Least Square Estimation-Based SDP Cuts for SOCP Relaxation of AC OPF

Zhixin Miao, *Senior Member, IEEE,* Lingling Fan, *Senior Member, IEEE*, Hoseein Ghassempour, and Bo Zeng, *Member, IEEE*

***Abstract*—It has been known that the second-order conic programming (SOCP) relaxation of an alternating current optimal power flow (AC OPF) problem is a computationally friendly formulation, while the semi-definite programming (SDP) relaxation is a theoretically stronger one. This paper presents a method to strengthen the (SOCP) relaxation by generating new cutting planes, i.e., valid inequalities, using SDP relaxation, which remove SOCP solutions that are infeasible to SDP formulation. This new method relies on solving a least square estimation (LSE) problem for every cycle in a cycle basis. General feasibility cutting plane method is also employed for cuts generation. We show that the SDP cuts generated by the LSE method are indeed feasibility cuts. Numerical results show that those new cuts can effectively reduce the search space and lead to a tighter relaxation. The new cuts are comparable to the SDP cuts in [1]. Case studies on systems with several buses to thousands buses have demonstrated the method is also scalable.**

***Index Terms*—SOCP, SDP, AC OPF, Least square estimation**

I. Introduction

Convex relaxation has been applied to AC OPF to obtain a strong solution within polynomial solving time. Among them, SDP relaxation have been applied in AC OPF first in [2]. Lavaei and Low showed that SDP relaxation of AC OPF provides a tight bound [3]. It has also been shown that moment relaxation [4]–[6] enables the application of the Lassere hierarchy for polynomial optimization problems. The first-order moment relaxation is equivalent to the SDP relaxation in [3]. The disadvantage of SDP relaxation of AC OPF is that it has scalability issues as demonstrated by computing experiments in [7].

Therefore, other types of convex relaxations with computing efficiency have been proposed, e.g., linear programming relaxation [8], quadratic programming relaxation [9], and SOCP relaxation [7], [10]. In particular, the latter one has demonstrated a great potential to achieve a desired trade-off between the tightness of relaxation and the computational advantage [1], [7]. Nevertheless, the standard SOCP relaxation does not take care of the meshed network cycle constraints (i.e., sum of the voltage angle differences across a cycle is zero). Most recently, [1] presents three approaches to strengthen SOCP relaxation of AC OPF. Note that the feasible region of SDP relaxation is contained inside the feasible region of the SOCP relaxation [1], [9]. So, one of those approaches is to generate valid homogeneous inequalities to reduce the search space by excluding an SOCP relaxation solution that is infeasible to the SDP relaxation. Such valid inequalities, i.e., cuts, are named as SDP cuts. This approach shows it can produce relaxations with smaller gaps compared to other two approaches in [1].

Z. Miao, L. Fan, and H Ghassempour are with Department of Electrical Engineering at University of South Florida, Tampa, FL 33620 (Email: `linglingfan@usf.edu`). B. Zeng is with University of Pittsburgh.

A similar idea regarding SDP cuts can be found in [11], where nonlinear inequalities are generated to represent SDP cuts for SOCP relaxation problems. The inequalities are based on the matrix determinant. The method in [11] can be applied to large networks by exploiting sparsity using the branch decomposition.

Compared to nonlinear inequalities, linear or affine inequalities will have computational advantages for large-size networks. In this paper, we will propose a least square estimation (LSE)-based method to generate a new type of SDP cuts that are affine inequalities. We then compare the LSE-based method with the general feasibility cutting plane method [12] and show that those two approaches generate the same cuts. On instances of different scales, we observed that our method is comparable to the SDP separation method in [1].

The rest of the paper is organized as follows. Section II presents AC OPF relaxations. Both SOCP and SDP relaxations are presented. The SDP separation method presented in [1] is examined in details. Section III presents the proposed LSE method and feasibility cut method. Section IV presents case study results and Section V concludes the paper.

II. AC OPF Relaxations

AC OPF is formulated as an optimization problem with the objective function as the cost of generation or power loss, equality constraints representing the relationship of bus power injection versus bus voltage magnitudes (notated by a vector $V \in \mathbb{R}^{|\mathcal{B}|}$, where $\mathcal{B}$ is the set of the buses in the system and $|.|$ notates the cardinality of a set) and phase angles ($\theta \in \mathbb{R}^{|\mathcal{B}|}$), and inequality constraints representing voltage limits, line flow limits, generation limits, etc [13]. The set of the branches are notated as $\mathcal{L}$.

