed from DCOPF and ACOPF optimal solutions, respectively, where the demand is parameterized as $P_n^d \sim \mathcal{U}(0.9P_n^d, 1.1P_n^d)$ for all $n \in \mathcal{N}$. The uniform starts assume that $v_n^r \sim \mathcal{U}(V_n^{min}, V_n^{max})$ and $v_n^j = 0$ for all $n \in \mathcal{N}$. The sample size for the various initialization types is one for the flat start and 10 for the remaining start types. To reduce variance in the comparison, we use the same starting points to test both the nonlinear ACOPF and the SLP IV-ACOPF approaches.

In general, CPLEX converged to an optimal solution in 98.4% of the runs, Gurobi in 99.1% of the runs, and IPOPT in 100% of the runs for both baseline and thermally constrained networks. In Table II, we report the fastest recorded solver CPU time for each test configuration. The solution times for the SLP IV-ACOPF are reported as a multiplier of a reference CPU time, taken as the time required by IPOPT to solve the nonlinear ACOPF. To analyze algorithm scaling properties, we fit regression models relating network size to run time. Empirically, CPLEX scaling is approximately linear, i.e. $O(n^{0.98})$ for the baseline and $O(n^{1.01})$ for the thermally constrained case. Gurobi empirical scaling is slightly worse, with $O(n^{1.02})$ for the baseline and $O(n^{1.06})$ for the

TABLE IV: The network power factors for the nonlinear ACOPF that correspond to the fastest recorded run; the analogous SLP IV-ACOPF power factors are within two significant digits of these baseline values.

Network	Baseline	Thermal Line Limits
IEEE-14	0.97	0.97
IEEE-30	0.89	0.89
IEEE-57	0.98	0.98
IEEE-118	1.00	1.00
IEEE-300	0.96	0.96
Polish-2,383	0.96	0.96
Polish-2,737	1.00	1.00
Polish-2,746	0.99	0.99
Polish-3,012	0.96	0.96
Polish-3,120	0.97	0.97
Polish-3,375	0.98	0.98

thermally constrained case. The IPOPT solver performance scales more poorly than either SLP IV-ACOPF configuration, with $O(n^{1.13})$ for the baseline and $O(n^{1.14})$ for the thermally constrained case. When configured with CPLEX, the SLP IV-ACOPF is faster than IPOPT solving the nonlinear ACOPF on the largest network (Polish-3,375). In general, SLP IV-ACOPF scales linearly even when binding thermal line limits are present.

In Table III we report the aggregate bid production costs, i.e., bids $(\cdot)^{(h)}$ in Equation (38), that correspond to runs with the fastest recorded solver CPU time. The results indicate that penalty costs are present in all solutions to the Polish networks obtained using IPOPT on the nonlinear ACOPF formulation. Furthermore, penalty costs are present for both approaches when solving the thermally constrained IEEE-118 and Polish-2,383 networks, and for the SLP IV-ACOPF when solving the thermally constrained IEEE-30 network. The SLP IV-ACOPF can yield lower bid production costs than IPOPT, particularly on runs where penalty costs are present in the nonlinear ACOPF solution. As we report in Table IV, the power factors for solutions obtained using both approaches are identical to two significant digits. A unity power factor indicates that less current is required to supply the power demanded. Therefore, power factors near 1.0 are desirable.

In Figures 4a and 4b, we report the fastest recorded solve times across the four initialization types for CPLEX and Gurobi configurations of SLP IV-ACOPF, respectively. The figures aggregate both the baseline and thermally constrained network cases. The flat start does not perform competitively for this formulation; with the flat start ($v_n^r = 1$ and $v_n^j = 0$ for all $n \in \mathcal{N}$), the current flows are initialized to zero across symmetric transmission elements, and the Taylor series approximations in Equations (24) – (26) are such that the entire first order expansion is not assessed for the initial iteration $h = 0$. However, the uniform starts perform competitively compared to the DC and AC warm starts. The uniform starts do not require any knowledge of the prior operating state; by initializing v_n^r with some variation, the subsequent current flows become nonzero. Figures 4a and 4b illustrate that for many of the cases tested, the SLP IV-ACOPF acquires the fastest recorded solver CPU time with the uniform start. More comprehensive results can be found in our online addendum [22].

