This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2016.2615688, IEEE Transactions on Power Systems

Alternative LP and SOCP hierarchies for ACOPF problems

Xiaolong Kuang, Bissan Ghaddar, Joe Naoum-Sawaya, and Luis F. Zuluaga

***Abstract*—The alternating current optimal power flow (ACOPF) problem optimizes the generation and the distribution of electric energy taking into account the active and the reactive power generation limits, demand requirements, bus voltage limits, and network flow limits. The ACOPF problem can be formulated as a non-convex polynomial program that is generally difficult to solve due to the non-linear power flow constraints. A recently proposed approach to globally solve the ACOPF problem is through the formulation of a hierarchy of semidefinite programs that are computationally challenging to solve for large-scale problems. In this paper we explore a solution approach that alleviates this computational burden by using hierarchies of linear and second order programs and by exploiting the network structure of the transmission grid. Furthermore, we show that the first level of the second order cone hierarchy is equivalent to solving the conic dual of the approximation that was recently proposed in the literature, which provides the optimal solution of the ACOPF problem for special network topologies.**

***Index Terms*—AC optimal power flow, polynomial programming, second-order cone programming**

I. INTRODUCTION

The Alternating Current Optimal Power Flow (ACOPF) problem is a challenging optimization problem in power systems. Several optimization methods have been devised to efficiently solve different variations of this problem. However, the ACOPF problem continues to be challenging. This is mainly due to the large-scale size of real systems and the nonlinearities and the nonconvexities in the underlying formulation.

One solution approach that has been actively investigated to address the global solution of the ACOPF relies on reformulating the problem as a polynomial program (PP); that is an optimization problem whose objective and constraints can be written in terms of polynomials on the decision variables. Then, a hierarchy of *semidefinite programming* (SDP) [3] relaxations can be used to obtain globally optimal solutions under mild conditions [7, 11, 18, 19]. This hierarchy of SDP relaxations is based on the seminal work of Lasserre [14], who showed that SDP relaxations based on *sum of square* (SOS) polynomials can provide global bounds for a general class of PPs. However, the associated SDP relaxations are computationally expensive and thus, even using the hierarchy's low-order relaxations to approximate large-scale PPs like the ACOPF becomes computationally intractable in practice [16].

To improve the computational performance of the SDP based hierarchies, prior work has focused on exploiting the problem structure through sparsity [13] and symmetry [6], improving the relaxation through the generation of valid inequalities [9], and more recently through devising more computationally efficient hierarchies based on the use of linear programming (LP) and second-order cone programming (SOCP) relaxations of the problem of interest. In particular, consider the work of [2, 8, 23] base on the use of polynomial programming techniques, as well as the work of [12] based on the SOCP approximations of the sinusoidal terms in the ACOPF formulation.

Rather than using the computationally expensive SDP hierarchy, in this paper we explore the use of LP and SOCP hierarchies for solving the ACOPF problem and show that the SOCP approach that is proposed in this paper can be used to obtain strong bounds for ACOPF within limited computational time. Furthermore, we show that the first-order SOCP hierarchy obtained by weakening the more common hierarchy of SDP relaxations for this problem is equivalent to solving the conic dual of the SOCP approximations of ACOPF that have been recently proposed in [4, 10, 25], which provide the optimal solution of the ACOPF problem for special network topologies. In turn, the SOCP hierarchy approach provides a natural hierarchy of increasingly stronger SOCP approximations for the ACOPF problem.

The remainder of the article is organized as follows. We briefly introduce PPs and discuss the construction of hierarchies of tractable *conic programming* [3] relaxations for them in Section II. The application of the proposed LP and SOCP based hierarchies to the ACOPF problem is discussed in Section III. Finally, Section IV provides concluding remarks and future research directions.

II. PRELIMINARIES

In this section, we first introduce the notation that will be used throughout the paper. We then present the general approach in [2, 14] for solving a *polynomial program* (PP) by solving a hierarchy of tractable conic programming relaxations of the problem [3]. Let $\mathbf{R}_d[x] := \mathbf{R}_d[x_1, \dots, x_n]$ be the set of polynomials in n variables with real coefficients of degree at most d. Also, given any $\mathcal{G} \subseteq \mathbf{R}_d[x]$, let $S_{\mathcal{G}} = \{x \in \mathbb{R}^n : g(x) \geq 0, \forall g \in \mathcal{G}\}$. We define

$$SOS_{2d} := \left\{ \sum_{i=1}^{k} p_i(x)^2 : p_i(x) \in \mathbf{R}_d[x], k \in \mathbb{Z}_+ \right\},$$

as the cone of sum of square polynomials in x of degree at most $2d$. For any $S \subseteq \mathbb{R}^n$, let $\mathcal{P}_d(S)$ be the cone of

---

X. Kuang and L. F. Zuluaga are with the Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, 18015, USA.

B. Ghaddar is with the Department of Management Sciences, University of Waterloo, Canada.

J. Naoum-Sawaya is with the Ivey Business School, University of Western Ontario, Canada.

polynomials of degree at most d that are non-negative over the set S. We consider the following general *primal* polynomial program,

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \geq 0, i = 1, \dots, m. \end{aligned} \tag{PP-P}$$

After letting $\mathcal{G} = \{g_i(x) : i = 1, \dots, m\}$, and using duality, problem (PP-P) can be equivalently rewritten as a conic program [9]:

$$\begin{aligned} \max \quad & \lambda \\ \text{s.t.} \quad & f(x) - \lambda \in \mathcal{P}_d(S_{\mathcal{G}}). \end{aligned} \tag{PP-D}$$

In general, solving problem (PP-P) is NP-hard [21]. Problem (PP-D) is a (linear) conic program whose complexity is captured in the cone $\mathcal{P}_d(S_{\mathcal{G}})$, which is not tractable in general.

Consider a tractable convex cone $\mathcal{M}_d(S_{\mathcal{G}}) \subseteq \mathcal{P}_d(S_{\mathcal{G}})$, then the following convex program

$$\begin{aligned} \max \quad & \lambda \\ \text{s.t.} \quad & f(x) - \lambda \in \mathcal{M}_d(S_{\mathcal{G}}) \end{aligned} \tag{PP-M}$$

will provide a lower bound for (PP-D), and hence a lower bound for (PP-P). The choice of the tractable cone $\mathcal{M}_d(S_{\mathcal{G}})$ is a key factor in obtaining good approximation bounds for (PP-D), and in turn (PP-P).

In his seminal work, Lasserre [14] proposed a hierarchy that builds up a sequence of SDP relaxations for (PP-D) by letting $\mathcal{M}_d(S_{\mathcal{G}}) = \mathcal{K}_{\mathcal{G}}^r$ in (PP-M), where

$$\begin{aligned} \mathcal{K}_{\mathcal{G}}^r = & \{p(x) \in \mathbf{R}_d[x] : p(x) = s_0(x) + \sum_{i=1}^m s_i(x) g_i(x), \\ & s_0(x) \in SOS_r, s_i(x) \in SOS_{r-\deg(g_i)}\}, \end{aligned}$$

and $r \geq d$. The corresponding optimization problem is

$$\begin{aligned} \max \quad & \lambda \\ \text{s.t.} \quad & f(x) - \lambda = s_0(x) + \sum_{i=1}^m s_i(x) g_i(x), \\ & s_0(x) \in SOS_r, s_i(x) \in SOS_{r-\deg(g_i)}. \end{aligned} \tag{PP-SOS$_r$}$$

The hierarchy of problems (PP-SOS$_r$) converges to the global solution of (PP-P) under some conditions related to the compactness of the set $S_{\mathcal{G}}$, thanks to the fact that in such cases $\mathcal{K}_{\mathcal{G}}^r \rightarrow \mathcal{P}_d(S_{\mathcal{G}})$ as $r \rightarrow \infty$ [14]. However, as r increases, the size of the positive semidefinite matrices required to reformulate (PP-SOS$_r$) as a SDP increases exponentially. As a result, this approach is computationally expensive for large-scale problems and even for small-scale problems that require high levels of the hierarchy to obtain good approximations for the optimal solution.