The decision variables of AC OPF are voltage magnitudes V, phase angles θ, and generators' real and reactive power outputs, notated as P_i^g, Q_i^g, where $i \in \mathcal{G}$, and $\mathcal{G} \subseteq \mathcal{B}$ is the subset of the buses.

Given the system admittance matrix $Y = G+jB$, the power injection at every node can be expressed by V and θ.

$$P_i^g - P_i^d = \sum_{j\in\delta_i} V_iV_j(G_{ij}\cos(\theta_i-\theta_j)+B_{ij}\sin(\theta_i-\theta_j))$$

$$Q_i^g - Q_i^d = \sum_{j\in\delta_i} V_iV_j(G_{ij}\sin(\theta_i-\theta_j)-B_{ij}\cos(\theta_i-\theta_j))$$

where superscript g notates generator's output and d notates load consumption, δ_i is the set of the buses that are directly connected to Bus i through branches.

The mathematical program is shown in (1).

$$
\begin{aligned}
\min \quad & \sum_{i\in\mathcal{G}} f_i(P_i^g) && && \text{(1a)}\\
s.t. \quad & P_i^g - P_i^d - P_i(V,\theta) = 0, && i \in \mathcal{B} && \text{(1b)}\\
& Q_i^g - Q_i^d - Q_i(V,\theta) = 0, && i \in \mathcal{B} && \text{(1c)}\\
& |S_{ij}(V,\theta)| - S_{ij}^M \le 0, && (i,j) \in \mathcal{L} && \text{(1d)}\\
& V_i^m \le V_i \le V_i^M, && i \in \mathcal{B} && \text{(1e)}\\
& P_i^{gm} \le P_i^g \le P_i^{gM}, && i \in \mathcal{G} && \text{(1f)}\\
& Q_i^{gm} \le Q_i^g \le Q_i^{gM}, && i \in \mathcal{G} && \text{(1g)}
\end{aligned}
$$

where $f(.)$ is the cost function, superscript M denotes upper bound and m denotes low bound, P_i^d, Q_i^d are the real and reactive power load consumption at Bus i, $P_i(V,\theta)$ and $Q_i(V,\theta)$ are Bus i's power injection expressions in terms of bus voltage magnitudes and phase angles, and $S_{ij}(V,\theta)$ is the complex power flow from Bus i to Bus j on the branch connecting the two buses. The decision variables consist of $P_i^g, Q_i^g, i \in \mathcal{G}$, V, and θ. The dimension of the decision variable vector is $2|\mathcal{G}| + 2|\mathcal{B}|$.

Note that the equality constraints of power injections are non-convex in terms of V and θ. Relaxations have been developed in the literature to have a convex feasible region. These methods deal with new sets of decision variables to replace V and θ.

A. SOCP relaxation

In SOCP relaxation [10], a new set of variables c_{ij} and s_{ij} is used to replace the voltage phasors $V_i\angle\theta_i, i \in \mathcal{B}$, where $\mathcal{B}$ is the set of buses in a power network.

$$
\begin{aligned}
& c_{ii} = V_i^2,\ c_{ij} = V_iV_j\cos(\theta_i - \theta_j)\\
& s_{ii} = 0,\ s_{ij} = -V_iV_j\sin(\theta_i - \theta_j)
\end{aligned}
\tag{2}
$$

where $c_{ij} = c_{ji}$ and $s_{ij} = -s_{ji}$.

It is easy to find the following relationship:

$$
c_{ij}^2 + s_{ij}^2 = V_i^2V_j^2 = c_{ii}c_{jj}. \tag{3}
$$

The AC OPF problem's power injection constraints are linear in terms of c_{ij} and s_{ij}.

$$
\begin{aligned}
P_i^g - P_i^d &= \sum_{j\in\delta_i}(G_{ij}c_{ij} - B_{ij}s_{ij}),\\
Q_i^g - Q_i^d &= \sum_{j\in\delta_i}(-G_{ij}s_{ij} - B_{ij}c_{ij}).
\end{aligned}
\tag{4}
$$

There will be $|\mathcal{L}|$ number c_{ij} and s_{ij} to be defined as decision variables. If there is no direct connection between Bus i and Bus j, the power injection equations will not contain c_{ij} nor s_{ij}. The decision variables V and θ are now replaced by $c_{ii}, i \in \mathcal{B}$, and $c_{ij}, s_{ij}, (i,j) \in \mathcal{L}$. The dimension of the new set of the variables is $|\mathcal{B}| + 2|\mathcal{L}|$.