TABLE II: The fastest recorded solver CPU time across all simulations for both baseline and thermally constrained networks.

Network	Baseline				With Thermal Line Limits			
	Solver CPU Time (s)		Multiplier (\times)		Solver CPU Time (s)		Multiplier (\times)	
	ACOPF	IPOPT	SLP IV-ACOPF	Gurobi	IV-ACOPF	IPOPT	CPLEX	Gurobi
IEEE-14	0.10		3.40	2.50	0.11		1.73	1.55
IEEE-30	0.15		1.40	5.93	0.14		3.86	4.79
IEEE-57	0.31		2.65	3.06	0.29		2.72	3.55
IEEE-118	0.94		2.33	3.71	0.83		2.99	4.52
IEEE-300	2.19		2.14	4.21	1.82		2.73	5.79
Polish-2,383	25.59		1.13	2.68	20.72		1.75	3.10
Polish-2,737	24.28		1.39	2.41	24.04		1.70	2.53
Polish-2,746	22.48		1.70	3.14	24.50		1.72	2.81
Polish-3,012	38.44		1.14	1.77	32.23		1.46	2.07
Polish-3,120	27.10		1.68	2.60	37.73		1.42	1.93
Polish-3,375	68.83		0.70	1.22	65.24		0.99	1.30

TABLE III: The aggregate bid production costs that correspond to the fastest recorded run, as presented in Table II. Entries in **bold** indicate the presence of penalty costs in the obtained solutions. Note that there are no active thermal line limits for IEEE-300 runs.

Network	Baseline				With Thermal Line Limits			
	Bid Production Cost (\$)		Relative Change		Bid Production Cost (\$)		Relative Change	
	ACOPF	IPOPT	SLP IV-ACOPF	Gurobi	ACOPF	IPOPT	CPLEX	Gurobi
IEEE-14	8,091		1.2E-04	1.2E-04	9,294		-3.2E-04	-3.2E-04
IEEE-30	575		0.0E+00	0.0E+00	582		1.7E-02	2.6E-02
IEEE-57	41,817		5.3E-04	1.7E-04	41,978		2.4E-04	-4.8E-05
IEEE-118	129,903		1.2E-03	1.2E-03	136,575		1.8E-03	1.8E-03
IEEE-300	720,149		3.6E-05	1.9E-05	720,149		1.8E-05	6.9E-05
Polish-2,383	1,858,445		2.6E-04	4.8E-04	1,882,042		-1.2E-02	-1.2E-02
Polish-2,737	742,678		1.1E-04	1.2E-04	742,687		9.3E-05	9.7E-05
Polish-2,746	1,185,113		1.8E-04	1.8E-04	1,185,505		2.4E-04	2.4E-04
Polish-3,012	2,581,018		1.4E-04	3.7E-04	2,598,020		1.5E-04	1.9E-04
Polish-3,120	2,137,307		2.0E-04	1.4E-04	2,142,922		-6.8E-04	-7.1E-04
Polish-3,375	7,402,883		1.2E-03	1.4E-03	7,415,409		1.2E-03	1.4E-03