Ahmadi and Majumdar [2] recently proposed a restriction of the SOS condition to address the computational shortcoming of the SDP based hierarchies, which is done by introducing the use of *diagonally dominant SOS* (DSOS) polynomials and *scaled diagonally dominant SOS* (SDSOS) polynomials instead of SOS polynomials in (PP-SOS$_r$).

Definition 1 (DSOS polynomials [2])**.** *Let J be an index set, $m_i(x) \in \mathbf{R}_d[x]$ be a monomial in $x \in \mathbf{R}^n$ for all $i \in J$, and $\alpha_i, \beta_{ij} \in \mathbb{R}_+$ for all $i, j \in J$, then $m_i(x) = \sum_i \alpha_i m_i(x)^2 + \sum_{i,j} \beta_{ij}(m_i(x) \pm m_j(x))^2$ is a diagonally dominant sum of square (DSOS) polynomial in $\mathbf{R}_{2d}[x]$.*

Equivalently, DSOS polynomials can be defined as those that can be constructed from a *diagonally dominant (DD) matrix*. Namely, let $z(x)$ be a vector with the monomials $m_i(x)$ for all $i \in J$, and $Q \in \mathbb{R}^{|J| \times |J|}$ be a (symmetric) DD matrix, then $p(x) = z^T(x) Q z(x)$ is a DSOS polynomial.

Definition 2 (SDSOS polynomials [2])**.** *Let J be an index set, $m_i(x) \in \mathbf{R}_d[x]$ be a monomial in $x \in \mathbf{R}^n$ for all $i \in J$, and $\alpha_i, \beta_i, \beta_j \in \mathbb{R}_+$ for all $i, j \in J$, then $p(x) = \sum_i \alpha_i m_i(x)^2 + \sum_{i,j} (\beta_i m_i(x) \pm \beta_j m_j(x))^2$, is a scaled diagonally dominant sum of square (SDSOS) polynomial in $\mathbf{R}_{2d}[x]$.*

Equivalently, SDSOS polynomials can be defined as those that can be constructed from a *scaled diagonally dominant (SDD) matrix.* Let $z(x)$ be a vector with the monomials $m_i(x)$ for all $i \in J$, and $Q \in \mathbb{R}^{|J| \times |J|}$ be a (symmetric) SDD matrix, then $p(x) = z^T(x) Q z(x)$ is an SDSOS polynomial.

Similar to the definition of SOS_{2d}, let the sets of all DSOS and SDSOS polynomials with degree $2d$, be respectively denoted by $DSOS_{2d}$ and $SDSOS_{2d}$. Then, we have

$$DSOS_{2d} \subseteq SDSOS_{2d} \subseteq SOS_{2d}. \tag{1}$$

By replacing the SOS cones in (PP-SOS$_r$), the associated DSOS and SDSOS hierarchies are:

$$\begin{aligned} \max_{\lambda, d_i(x)} \quad & \lambda \\ \text{s.t.} \quad & f(x) - \lambda = d_0(x) + \sum_{i=1}^m d_i(x) g_i(x), \\ & d_0(x) \in DSOS_r, \\ & d_i(x) \in DSOS_{r-\deg(g_i)}, \end{aligned} \tag{PP-DSOS$_r$}$$

and

$$\begin{aligned} \max_{\lambda, d_i(x)} \quad & \lambda \\ \text{s.t.} \quad & f(x) - \lambda = d_0(x) + \sum_{i=1}^m d_i(x) g_i(x), \\ & d_0(x) \in SDSOS_r, \\ & d_i(x) \in SDSOS_{r-\deg(g_i)}. \end{aligned} \tag{PP-SDSOS$_r$}$$

Thus due to (1), using DSOS and SDSOS polynomials instead of SOS polynomials in (PP-SOS$_r$) provides a hierarchy of lower bounds for the SOS hierarchy (PP-SOS$_r$) by restricting the set of SOS polynomials to those that are DSOS and SDSOS. Moreover, the resulting hierarchies (PP-DSOS$_r$) and (PP-SDSOS$_r$) are computationally easier to solve than (PP-SOS$_r$) since they reduce to solving LPs and SOCPs respectively. Particularly, (PP-DSOS$_r$) reduces to an LP since DSOS polynomials can be constructed from DD matrices, and a symmetric matrix $A \in \mathbb{R}^{n \times n}$ is DD if $A_{ii} \geq \sum_{j \neq i} |A_{ij}|, \forall i = 1, \dots, n$ which can be formulated using linear constraints. Similarly, SDSOS hierarchy reduces to an SOCP since SDSOS polynomials can be constructed from SDD matrices, and a symmetric matrix $A \in \mathbb{R}^{n \times n}$ is

SDD if $A = \sum_{i,j \in \{1,\dots,n\}} A^{ij}$ for some $A^{ij} \in \mathbb{R}^{n \times n}$, $A^{ij} \succeq 0$ with $A^{ij}_{kl} = 0$ for any $k, l \in \{1, \dots, n\}$ such that $(k, l) \notin \{i, j\} \times \{i, j\}$. Futhermore, the condition $A^{ij} \succeq 0$ is equivalent to

$$\begin{pmatrix} A^{ij}_{ii} & A^{ij}_{ij} \\ A^{ij}_{ji} & A^{ij}_{jj} \end{pmatrix} \succeq 0 \tag{2}$$

which is enforced using the second order cone constraint

$$\frac{A^{ij}_{ii} + A^{ij}_{jj}}{2} \geq \left\| \begin{pmatrix} A^{ij}_{ij} \\ \frac{A^{ij}_{ii} - A^{ij}_{jj}}{2} \end{pmatrix} \right\|_2 . \tag{3}$$

Table II.1 summarizes a comparison in terms of number of variables and conic constraints in the first level of the (PP-SOS$_r$), (PP-SDSOS$_r$), and (PP-DSOS$_r$) hierarchies for a *quadratic* PP (i.e., when all the polynomials defining the PP are quadratic). One can see that the number of variables times the number of conic constraints for all of them is of order $O(n^2)$; however, the SDP relaxations (PP-SOS$_r$) are typically harder to solve than the SOCP relaxations (PP-SDSOS$_r$), or the LP relaxations (PP-DSOS$_r$). On the other hand, whether the SOCP relaxations are harder to solve than the LP relaxations (PP-DSOS$_r$), depends on the size of the problem, thanks to the recent advances in the solution of SOCP problems [see, e.g., 20]. This is illustrated later in the numerical experiments presented in Table III.2.