In addition, (3) will be relaxed as a second-order cone:

$$
c_{ij}^2 + s_{ij}^2 \le c_{ii}c_{jj}. \tag{5}
$$

This relaxation is first proposed in [10] for AC OPF and named as SOCP relaxation.

Research work has been conducted in this area and sufficient conditions for SOCP relaxation being exact are also found [7], [14]. In spanning tree power networks, under a mild condition (e.g., voltage upper bounds not binding), SOCP relaxations are exact [7]. For meshed networks, SOCP relaxation is not exact since the following constraint is not considered in SOCP relaxation:

$$
\tan(\theta_i - \theta_j) = -\frac{s_{ij}}{c_{ij}} \tag{6}
$$

For a cycle C, the angle constraint is as follows

$$
\sum_{(i,j)\in C}\theta_{ij} = 2\pi k, \text{for some } k \in \mathbb{Z}. \tag{7}
$$

Without imposing cycle constraints (6) and (7), it is possible to end up with c_{ij}, s_{ij} that violate the cycle constraint. In that case, the solution from SOCP relaxation is infeasible to the original AC OPF problem.

To solve this issue, in [15], Jabr proposed linear approximation for (6). This method requires iteration. In addition, feasible points could be lost due to the imposed linear constraint. Kocuk *et al* proposed three methods in [1], including McCormick based linear programming relaxation and separation, SDP separation, and arctangent envelopes to deal with the cycle constraints. In [1], the authors claim that the SDP separation works the best in term of providing tight gaps. The SDP cuts in [1] are represented by linear inequalities.

In this article, we will examine the SDP separation in [1] and develop SDP cuts using least square estimation (LSE). As a first task, the relationship of the decision variables of SOCP relaxation (c_{ii}, c_{ij}, s_{ij}) and those in SDP relaxation will be examined.

B. SDP relaxation

In SDP relaxation, rectangular expressions are used to represent the voltage phasors.

$$
V_i = V_i\angle\theta_i = \underbrace{V_i\cos\theta_i}_{e_i} + j\underbrace{V_i\sin\theta_i}_{f_i} \tag{8}
$$

A matrix W is defined as follows

$$
W = \begin{bmatrix} e \\ f \end{bmatrix}\begin{bmatrix} e^T & f^T \end{bmatrix} \tag{9}
$$

where $f = (f_1, f_2, \cdots, f_n)^T$, $e = (e_1, e_2, \cdots, e_n)^T$, $n = |\mathcal{B}|$.

It is obvious to find the following characteristics:

$$
W = W^T, \quad W \succeq 0, \text{and rank}(W) = 1. \tag{10}
$$

$W \succeq 0$ means that this matrix is positive semi-definite (PSD).

The power injection constraints will be shown to be linear with the elements of W. Define

$$
i' = i + |\mathcal{B}|, \quad j' = j + |\mathcal{B}|, \tag{11}
$$

where $|.|$ notates the cardinality of a set.

$$
\begin{aligned}
P_i^g - P_i^d &= \sum_{j=1}^{n} \left(G_{ij}(W_{ij} + W_{i'j'}) + B_{ij}(W_{ji'} - W_{ij'})\right) \\
Q_i^g - Q_i^d &= \sum_{j=1}^{n} (G_{ij}(W_{i'j} - W_{ij'}) + B_{ij}(W_{ij} + W_{i'j'}))
\end{aligned} \tag{12}
$$

The above expressions indicate that the equality constraints of power injection are linear in terms of W. If the cost function is quadratic to P^g, then the cost function is quadratic to the elements of W. Without the rank 1 constraint rank$(W) = 1$, this problem is a convex problem and a semi-definite programming (SDP) problem.

The SDP relaxation of AC OPF is first introduced in [2]. The decision variable for SDP relaxation is $W \in \mathbb{R}^{2|\mathcal{B}|\times 2|\mathcal{B}|}$, P_i^g and Q_i^g for $i \in \mathcal{G}$. For large-scale networks, the size of W becomes large which makes computing cost of SDP relaxation increase significantly.