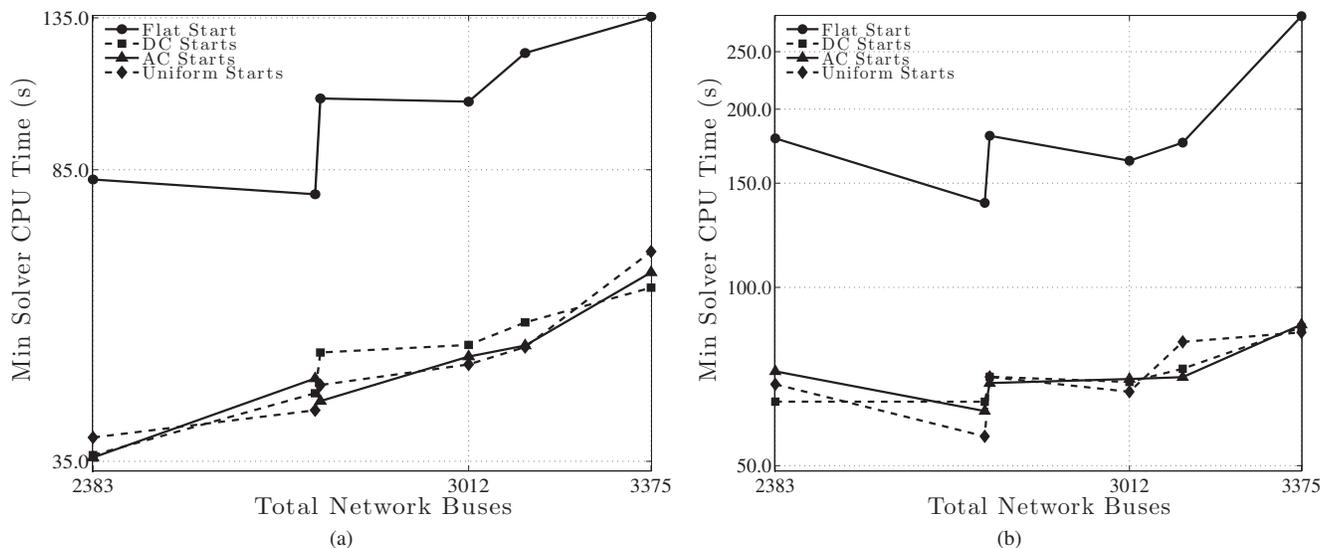


Fig. 4: The fastest recorded CPU time for CPLEX in (a) and Gurobi in (b) on each initialization type in both the baseline and thermally constrained Polish networks.

VI. DISCUSSION

We propose a linearization of the ACOPF, the SLP IV-ACOPF, that can be extended and integrated into practical applications. The algorithm can leverage commercial LP solvers such as Gurobi and CPLEX, and can be implemented on any combination of software and hardware platforms that support optimization in linear programming. The SLP IV-

ACOPF demonstrates a balance between accuracy and efficiency. Whereas the nonlinear ACOPF scales in polynomial time, the solver CPU time for the SLP IV-ACOPF demonstrates linear time complexity on the IEEE and Polish networks, without and with network congestion. Thus SLP IV-ACOPF demonstrates robust computational performance even in the thermally constrained networks. The algorithm does

not require a warm start to perform competitively with the nonlinear ACOFP.

VII. APPENDIX: SUPPORTING DETAILS

A. Step-Size Limits

At the end of each iteration $h > 0$, we modify the tunable parameter $V_n^{(h)}$ to control the approximation error. As the approximation nears ACOFP feasibility, we accelerate the convergence rate; if the approximation worsens, we decelerate this rate through calculating the ratio

$$\gamma^{(h)} = \frac{\text{penalty}(\cdot)^{(h)} + f(\cdot)^{(h)}}{\text{cost}(\cdot)^{(h)} + f(\cdot)^{(h)}} \quad (\text{A.41})$$

$$\text{where } f(\cdot)^{(h)} = \sum_{n \in \mathcal{N}} \text{MISMATCH}_n^\epsilon |P_n^* - p_n^g| \quad (\text{A.42})$$

$$\text{and } P_n^* = P_n^d + v_n^{r*} i_n^{r*} + v_n^{j*} i_n^{j*}. \quad (\text{A.43})$$

As $\gamma^{(h)} \rightarrow 0$, the solution becomes ACOFP feasible, i.e. $f(\cdot)^{(h)} = 0$ and $\text{penalty}(\cdot)^{(h)} = 0$. We calculate tunable parameters $\beta = -a \log \gamma^{(h)} + b$ and $\alpha = 1/\beta$ and the allowable stepsize as

$$V_n^{(h)} \leftarrow \alpha |V_n^{\max}| / h^\beta. \quad (\text{A.44})$$

For faster decay, the user can increase a . When $\gamma^{(h)} = 1$, then $\beta = b$.