TABLE II.1

COMPARISON OF NUMBER OF CONIC CONSTRAINTS (CC) IN FIRST LEVEL OF THE (PP-SOS$_r$),(PP-SDSOS$_r$), AND (PP-DSOS$_r$) CONIC RELAXATIONS, FOR QUADRATIC PPS WITH n VARIABLES

	Type	Number of CC	Number of variables in CC
(PP-SOS$_2$)	SDP	1	$(n+1) \times (n+1)$
(PP-SDSOS$_2$)	SOCP	$(n+1)n/2$	3
(PP-DSOS$_2$)	LP	$n+1$	n
	LP	n^2+n	2

Ahmadi and Majumdar [2] have shown that the approximation hierarchies (PP-DSOS$_r$) and (PP-SDSOS$_r$) can be successfully used to approximate PPs arising in control, combinatorics, and general non-linear non-convex optimization, in lieu of using the stronger (PP-SOS$_r$) hierarchy. In the next section, we use this approach to address the solution of the ACOPF problem.

III. ACOPF FORMULATION

In this section, we apply the LP and SOCP hierarchies (PP-DSOS$_r$) and (PP-SDSOS$_r$) to the alternating current optimal flow problem and we exploit the network structure of electricity transmission grids to specialize the proposed SOCP hierarchy (PP-SDSOS$_r$) for the ACOPF problem and obtain improving results. The results are then compared to the SDP based hierarchy [14] and to the SOCP relaxation that is proposed in [4].

A. ACOPF problem as a Polynomial Program

To formulate the ACOPF problem as a polynomial problem, we follow the same notation as in [17, 19], that is, we consider an undirected graph $P(N, E)$ where each vertex $n \in N$ is called a “bus” and each edge $e \in E$ is called a “branch”. The subset $G \subseteq N$ denotes the set of generators. Additionally, we define the following parameters:

- P^-_k, P^+_k, Q^-_k, Q^+_k, V^-_k and V^+_k are respectively the limits on active and reactive generation capacity and the absolute value of the voltage at bus k.
- S^+_{lm} is the limit on the absolute value of the apparent power of a branch $(l, m) \in E$.
- P^d_k and Q^d_k are the active and reactive power demand respectively.
- c^2_k, c^1_k, c^0_k are nonnegative coefficients for the power generation cost function.

We also define the following decision variables:

- P^g_k and Q^g_k: active and reactive power generated at bus k.
- P_{lm} and Q_{lm}: active and reactive power flow on branch (l, m).

Given a complex voltage V_i at bus i, let $\Re V_i$ denote the real part of V_i and $\Im V_i$ denote the imaginary part. The power flow equations are

$$\begin{aligned} P^g_k &= P^d_k + \Re V_k \sum_{i=1}^{n} (\Re y_{ik} \Re V_i - \Im y_{ik} \Im V_i) \\ &\quad + \Im V_k \sum_{i=1}^{n} (\Im y_{ik} \Re V_i - \Re y_{ik} \Im V_i), \\ Q^g_k &= Q^d_k + \Re V_k \sum_{i=1}^{n} (-\Im y_{ik} \Re V_i - \Re y_{ik} \Im V_i) \\ &\quad + \Im V_k \sum_{i=1}^{n} (\Re y_{ik} \Re V_i - \Im y_{ik} \Im V_i), \\ P_{lm} &= b_{lm}(\Re V_l \Im V_m - \Re V_m \Im V_l) + g_{lm}(\Re V_l{}^2 + \Im V_m{}^2 \\ &\quad - \Im V_l \Im V_m - \Re V_l \Re V_m), \\ Q_{lm} &= b_{lm}(\Re V_l \Im V_m - \Im V_l \Im V_m - \Re V_l{}^2 - \Im V_l{}^2) \\ &\quad + g_{lm}(\Re V_l \Im V_m - \Re V_m \Im V_l - \Re V_m \Im V_l) \\ &\quad - \frac{\bar{b}_{lm}}{2}(\Re V_l{}^2 + \Im V_l{}^2). \end{aligned}$$

Additionally, the network admittance matrix and other related matrices are defined as follows:

$$y_k = e_k e_k^T y,$$

$$y_{lm} = (j\frac{\bar{b}_{lm}}{2} + g_{lm} + jb_{lm}) e_l e_l^T - (g_{lm} + jb_{lm}) e_l e_m^T,$$

$$Y_k = \frac{1}{2} \begin{bmatrix} \Re(y_k + y_k^T) & \Im(y_k^T - y_k) \\ \Im(y_k - y_k^T)) & \Re(y_k + y_k^T) \end{bmatrix},$$

$$\bar{Y}_k = -\frac{1}{2} \begin{bmatrix} \Im(y_k + y_k^T) & \Re(y_k - y_k^T) \\ \Re(y_k^T - y_k) & \Im(y_k + y_k^T) \end{bmatrix},$$

$$M_k = \begin{bmatrix} e_k e_k^T & 0 \\ 0 & e_k e_k^T \end{bmatrix},$$

$$Y_{lm} = \frac{1}{2}\begin{bmatrix} \Re(y_{lm} + y_{lm}^T) & \Im(y_{lm}^T - y_{lm}) \\ \Im(y_{lm} - y_{lm}^T) & \Re(y_{lm} + y_{lm}^T) \end{bmatrix},$$

$$\bar{Y}_{lm} = -\frac{1}{2}\begin{bmatrix} \Im(y_{lm} + y_{lm}^T) & \Re(y_{lm}^T - y_{lm}) \\ \Re(y_{lm}^T - y_{lm}) & \Im(y_{lm} + y_{lm}^T) \end{bmatrix}.$$

where e_k is the k^{th} standard basis vector in $\mathbb{R}^{|N|}$. Let $x := [\Re V_k \quad \Im V_k]^T$ be the vector of variables in addition to the variables P_k^g, P_{lm} and Q_{lm}. The ACOPF can be formulated as the following degree-2 PP:

$$\begin{aligned} \min \quad & \sum_{k \in G} \left(c_k^2 (P_k^g)^2 + c_k^1 (P_k^d + \text{tr}(Y_k x x^T)) + c_k^0 \right) && (4) \\ \text{s.t.} \quad & P_k^- \le \text{tr}(Y_k x x^T) + P_k^d \le P_k^+, \ \forall k \in G, && (5) \\ & Q_k^- \le \text{tr}(\bar{Y}_k x x^T) + Q_k^d \le Q_k^+, \ \forall k \in G, && (6) \\ & (V_k^-)^2 \le \text{tr}(M_k x x^T) \le (V_k^+)^2, \ \forall k \in N, && (7) \\ & P_{lm}^2 + Q_{lm}^2 \le (S_{lm}^+)^2, \ \forall (l,m) \in E, && (8) \\ & P_k^g = \text{tr}(Y_k x x^T) + P_k^d, \ \forall k \in G, && (9) \\ & P_{lm} = \text{tr}(Y_{lm} x x^T), \ \forall (l,m) \in E, && (10) \\ & Q_{lm} = \text{tr}(\bar{Y}_{lm} x x^T), \ \forall (l,m) \in E. && (11) \end{aligned}$$

(PP-OPF)

The objective function (4) minimizes the cost of power generation. Constraints (5), (6), and (7) set limits on the active power, reactive power, and voltage on each bus. Constraints (8) set a limit on the apparent power flow at every branch. Constraints (9) define the power generated while constraints (10), (11) define the active and reactive power flow. Problem (PP-OPF) is a PP of degree two and the proposed hierarchies are applicable to this problem.