C. Relations of SOCP and SDP decision variables

In SDP relaxation of OPF, W is treated as a decision variable and W should be PSD. For SOCP relaxation of OPF, the decision variables are c_{ii}, c_{ij} and s_{ij}. The following relationship should hold:

$$
\begin{aligned}
c_{ij} &= e_i e_j + f_i f_j = W_{ij} + W_{i'j'}, && (13a)\\
s_{ij} &= e_i f_j - e_j f_i = W_{ij'} - W_{ji'}, && (13b)\\
c_{ii} &= e_i^2 + f_i^2 = W_{ii} + W_{i'i'}. && (13c)
\end{aligned}
$$

For every c_{ij}, s_{ij} and c_{ii}, we can express them to be the Frobenius product related to the (PSD) matrix W: $z_l = A_l \bullet W$, where $\bullet$ denotes Frobenious product.

D. SDP separation in [1]

Research in [1] indicates that once a solution vector z is found after solving the SOCP relaxation of OPF, this z should be examined. If z is in the set that can be connected with a PSD W as shown in (13), then z is a solution of SDP relaxation. Otherwise we should create inequalities as cuts. (13) is further expressed as (14) for each chordless cycle in the power network. $\tilde{W}$ is the corresponding W for this cycle.

$$
\begin{aligned}
\mathcal{S} := \Big\{ & z \in \mathbb{R}^{3|C|} : \exists \tilde{W} \in \mathbb{R}^{2|C|\times 2|C|} \\
\text{s.t. } & -z_l + A_l \bullet \tilde{W} = 0 \quad \forall l \in L, \quad \tilde{W} \succeq 0 \Big\},
\end{aligned} \tag{14}
$$

where C is a cycle, and L is the index set for all equalities, including buses and lines in this cycle. In a chordless cycle, the number of branches equals the number of nodes. Therefore, the dimension of z should be $3|C|$. For example, for a three-bus system with three branches connecting every two buses, the dimension of L is $3|C| = |\mathcal{B}| + 2|\mathcal{L}| = 9$.

The method to create cuts in [1] is shown as follows. For a given z^*, the separation problem over $\mathcal{S}$ can be written as follows,

$$
\begin{aligned}
v^* := \min_{\alpha,\lambda} \quad & -\alpha^T z^* && (15a)\\
\text{s.t.} \quad & \sum_{l\in L} \lambda_l A_l \succeq 0 && (15b)\\
& \alpha + \lambda = 0 && (15c)\\
& -e \le \alpha \le e. && (15d)
\end{aligned}
$$

If $v^* < 0$, that means z^* is infeasible to $\mathcal{S}$. The cut generated should be $\alpha^T z \le 0$.

The principle that leads to the above optimization problem was briefly mentioned in [1] as SDP duality. In this article, we give a detailed explanation of the principle used in [1].

Based on SDP duality [16], the following two problems are dual problems.

Primal problem:

$$
\boxed{\begin{aligned}
\min_X \quad & C \bullet X \\
s.t. \quad & A_i \bullet X = b_i, \\
& i = 1, \cdots, m \\
& X \succeq 0
\end{aligned}}
$$

Dual problem:

$$
\boxed{\begin{aligned}
\max_y \quad & y^T b \\
s.t. \quad & \sum_{i=1}^{m} A_i y_i \preceq C
\end{aligned}} \tag{16}
$$

where $A_i, i = 1, \cdots, m$ and C are symmetric matrices, $b \in \mathbb{R}^m$.

(14) can be written in a format similar as the primal problem as shown in (17) and its dual problem can also be found, as shown in (18).

Primal problem:

$$
\boxed{\begin{aligned}
p^* = \min_{\tilde{W}} \quad & \mathbf{0} \bullet \tilde{W} \\
s.t. \quad & A_l \bullet \tilde{W} = z_l^*, \\
& l \in L \\
& \tilde{W} \succeq 0
\end{aligned}} \tag{17}
$$

Dual problem:

$$
\boxed{\begin{aligned}
d^* = \max_{\alpha} \quad & \alpha^T z^* \\
s.t. \quad & \sum_{l\in L} \alpha_l A_l \preceq \mathbf{0}
\end{aligned}} \tag{18}
$$

If $z^* \in \mathcal{S}$, then the primal problem is feasible and the value of the primal problem $p^* = 0$. Based on weak duality, the value of the dual problem $d^* \le p^* = 0$. If z^* is not a SDP feasible solution, then $d^* > 0$. This also translates to $v^* = -d^* < 0$ when z^* is infeasible.