B. Infeasibility Handling

The following routine determines the updated evaluation point and resulting tangential cutting plane when the voltage magnitude upper bound in (17) or the thermal line limit in (18) is violated by the optimal solution of the LP subproblem in iteration h . Without loss of generality, we denote the real part as x^r , the imaginary part as x^j , and the upper bound as X^{\max} .

```

1: if  $(\hat{x}^r(h))^2 + (\hat{x}^j(h))^2 > (X^{\max})^2$  then
2:    $\hat{x}^r(h) \leftarrow (\hat{x}^r(h)) \sqrt{(X^{\max})^2 / ((\hat{x}^r(h))^2 + (\hat{x}^j(h))^2)}$ 
3:    $\hat{x}^j(h) \leftarrow (\hat{x}^j(h)) \sqrt{(X^{\max})^2 / ((\hat{x}^r(h))^2 + (\hat{x}^j(h))^2)}$ 
4:   add constraint:
       $\hat{x}^r(h) x^r + \hat{x}^j(h) x^j \leq (X^{\max})^2 + x^{\text{viol},+}$ 
5: end if

```

C. Stopping Criteria

The stopping criteria assesses three possible scenarios: (1) the mismatch error on real and reactive power injections for all buses $n \in \mathcal{N}$ is less than a specified tolerance, (2) the net of these mismatches is less than a specified tolerance, or (3) the maximum iteration limit has been reached.

```

1:  $P_n^* = P_n^d + v_n^{r*} i_n^{r*} + v_n^{j*} i_n^{j*}$ 
2:  $Q_n^* = Q_n^d + v_n^{j*} i_n^{r*} - v_n^{r*} i_n^{j*}$ 
3: for all  $n \in \mathcal{N}$  do
4:    $\delta_n^p \leftarrow |P_n^* - p_n^g| / \min(|P_n^*|, |p_n^g|)$ 
5:    $\delta_n^q \leftarrow |Q_n^* - q_n^g| / \min(|Q_n^*|, |q_n^g|)$ 
6: end for

```

```

7: if  $\left( \begin{array}{l} \max_{n \in \mathcal{N}} \delta_n^p \leq \Delta^{P-tol} \text{ and } \max_{n \in \mathcal{N}} \delta_n^q \leq \Delta^{Q-tol} \\ \text{or } \sum_{n \in \mathcal{N}} \delta_n^p \leq \Delta^{P-tol} \text{ and } \sum_{n \in \mathcal{N}} \delta_n^q \leq \Delta^{Q-tol} \\ \text{or } h \geq \text{LIM} \end{array} \right)$ 
   then
8:   return solution;
9: end if

```

ACKNOWLEDGMENT

We acknowledge the contributions of our co-authors on the FERC ‘‘Optimal Power Flow and Formulation Papers.’’