B. Structured SOCP hierarchy for ACOPF

Problem (PP-OPF) is a quadratic program where hierarchy (PP-SDSOS$_r$) can be directly applied, which is an SOCP program. Moreover, the ACOPF problem exhibits a structure that can be exploited to improve the computational performance of the solution approach discussed here.

Proposition 1. *Let $\mathcal{X}_{ori}$ denote the set of x variables in* (PP-OPF) *and let $\mathcal{X}_{ext}$ denote the set of P_k^g, P_{lm}, and Q_{lm} variables. Then for $x \in \mathcal{X}_{ori}$ and $v \in \mathcal{X}_{ext}$, xv does not appear in the quadratic model* (PP-OPF).

Proposition 1 allows us to develop a specific hierarchy of (PP-SDSOS$_r$) that takes advantage of the structure of ACOPF problem. Note that such structure is also exploited in the related conic programming relaxations for the ACOPF problem studied in [4]. Additionally, some product terms of variables in $\mathcal{X}_{ori}$ do not appear based on the network structure.

Proposition 2. *For any $k \in N$, the term $tr(Y_k x x^T)$ in the objective function* (4) *is equivalent to*

$$tr(Y_k x x^T) = \alpha_k \Re V_k^2 + \beta_k \Im V_k^2 + \sum_{j:(k,j) \in E} (\gamma_1^{kj} \Re V_k \Re V_k + \gamma_2^{kj} \Re V_k \Im V_j + \gamma_3^{kj} \Im V_k \Re V_j + \gamma_4^{kj} \Im V_k \Im V_j), \forall k \in N,$$

where $\alpha_k, \beta_k, \gamma_1^{kj}, \gamma_2^{kj}, \gamma_3^{kj}, \gamma_4^{kj}$ are parameters.

Proof. Directly follows by exploring the structure of Y_k, $k \in N$ by removing product terms where $(i,j) \notin E$. ☐

Proposition 2 uses the branch information in the network to simplify (PP-SDSOS$_r$) by removing unnecessary monomial terms. Only the cross product of voltages of two connected generators appears in $\text{tr}(Y_k x x^T)$ and the same goes for $\text{tr}(\bar{Y}_k x x^T)$, which means that there is no need to generate all the monomials in the right hand side of hierarchy (PP-SDSOS$_r$). We denote "structured (PP-SDSOS$_r$)" as the improved SOCP hierarchy by exploring the network structure. Since electricity transmission grids are normally sparse graphs then the computational effort of solving structured (PP-SDSOS$_r$) hierarchy is reduced significantly thanks to the reduction in the number of SOCP constraints required in the hierarchy formulation.

C. Duality in SOCP Formulations for ACOPF

It has been shown in [7, Theorem 1] that the first level ($r = 2$) of the SDP hierarchy obtained by using the (PP-SOS$_r$) on (PP-OPF) is equivalent to the conic dual of the SDP relaxation for the ACOPF problem that is considered in [17, Optimization 3]. In this section, we show that the first level of the structured (PP-SDSOS$_r$) hierarchy (cf. Proposition 1) of the ACOPF problem is the conic dual problem of the SOCP relaxation of the ACOPF problem that is presented in [4]. Intuitively, this result follows from the fact that the conic dual of the SDD matrices (2) is the set of symmetric matrices $A \in \mathbb{R}^{n \times n}$ such that every 2×2 *principal submatrix* of A is positive semidefinite. Similar to (3), the latter condition can be formulated using the second-order cone constraint, which corresponds to the relaxation approach used in [4]. Below, we state this result formally.

Theorem 1. *The first level of the SOCP hierarchy structured* (PP-SDSOS$_2$) *on* (PP-OPF) *is equivalent to the conic dual of the SOCP relaxation for the ACOPF problem considered in [4].*

Proof. We first derive the Lagrangian dual problem of the SOCP relaxation for the ACOPF problem considered in [4], which is also an SOCP program due to the self-dual property of the SOCP cone. Then we compare it with the first level of the structured (PP-SDSOS$_2$) and conclude that they are equivalent. A detailed proof is provided in the Appendix. ☐

It is worth to mention that Theorem 1 is related to the results in [7], where it is shown that the first level of the SDP relaxations proposed in [16], when applied to the ACOPF problem, are equivalent to the SDP relaxations proposed in [17] for the ACOPF problem.

Furthermore, for this pair of dual SOCP problems, strong duality holds.

Theorem 2. *Strong duality holds between the first level of the SOCP hierarchy structured* (PP-SDSOS$_2$) *on* (PP-OPF) *and the SOCP relaxation for the ACOPF problem considered in [4].*

Proof. Strong duality holds for a conic program if there exists a strictly feasible solution for the corresponding primal or dual problems. The basic idea of the proof is to find a strictly feasible solution for the dual problem. A detailed proof is provided in the Appendix. □

Both Theorem 1 and Theorem 2 have worthy numerical implications. Specifically, Theorem 1 establishes the conic dual relationship between the existing SOCP relaxations in [4] and the first level of SOCP hierarchy (PP-SDSOS$_2$) for ACOPF. Thus, Theorem 1 implies that the ACOPF bounds obtained in [4] can be strengthened and potentially provide the optimal ACOPF objective value, while still using SOCP techniques, by increasing the level r of the SOCP hierarchy (PP-SDSOS$_r$).

Theorem 2 states that strong duality holds between these two relaxations, which indicates that solving these two relaxations provides the same bound on the ACOPF problems. More importantly, strong duality ensures that the *interior point methods* used to solve the corresponding SOCPs will (in theory) converge to their optimal solution [24].

D. Numerical Results on ACOPF Instances

In this section we apply the LP and the SOCP hierarchies (PP-DSOS$_r$) and (PP-SDSOS$_r$) to solve the PP formulation of the ACOPF problem (PP-OPF), and then we compare the results to the ones that are obtained using the SDP hierarchy based on Lasserre's approach [14], using the SDP relaxation introduced in [17]. The SDP relaxaitons (PP-SOS$_r$), (PP-DSOS$_r$), and (PP-DSOS$_r$), are implemented in `Matlab`. We use `SeDuMi` [26] as the SDP solver (to take advantage of the well-known precision of `SeDuMi` in solving SDPs) and `MOSEK` [20] as the LP and SOCP solver. The ACOPF instances are taken from [5, 27] and the results are summarized in Table III.1. In computing the gaps, the best known bounds for the ACOPF instances are taken from [7]. From Table III.1, it is clear that the SDP relaxation in [17] (3rd and 5th columns in Table III.1) provides strong bounds on ACOPF problems; however, the solution time (4th column in Table III.1) increases very rapidly as the size of the ACOPF instance increases. This is a result of both the complexity of solving SDPs and the rate at which the size of the SDP relaxations for larger ACOPF problems increase. The SDP relaxation (PP-SOS$_r$) (6th, 7th and 8th columns in Table III.1) provides solutions for instances up to 14 buses within 1 hour using the first level of the hierarchy. Instead, by using (PP-SDSOS$_r$) (9th, 10th, and 11th columns in Table III.1), we obtained lower bounds for instances of up to 300 buses using the first level of the hierarchy with an average gap of 2% from the best known bounds. Also, the results obtained using the LP relaxation (PP-DSOS$_2$) (12th, 13th, and 14th columns in Table III.1), illustrate that unlike the DSOS relaxation of the SDP constraints in the (PP-SOS$_r$) hierarchy, the SDSOS relaxation of the SDP constraints in the (PP-SOS$_r$) hierarchy captures most of the strength of the (PP-SOS$_r$) hierarchy at a much lower solution time for instances of the ACOPF problem. The fact that the LP relaxations (PP-DSOS$_2$) provide weak bounds is in line with other studies made on convex relaxations for general PP problem based on LP techniques [see, e.g., 15, 22].