Remarks: An important assumption used to create the minimization problem in (15) is that matrices $A_l, l \in L$ should be symmetric. If we build those matrices based on (13) directly, the resulting A_l matrices are not symmetric. This will cause the minimization problem (15) to have a value always zero and the resulting vector α is always zero.

Since $\tilde{W}$ is a symmetric matrix, we can construct A_l to be symmetric. For example, let

$$
c_{ij} = W_{ij} + W_{i'j'} = \frac{1}{2}(W_{ij} + W_{ji} + W_{i'j'} + W_{j'i'}).
$$

This operation makes sure that the A_l constructed will be symmetric.

An **alternative** intuition is also given. The sufficient and necessary condition for a matrix $A \in \mathbb{R}^{n\times n}$ is PSD is for all $B \in \mathbb{R}^{n\times n}$ and $B \succeq 0$, $A \bullet B \ge 0$ [17].

The objective of the SDP separation in [1] is find a vector α that can make $\alpha^T z^* > 0$ if $z^* \notin \mathcal{S}$ while it can make $\alpha^T z \le 0$, where $z \in \mathcal{S}$. Since the SDP feasible region is a

closed convex cone, such a homogenous separation hyperplane always exists ([6], chapter 4, pp. 51).
The constraint can be further written as

$$\alpha^T z = \sum_l \alpha_l z_l = (\sum_l \alpha_l A_l) \bullet \tilde{W} \leq 0, \quad \text{where } \tilde{W} \succeq 0.$$

The sufficient and necessary condition for the above relation to be true is to have

$$-\sum_l \alpha_l A_l \succeq 0.$$

Therefore, the cut creation can be written in the same format as shown in (15).

$$\begin{aligned} &\max_{\alpha} \quad \alpha^T z^* \\ &\text{s.t.} \quad -\sum_l \alpha_l A_l \succeq 0 \end{aligned} \tag{19}$$

The value of the above maximization should be greater than zero for $z^* \notin \mathcal{S}$.

III. Generating SDP Cuts

A. Generating SDP Cuts based on Least Square Estimation

In this subsection, LSE based method is used to find new SDP cuts. Suppose we have a given z_0 (z_0 can be obtained after solving the SOCP relaxation of AC OPF) and the SDP feasible set is $\mathcal{S}$. We would like to generate cuts to get rid of z_0 and reduce the search space. We use the following method as shown in Fig. 1.

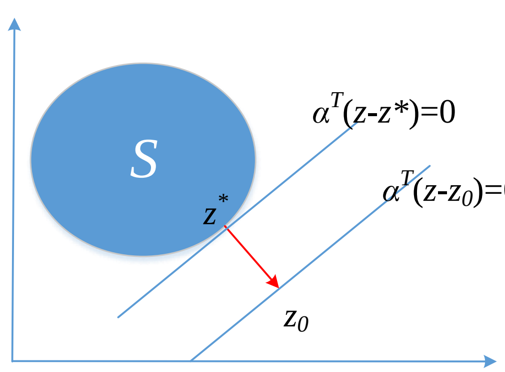

Fig. 1. z^* is z_0's projection to $\mathcal{S}$. The vector $(z_0 - z^*)$ can be used to create a cut. The cut generated will be $(z_0 - z^*)^T(z - z^*) \leq 0$.

First, we will find the shortest distance from z_0 to the set $\mathcal{S}$, z^* is the corresponding point found in $\mathcal{S}$. The we generate a line that is orthogonal to $z_0 - z^*$. Due to the orthogonality, the vector $z - z_0$ for any z located on the line, is orthogonal to the vector $z_0 - z^*$. Therefore, their inner product is zero.

$$(z_0 - z^*)^T(z - z_0) = 0 \tag{20}$$

for any z located on the line. Hence this line is defined as $\alpha^T(z - z_0) = 0$, where α is $(z_0 - z^*)$.

The set $\mathcal{S}$ is now located at the left of the line. Therefore the cut is generated as

$$\alpha^T(z - z_0) \leq 0, \tag{21}$$

where z is the variable.

The task left to us is to find z^*. This can be done through minimizing the distance from z_0 to z where $z \in S$. The formulation is as follows.

$$\min_z \quad \|z_0 - z\|_2 \tag{22a}$$
$$s.t. \quad z_l = \text{Trace}(A_l W^T), \text{ for all } l \in L \tag{22b}$$
$$W \succeq 0 \tag{22c}$$

where L the the index set including buses and branches. The optimal solution is notated as z^*.