REFERENCES

- [1] J. Carpentier, ‘‘Contribution a l’etude du dispatching economique,’’ *Bull. Soc. Francaise Elect.*, vol. 3, pp. 431–447, 1962.
- [2] M. B. Cain, R. P. O’Neill, and A. Castillo, ‘‘Optimal power flow papers: Paper 1. history of optimal power flow and formulations,’’ Federal Energy Regulatory Commission, Tech. Rep., 2013.
- [3] A. Castillo and R. P. O’Neill, ‘‘Survey of approaches to solving the ACOFP,’’ Federal Energy Regulatory Commission, Tech. Rep., 2013.
- [4] S. Frank, I. Steponavice, and S. Rebennack, ‘‘Optimal power flow: a bibliographic survey 1,’’ *Energy Systems*, vol. 3, no. 3, pp. 221–258, 2012.
- [5] —, ‘‘Optimal power flow: a bibliographic survey II,’’ *Energy Systems*, vol. 3, no. 3, pp. 259–289, 2012.
- [6] H. Zhang, V. Vittal, G. T. Heydt, and J. Quintero, ‘‘A relaxed optimal power flow model based on a taylor series,’’ in *2013 IEEE Innovative Smart Grid Technologies - Asia*, 2013, pp. 1–5.
- [7] 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 Trans. on Power Sys.*, vol. 26, no. 1, pp. 12–19, 2011.
- [8] C. A. N. A. Borghetti and M. Paolone, ‘‘A mixed integer linear programming approach to the optimal configuration of electrical distribution networks with embedded generators,’’ in *Proc of the 17th PSCC*, 2011.
- [9] R. Ferreira, C. Borges, and M. Pereira, ‘‘Distribution network reconfiguration under modeling of AC optimal power flow equations: A mixed-integer programming approach,’’ in *2013 IEEE PES Conference On Innovative Smart Grid Technologies Latin America*, 2013, pp. 1–8.
- [10] J. Franco, M. J. Rider, M. Lavorato, and R. Romero, ‘‘A set of linear equations to calculate the steady-state operation of an electrical distribution system,’’ in *2011 IEEE PES Conference on Innovative Smart Grid Technologies (ISGT Latin America)*, 2011, pp. 1–5.
- [11] A. Mohapatra, P. Bijwe, and B. Panigrahi, ‘‘Efficient sequential non-linear optimal power flow approach using incremental variables,’’ *IET Generation, Trans. & Distr.*, vol. 7, no. 12, pp. 1473–1480, 2013.
- [12] C. Coffrin, P. V. Hentenryck, and R. Bent, ‘‘Approximating line losses and apparent power in AC power flow linearizations,’’ in *2012 IEEE Power and Energy Society General Meeting*, 2012.
- [13] C. Coffrin and P. V. Hentenryck, ‘‘A linear-programming approximation of AC power flows,’’ University of Melbourne, Tech. Rep. arXiv:1206.3614v3, 2013.
- [14] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, ‘‘Semidefinite programming for optimal power flow problems,’’ *Intl Journal of Electrical Power and Energy Systems*, vol. 30, no. 6–7, pp. 383–392, 2008.
- [15] J. Lavaei and S. H. Low, ‘‘Zero duality gap in optimal power flow problem,’’ *IEEE Trans. on Power Sys.*, vol. 27, no. 1, pp. 92–107, 2012.
- [16] B. C. Lesieutre, D. K. Molzahn, A. R. Borden, and C. L. DeMarco, ‘‘Examining the limits of the application of semidefinite programming to power flow problems,’’ in *Proc. of the 49th Annual Allerton Conference on Communication, Control and Computing*, 2011.
- [17] W. E. Hart, C. Laird, J.-P. Watson, and D. L. Woodruff, *Pyomo - Optimization Modeling in Python*. Springer Publishing Company, 2012.
- [18] Gurobi Optimization, Inc., ‘‘Gurobi documentation: Release 5.6,’’ 2014.
- [19] IBM Corp., ‘‘CPLEX documentation: Release 12.4,’’ 2014.
- [20] A. Wachter and L. T. Biegler, ‘‘On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming,’’ *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.
- [21] P. Lipka, R. O’Neill, and S. Oren, ‘‘Developing line current magnitude constraints for IEEE test problems,’’ Federal Energy Regulatory Commission, Tech. Rep., 2013.
- [22] A. Castillo, P. Lipka, J.-P. Watson, S. Oren, and R. P. O’Neill, ‘‘The current-voltage (IV) successive linear program of the ACOFP: Addendum,’’ Federal Energy Regulatory Commission, Tech. Rep., 2014, available upon request.