Furthermore, the solution times of the (PP-SDSOS$_2$) relaxation can be substantially reduced when the structure of the ACOPF problem is exploited as described in Table III.2 (i.e., compare columns 6th and 7th with columns 9th and 10th in Table III.2). Also, Table III.2 illustrates Theorem 1; that is, it shows that the ACOPF bounds obtained with the structured SOCP approach (cf., Proposition 1), and the SOCP relaxation of the ACOPF problem R_2 presented in [4] are equivalent up to numerical precision (i.e., compare columns 3rd and 4th with columns 9th and 10th in Table III.2). The fact that the structured SOCP relaxation and the R_2 relaxation produce equivalent results when using an interior point method to solve them also illustrates the strong duality result presented in Theorem 2. Finally, columns 12th, 13th, and 14th in Table III.2 also illustrate another consequence of Theorem 1; namely, that the SOCP relaxation R_2 presented in [4] can be strengthened and potentially provide the optimal ACOPF objective value, while still using SOCP techniques, by increasing the level r of the SOCP hierarchy (PP-SDSOS$_r$).

IV. CONCLUDING REMARKS

In this paper, we applied the recently proposed LP and SOCP approximation hierarchies for polynomial programs proposed in [2] to the ACOPF problem. Numerical results on ACOPF instances from the literature show that the use of these hierarchies, together with the sparsity structure of the problem, allow the computation of global bounds for large-scale ACOPF problems where the SDP hierarchy for PP problems proposed in [14] fail to provide such bounds. Moreover, the first level of the SOCP hierarchy is shown to be equivalent to the dual of the SOCP relaxations proposed in [4] for these problems. In turn, this implies that the later SOCP based bounds can be strengthened while still using SOCP techniques.

The fact that the hierarchies considered here are based on using LP and SOCP allows for the future use of sparsity techniques proposed in [13, 19] and column generation approaches recently proposed in [1] to address the solution of larger-scale ACOPF problems. Additionally, we will explore the possibility to use the proposed SOCP hierarchy to approximate other energy related problems.

REFERENCES

[1] Ahmadi, A. A., Dash, S., and Hall, G. (2016). Optimization over structured subsets of positive semidefinite matrices via column generation. *Discrete Optimization*, to appear.

[2] Ahmadi, A. A. and Majumdar, A. (2014). DSOS and SDSOS optimization: LP and SOCP-based alternatives to sum of squares optimization. In *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*, pages 1–5. IEEE.

[3] Anjos, M. and Lasserre, J. B., editors (2012). *Handbook on Semidefinite, Conic and Polynomial Optimization Handbook on Semidefinite, Conic and Polynomial Optimization*, volume 166 of *International Series in Operations Research & Management Science*. Springer.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2016.2615688, IEEE Transactions on Power Systems

TABLE III.1
COMPARISON OF SDP RELAXATION, PP-SOS$_2$, PP-SDSOS$_2$ AND PP-DSOS$_2$.

	Best	SDP Relaxation			PP-SOS$_2$			PP-SDSOS$_2$			PP-DSOS$_2$		
Instance	Bound	Bound	Time (s.)	Gap	Bound	Time (s.)	Gap	Bound	Time (s.)	Gap	Bound	Time (s.)	Gap
case9Q	5297.4	5296.7	0.21	0.01%	5296.7	49.59	0.01%	5230.5	0.09	1.26%	4448.0	0.05	16.02%
case14	8081.7	8081.5	0.45	0.00%	8081.5	1.59	0.00%	7677.5	0.17	5.00%	6859.3	0.07	15.12%
case30	576.8	576.8	13.42	0.00%	-	-	-	567.9	0.48	1.53%	373.9	0.36	35.18%
case39	41889.1	41854.9	35.31	0.08%	-	-	-	41276.3	0.95	1.46%	3495.1	0.40	91.65%
case57	41738.3	41737.7	149.60	0.00%	-	-	-	41144.3	1.69	1.42%	38822.5	0.28	6.98%
case118	129372.4	129654.6	622.13	0.22%	-	-	-	126001.2	8.17	2.61%	96774.4	0.49	25.19%
case300	720031.0	719711.6	27361.52	0.04%	-	-	-	706616.5	81.69	1.86%	⋆	⋆	⋆

−: Program not solved within 1 hour.
⋆: Out of memory when generating the hierarchy.

TABLE III.2
COMPARISON OF R_2 WITH PP-SDSOS$_2$ AND STRUCTURED PP-SDSOS$_r$ APPROXIMATION.

	Best	R_2			PP-SDSOS$_2$			Structured PP-SDSOS$_2$			Structured PP-SDSOS$_4$		
Instance	Bound	Bound	Time(s)	Gap	Bound	Time(s)	Gap	Bound	Time(s)	Gap	Bound	Time(s)	Gap
case9Q	5297.4	5220.0	0.14	1.46%	5230.5	0.09	1.26%	5220.0	0.01	1.46%	5297.4	27.15	0.00%
case14	8081.7	7661.0	0.15	5.21%	7677.5	0.17	5.00%	7660.0	0.08	5.22%	7953.8	48.75	1.58%
case30	576.8	567.8	0.43	1.56%	568.0	0.48	1.53%	567.8	0.31	1.56%	⋆	⋆	⋆
case39	41889.1	41283.7	0.55	1.45%	41276.3	0.95	1.46%	41278.1	0.51	1.46%	⋆	⋆	⋆
case57	41738.3	41157.7	0.38	1.39%	41144.3	1.69	1.42%	41157.7	0.26	1.39%	⋆	⋆	⋆
case118	129372.4	126072.2	0.95	2.55%	126001.2	8.17	2.61%	126050.2	0.71	2.57%	⋆	⋆	⋆
case300	720031.0	706779.6	2.62	1.84%	706616.5	81.69	1.86%	706779.6	1.57	1.84%	⋆	⋆	⋆

⋆: Out of memory when generating the hierarchy.

[4] Bose, S., Low, S., Teeraratkul, T., and Hassibi, B. (2015). Equivalent relaxations of optimal power flow. *IEEE Transactions on Automatic Control*, 60(3):729–742.

[5] Bukhsh, W., Grothey, A., McKinnon, K., and Trodden, P. (2013). Local solutions of the optimal power flow problem. *IEEE Transactions on Power Systems*, 28(4):4780–4788.

[6] de Klerk, E. and Sotirov, R. (2010). Exploiting group symmetry in semidefinite programming relaxations of the quadratic assignment problem. *Mathematical Programming*, 122(2):225–246.

[7] Ghaddar, B., Marecek, J., and Mevissen, M. (2016a). Optimal power flow as a polynomial optimization problem. *IEEE Transactions on Power Systems*, 31(1):539–546.

[8] Ghaddar, B., Vera, J. C., and Anjos, M. F. (2011). Second-order cone relaxations for binary quadratic polynomial programs. *SIAM Journal on Optimization*, 21(1):391–414.

[9] Ghaddar, B., Vera, J. C., and Anjos, M. F. (2016b). A dynamic inequality generation scheme for polynomial programming. *Mathematical Programming*, 156(1):21–57.

[10] Jabr, R. (2006). Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems*, 21(3):1458–1459.

[11] Josz, C., Maeght, J., Panciatici, P., and Gilbert, J. (2015). Application of the moment-sos approach to global optimization of the opf problem. *IEEE Transactions on Power Systems*, 30(1):463–470.

[12] Kocuk, B., Dey, S. S., and Sun, X. A. (2016). Strong SOCP relaxations for the optimal power flow problem. *Operations Research*.

[13] Kojima, M., Kim, S., and Waki, H. (2003). Sparsity in sums of squares of polynomials. *Mathematical Programming*, 103(1):45–62.

[14] Lasserre, J. B. (2001). Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817.

[15] Lasserre, J. B. (2002). Semidefinite programming vs. lp relaxations for polynomial programming. *Mathematics of operations research*, 27(2):347–360.

[16] Lasserre, J. B. (2009). Moments and sums of squares for polynomial optimization and related problems. *Journal of Global Optimization*, 45(1):39–61.

[17] Lavaei, J. and Low, S. H. (2012). Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, 27(1):92–107.

[18] Molzahn, D. and Hiskens, I. (2014). Moment-based relaxation of the optimal power flow problem. In *Power Systems Computation Conference (PSCC), 2014*, pages 1–7.

[19] Molzahn, D. K., Holzer, J. T., Lesieutre, B. C., and DeMarco, C. L. (2013). Implementation of a large-scale optimal power flow solver based on semidefinite programming. *IEEE Transactions on Power Systems*, 28(4):3987–3998.

[20] Mosek, A. (2010). The MOSEK optimization software. *Online at http://www.mosek.com*.

[21] Murty, K. G. and Kabadi, S. N. (1987). Some np-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129.

[22] Pena, J., Vera, J., and Zuluaga, L. F. (2007). Computing the stability number of a graph via linear and semidefinite

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2016.2615688, IEEE Transactions on Power Systems

programming. *SIAM Journal on Optimization*, 18(1):87–105.

[23] Peña, J., Vera, J. C., and Zuluaga, L. F. (2014). A certificate of non-negativity for polynomials over unbounded sets. *Mathematics of Operations Research*.

[24] Renegar, J. (2001). *A Mathematical View of Interior-Point Methods in Convex Optimization*, volume 3 of *MPS/SIAM Ser. Optim.* SIAM.

[25] Sojoudi, S. and Lavaei, J. (2012). Physics of power networks makes hard optimization problems easy to solve. In *IEEE Power and Energy Society General Meeting*, pages 1–8.

[26] Sturm, J. F. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653.

[27] University of Washington (2015). Power systems test case archive. *http://www.ee.washington.edu/research/pstca/*.

APPENDIX

In this section, we provide a detailed proof of Theorem 1.

Proof. First, problem (PP-OPF) is reformulated as a PP of the form given in (PP-SDSOS$_r$). We notice that since not all the monomials appear in the polynomial formulation then using the first level of the SOCP hierarchy ($r = 2$), one can approximate (PP-OPF) as

$$
\begin{aligned}
&\max \ \varphi \\
&\text{s.t.} \ \sum_{k\in N}\left(c_k^2(P_k^g)^2 + c_k^1(P_k^d + \operatorname{tr}(Y_k x x^T)) + c_k^0\right) - \varphi = \\
&S(x) + S(P^g) + S(P,Q) + \sum_{k\in N}\overline{\lambda}_k\left(P_k^+ - P_k^d - \operatorname{tr}(Y_k x x^T)\right) \\
&+\sum_{k\in N}\underline{\lambda}_k\left(-P_k^- + P_k^d + \operatorname{tr}(Y_k x x^T)\right) + \sum_{k\in N}\overline{\gamma}_k\left(Q_k^+\right. \\
&\left.-Q_k^d - \operatorname{tr}(\bar{Y}_k x x^T)\right) + \sum_{k\in N}\underline{\gamma}_k\left(-Q_k^- + Q_k^d + \operatorname{tr}(\bar{Y}_k x x^T)\right) \\
&+\sum_{k\in N}\overline{\mu}_k\left((V_k^+)^2 - \operatorname{tr}(M_k x x^T)\right) + \sum_{k\in N}\underline{\mu}_k\left((-V_k^-)^2\right. \\
&\left.+\operatorname{tr}(M_k x x^T)\right) + \sum_{(l,m)\in L} a_{lm}\left((S_{lm}^+)^2 - P_{lm}^2 - Q_{lm}^2\right) \\
&+\sum_{k\in G} b_k\left(P_k^g - \operatorname{tr}(Y_k x x^T) - P_k^d\right) + \sum_{(l,m)\in L} c_{lm}\left(P_{lm}\right. \\
&\left.-\operatorname{tr}(Y_{lm} x x^T)\right) + \sum_{(l,m)\in L} d_{lm}\left(Q_{lm} - \operatorname{tr}(\bar{Y}_{lm} x x^T)\right),
\end{aligned}
\tag{12}
$$

where $S(x)$ is an SDSOS polynomial based on the branch information, and $S(P^g), S(P,Q)$ are SDSOS polynomials as a function of P_k^g, P_{lm} and Q_{lm} respectively. Furthermore, by observation of the monomials in the objective function and constraints of (PP-OPF) , we remove some unnecessary monomials in $S(x), S(P^g)$ and $S(P,Q)$ for simplicity. To be more specific, the following terms are not needed:

$$
\begin{aligned}
&x_k, \forall k \in N \text{ in } S(x), \\
&P_i^g P_j^g, \forall i,j \in N, i \neq j \text{ in } S(P^g), \\
&P_{l_1,m_1}P_{l_2,m_2}, \forall (l_1,m_1),(l_2,m_2) \in E, (l_1,m_1) \neq (l_2,m_2), \\
&Q_{l_1,m_1}Q_{l_2,m_2}, \forall (l_1,m_1),(l_2,m_2) \in E, (l_1,m_1) \neq (l_2,m_2), \\
&P_{l_1,m_1}Q_{l_2,m_2}, \forall (l_1,m_1),(l_2,m_2) \in E \text{ in } S(P,Q).
\end{aligned}
$$

The variables $\alpha, \beta, \theta, \alpha_k, \beta_k, \theta_{lm}, \delta_{im}, \overline{\lambda}_k, \underline{\lambda}_k, \overline{\gamma}_k, \underline{\gamma}_k, \overline{\mu}_k, \underline{\mu}_k$ and a_{lm} are non-negative variables and α_{lm}^1, α_{lm}^2, β_k^1, β_k^2, θ_{lm}^1, θ_{lm}^2, δ_{lm}^1, δ_{lm}^2, b_k, c_{lm} and d_{lm} are free variables. As discussed in Section II, an SDSOS polynomial has a natural second order cone representation, so we can write $S(x), S(P^g)$ and $S(P,Q)$ as

$$
\begin{aligned}
S(x) &= \alpha + \sum_{k\in N}\alpha_k x_k^2 + \sum_{(l,m)\in E}\begin{pmatrix} x_l \\ x_m \end{pmatrix}^T A_{lm}\begin{pmatrix} x_l \\ x_m \end{pmatrix}, \\
S(P^g) &= \beta + \sum_{k\in N}\beta_k (P_k^g)^2 + \sum_{k\in N}\begin{pmatrix} 1 \\ P_k^g \end{pmatrix}^T B_k \begin{pmatrix} 1 \\ P_k^g \end{pmatrix} \\
S(P,Q) &= \theta + \sum_{(l,m)\in E}(\theta_{lm}P_{lm}^2 + \delta_{lm}Q_{lm}^2) \\
&+ \sum_{(l,m)\in E}\left(\begin{pmatrix} 1 \\ P_{lm} \end{pmatrix}^T C_{lm}^1 \begin{pmatrix} 1 \\ P_{lm} \end{pmatrix} + \begin{pmatrix} 1 \\ Q_{lm} \end{pmatrix}^T C_{lm}^2 \begin{pmatrix} 1 \\ Q_{lm} \end{pmatrix}\right),
\end{aligned}
$$

Define η as $\eta = \alpha+\beta+\theta$, for ease of notation. By equating the coefficients of the monomials of Problem (12) and substituting some of the variables, we have the following second-order cone program:

$$
\begin{aligned}
\max \ & \sum_{k\in G} c_k^1 P_k^d - \sum_{k\in N} B_k^{(11)} - \eta - \sum_{(l,m)\in E}(C_{lm}^{1(11)} - C_{lm}^{2(11)}) \\
&+ \sum_{k\in G} c_k^0 - \sum_{k\in N}\overline{\lambda}_k(P_k^+ - P_k^d) - \sum_{k\in N}\underline{\lambda}_k(-P_k^- + P_k^d) \\
&- \sum_{k\in N}\overline{\gamma}_k(Q_k^+ - Q_k^d) - \sum_{k\in N}\underline{\gamma}_k(-Q_k^- + Q_k^d) \\
&- \sum_{k\in N}\overline{\mu}_k(V_k^+)^2 + \sum_{k\in N}\underline{\mu}_k(V_k^-)^2 - \sum_{k\in G} 2B_k^{1(12)}P_k^d \\
&- \sum_{(l,m)\in L}(\theta_{lm} + C_{lm}^{1(22)})(S_{lm}^+)^2 \\
\text{s.t.} \ & A_{ij}^{(12)} = \sum_{(l,m)\in L} 2C_{lm}^{1(12)}Y_{lm}^{(ij)} + \sum_{(l,m)\in L} 2C_{lm}^{2(12)}\bar{Y}_{lm}^{(ij)} \\
&+ \sum_{k\in N} c_k^1 Y_k^{(ij)} + \sum_{k\in N}\left(\overline{\lambda}_k Y_k^{(ij)} - \underline{\lambda}_k Y_k^{(ij)} + \overline{\gamma}_k \bar{Y}_k^{(ij)}\right. \\
&\left.-\underline{\gamma}_k \bar{Y}_k^{(ij)} + 2B_k^{1(12)}Y_k^{(ij)}\right), \ \forall (i,j) \in E, \\
&\alpha_i + \sum_{i:(i,j)\in E} A_{ij}^{(11)} + \sum_{j:(i,j)\in E} A_{ij}^{(22)} = \sum_{k\in N} c_k^1 Y_k^{(ii)} \\
&+ \sum_{(l,m)\in L}\left(2C_{lm}^{1(12)}Y_{lm}^{(ii)} + 2C_{lm}^{2(12)}\bar{Y}_{lm}^{(ii)}\right) \\
&+ \sum_{k\in N}\left(\overline{\lambda}_k Y_k^{(ii)} - \underline{\lambda}_k Y_k^{(ii)} + \overline{\gamma}_k \bar{Y}_k^{(ii)} - \underline{\gamma}_k \bar{Y}_k^{(ii)}\right. \\
&\left.+2B_k^{1(12)}Y_k^{(ii)} + \overline{\mu}_k - \underline{\mu}_k\right), \ \forall i \in N, \\
& c_k^2 = \beta_k + B_k^{(22)}, \qquad \forall k \in N, \\
& \theta_{lm} + C_{lm}^{1(22)} - \delta_{lm} - C_{lm}^{2(22)} = 0, \ \forall (l,m) \in E, \\
& \eta \geq 0, \\
& \alpha_k, \beta_k, \overline{\lambda}_k, \underline{\lambda}_k, \overline{\gamma}_k, \underline{\gamma}_k, \overline{\mu}_k, \underline{\mu}_k \geq 0, \quad \forall k \in N, \\
& \theta_{lm}, \delta_{lm} \geq 0, \quad \forall (l,m) \in E, \\
& B_k \succeq 0, \quad \forall k \in N, \\
& A_{lm}, C_{lm}^1, C_{lm}^2 \succeq 0, \quad \forall (l,m) \in E,
\end{aligned}
\tag{13}
$$

where $B_k, A_{lm}, C_{lm}^1, C_{lm}^2$ are 2×2 positive semidefinite matrices. From Equation (3), $B_k, A_{lm}, C_{lm}^1, C_{lm}^2 \succeq 0$ can be represented as second order cone constraints, so Problem (13) is a second order cone program.

Next we derive the dual of Problem $\mathcal{R}_2$ in [4]. In [4], the authors also use the notations in [17]. Also note that for ease of presentation, the authors omit some refinements (such as the shunt element) in their model.

Let $Z_k^{12} = Z_k^{21} = \sqrt{c_{k2}} \sum_{e \in E} \text{tr}(Y_k^e W_e) + b_k$ be the non-diagonal element of matrix Z_k, and we present the model of Problem $\mathcal{R}_2$ in [4] derived from Optimization 3 in [17] as follows:

$$
\begin{aligned}
\min \quad & \sum_{k \in N} \alpha_k \\
\text{s.t.} \quad & P_k^- \le \sum_{e \in E} \text{tr}(Y_k^e W_e) + P_k^d \le P_k^+, \ \forall k \in N, \\
& Q_k^- \le \sum_{e \in E} \text{tr}(\bar{Y}_k^e W_e) + Q_k^d \le Q_k^+, \ \forall k \in N, \\
& (V_k^-)^2 \le \sum_{e \in E} \text{tr}(M_k^e W_e) \le (V_k^+)^2, \ \forall k \in N, \\
& Z_k = \begin{bmatrix} \alpha_k - c_{k1} \sum\limits_{e \in E} \text{tr}(Y_k^e W_e) - a_k & Z_k^{12} \\ Z_k^{21} & 1 \end{bmatrix} \succeq 0, \ k \in N \\
& Z_{lm}^1 = \begin{bmatrix} S_{lm,\max}^2 & \sum\limits_{e \in E} \text{tr}(Y_k^e W_e) \\ \sum\limits_{e \in E} \text{tr}(Y_k^e W_e) & 1 \end{bmatrix} \succeq 0, \ \forall (l,m) \in E \\
& Z_{lm}^2 = \begin{bmatrix} S_{lm,\max}^2 & \sum\limits_{e \in E} \text{tr}(\bar{Y}_k^e W_e) \\ \sum\limits_{e \in E} \text{tr}(\bar{Y}_k^e W_e) & 1 \end{bmatrix} \succeq 0, \ \forall (l,m) \in E \\
& W_e \succeq 0, \ \forall e \in E,
\end{aligned}
\tag{14}
$$

where $\text{tr}(Y_k W)$ in Optimization 3 [17] is replaced by $\sum_{e \in E} \text{tr}(Y_k^e W_e)$ to take the branch structure into account and we relax constraint (5) in Optimization 3 [17] to two second-order cone constraints Z_{lm}^1, Z_{lm}^2. So basically, optimization problem (14) is a relaxation of Optimization 3 in [17] in terms of the variable W_e. The Lagrangian function of optimization problem (14) is

$$
\begin{aligned}
\mathcal{L}(W, Z, \alpha, \lambda, \gamma, \mu, \eta, A, B, C) &= \sum_{k \in G} \alpha_k + \sum_{e \in E} \text{tr}(W_e, A_e) \\
&+ \sum_{k \in N} \text{tr}(Z_k, B_k) + \sum_{(l,m) \in E} \left(\text{tr}(Z_{lm}^1, C_{lm}^1) + \text{tr}(Z_{lm}^2, C_{lm}^2) \right) \\
&+ \sum_{k \in N} \overline{\lambda}_k \left(P_k^+ - P_k^d - \sum_{e \in E} \text{tr}(Y_k^e W_e) \right) \\
&+ \sum_{k \in N} \underline{\lambda}_k \left(-P_k^- + P_k^d + \sum_{e \in E} \text{tr}(Y_k^e W_e) \right) \\
&+ \sum_{k \in N} \overline{\gamma}_k \left(Q_k^+ - Q_k^d - \sum_{e \in E} \text{tr}(\bar{Y}_k^e W_e) \right) \\
&+ \sum_{k \in N} \underline{\gamma}_k \left(-Q_k^- + Q_k^d + \sum_{e \in E} \text{tr}(\bar{Y}_k^e W_e) \right) \\
&+ \sum_{k \in N} \overline{\mu}_k \left((V_k^+)^2 - \sum_{e \in E} \text{tr}(M_k^e W_e) \right) \\
&+ \sum_{k \in N} \underline{\mu}_k \left((-V_k^-)^2 + \sum_{e \in E} \text{tr}(M_k^e W_e) \right),
\end{aligned}
$$

where $A_e, B_k, C_{lm}^1, C_{lm}^2$ are 2×2 positive semidefinite matrices and $\lambda, \gamma, \mu, \eta$ are nonnegative Lagrangian multipliers. Therefore the Lagrangian dual problem of optimization problem (14) is

$$
\begin{aligned}
\max_{\lambda,\gamma,\mu,\eta,A,B,C} \quad & \min_{W, Z \succeq 0, \alpha} \mathcal{L}(W, Z, \alpha, \lambda, \gamma, \mu, \eta, A, B, C) \\
\text{s.t.} \quad & \lambda, \gamma, \mu, \eta \ge 0 \\
& A_e \succeq 0, \ \forall e \in E, \\
& B_k \succeq 0, \ \forall k \in N, \\
& C_{lm}^1, C_{lm}^2 \succeq 0, \ \forall (l,m) \in E.
\end{aligned}
\tag{15}
$$

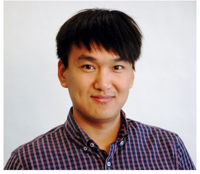
Once expanded, problem (15) has the same structure to the first level SOCP hierarchy for (PP-OPF) given in (12), with exactly the same conic constraints. ☐

Next we prove the strong duality in Theorem 2.

Proof. In order to prove strong conic duality, we need to find a strict feasible solution for either the primal or dual problem. For the dual SOCP problem (12), consider the following point,

$$
\begin{aligned}
& \eta > 0, \\
& \alpha_k = |E|_{max} + 1 - |E_k| > 0, \beta_k = c_k^2 - \epsilon > 0, \quad \forall k \in N, \\
& \theta_{lm} = \delta_{lm} = 1, \quad \forall (l,m) \in E, \\
& \underline{\lambda}_k = c_k^1 + \Lambda > 0, \overline{\lambda}_k = \Lambda, \underline{\gamma}_k = \overline{\gamma}_k = 1, \quad \forall k \in N, \\
& \overline{\mu}_k = 1 + \frac{1 + |E|_{max}}{|N|} > 0, \underline{\mu}_k = 1, \quad \forall k \in N, \\
& B_k = \begin{bmatrix} \epsilon & 0 \\ 0 & \epsilon \end{bmatrix} \succ 0, \quad \forall k \in N, A_{lm} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \succ 0, \\
& C_{lm}^1 = C_{lm}^2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \succ 0, \quad \forall (l,m) \in E.
\end{aligned}
\tag{16}
$$

In (16), $|E_k|$ is the number of edges whose endpoints contain node k, $|E|_{max} = \max\{|E_k|, k \in N\}$, for a fixed graph, $|E_k|, |E|_{max}$ are parameters; ϵ is a sufficient small positive number such that $c_k^2 - \epsilon > 0, \forall k \in N$, as c_k^2 is a positive parameter; Λ is a sufficient large positive number such that $c_k^1 + \Lambda > 0$. We substitute (16) into the constraints in Problem (12) and it is easy to verify that it satisfies all the constraints. Thus it is straightforward to see that the point is in the interior of the feasible set, which means point (16) is a strict feasible point for problem (12). Therefore, strong duality holds for Theorem 2. ☐



Xiaolong Kuang is a Ph.D. candidate in Department of Industrial and Systems Engineering at Lehigh University. He received his B.S. degree in Systems Engineering from Huazhong University of Science and Technology in 2012 and received his M.Eng. degree in Industrial and Systems Engineering from Lehigh University in 2014. His research areas mainly focus on polynomial optimization, conic optimization and their applications in power systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPWRS.2016.2615688, IEEE Transactions on Power Systems

Bissan Ghaddar is an Assistant Professor in Data Analytics and Operations Research at the University of Waterloo working on problems at the intersection of smart cities, data analytics, and mathematical modeling and optimization for large-scale problems. Prior to joining the University of Waterloo, Bissan worked on energy and transportation network optimization at IBM Research Dublin and on inventory management problems at the Centre for Operational Research and Analysis, Department of National Defence Canada. Her research has been supported by prestigious national and international scholarships and grants.

Joe Naoum-Sawaya is an Assistant Professor of Management Science at the Ivey Business School. He holds a Ph.D. in Management Science/Operations Research from the University of Waterloo. Prior to joining Ivey, Joe was a research scientist at IBM Research. Joes research focus is on machine learning and optimization with applications in smarter cities (energy, mobility, water, and telecommunication). His work has appeared in INFORMS Journal on Computing, European Journal of Operational Research, Transportation Research, and Naval Research Logistics among others.

Luis F. Zuluaga earned his Ph.D. in Operations Research from the Tepper School of Business, Carnegie Mellon University. Currently, he serves as Assistant Professor of the Industrial and Systems Engineering, Lehigh University. Prior to joining the Lehigh faculty, he served as an Associate Professor at the Faculty of Business Administration, University of New Brunswick. Luis has authored multiple articles in journals such as Operations Research, SIAM Journal on Optimization, and Mathematics of Operations Research. The recipient of numerous awards, honors, and grants. Luis' research looks at developing solution schemes and effective algorithms for problems involving optimization over polynomials; especially those arising from integer programming, probability, and finance applications.