If the norm of $z_0 - z^*$ is zero, that means z_0 belongs to the SDP set $\mathcal{S}$ and $\alpha = 0$ or no cuts will be generated.

The cut generated by (21) is a neutral cut [12], i.e., when $z = z_0$, $\alpha^T(z - z_0) = 0$. z_0 still belongs to the search space. To have a deep cut so that z_0 will be excluded from the search space or the feasible region, we will use z^*, the optimal solution from (22) to generate cut. The cut is expressed as follows.

$$\alpha^T(z - z^*) \leq 0 \tag{23}$$

where z is the decision variable vector.

(22) requires to find a W of a large size. To alleviate the computational burden, we follow the strategy presented in [1] to consider a small-size one constructed using cycle basis. For every cycle in the cycle basis of the network, we will solve a minimization problem (22) and generate a cut if the solution z_0 generated by SOCP relaxation is not in the feasible region of the SDP relaxation. Cycle basis identification algorithm in [18] is used to identify the cycle basis.

B. Generating SDP Cuts based on feasibility cuts

We further examine how to generate feasibility cuts for a given infeasible decision variable. The basic concept of feasibility cut [12] is first described. Then we examine the feasibility problem presented in (14).

For an inequality constraint $f(x) \leq 0$, where $x \in \mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function, notate the feasible region as $\mathcal{X}$. If x is not feasible and makes $f(x) > 0$, we can find the following relationship based on the convexity of the function f:

$$f(z) \geq f(x) + g^T(z - x) \tag{24}$$

where g is a gradient or subgradient. Any feasible z should satisfies the inequality $f(z) \leq 0$. Therefore, we can generate a deep cut as

$$0 \geq f(z) \geq f(x) + g^T(z - x) \tag{25}$$
$$\Rightarrow f(x) + g^T(z - x) \leq 0 \tag{26}$$

where z is the decision variable, x is given.

For the feasibility problem described in (14), we will first come up with an inequality constraint. Note that if z is infeasible, then the following relationship is true.

$$\sqrt{\sum_l (z_l - A_l \bullet \tilde{W})^2} > 0 \tag{27}$$

for some PSD $\tilde{W}$.

For a feasible z, then

$$\sqrt{\sum_l (z_l - A_l \bullet \tilde{W})^2} \leq 0, \tag{28}$$

for some PSD $\tilde{W}$.

The inequality constraint is identified as

$$f(z) = \left\{ \min_{\tilde{W}} \sqrt{\sum_l (z_l - A_l \bullet \tilde{W})^2} \right\} \leq 0, \tag{29}$$

for some PSD $\tilde{W}$.

The above inequality can be further written as

$$f(z) = \left\{ \min_{\tilde{W}} \quad \|z - z^*\|_2 \right\} \leq 0 \tag{30}$$

where $z_l^* = A_l \bullet \tilde{W}, l \in L$ and $\tilde{W} \succeq 0$.

If z_0 is infeasible, the feasibility cut should be

$$g^T(z - z_0) + f(z_0) \leq 0 \tag{31}$$

where g is the gradient. The gradient of $f(z)$ can be found as:

$$g(z) = \frac{\partial f}{\partial z} = \frac{z - z^*}{\sqrt{(z - z^*)^T(z - z^*)}} \tag{32}$$

Evaluated at z_0, $g(z)$ becomes

$$g(z_0) = \frac{\partial f}{\partial z} = \frac{z_0 - z^*}{\sqrt{(z_0 - z^*)^T(z_0 - z^*)}} = \frac{z_0 - z^*}{f(z_0)} \tag{33}$$

Therefore the feasibility cut is

$$\begin{aligned} & g^T(z - z_0) + f(z_0) \leq 0 \\ \Rightarrow & (z_0 - z^*)^T(z - z_0) + (z_0 - z^*)^T(z_0 - z^*) \leq 0 \\ \Rightarrow & (z_0 - z^*)^T(z - z^*) \leq 0 \end{aligned} \tag{34}$$

Using feasibility cut, we have successfully proved that the LSE-cut is indeed a feasibility cut.

C. Implementing procedure

The cuts will be generated for a solution from SOCP relaxation. They will then be added as inequality constraints for the SOCP problem. With the new SOCP problem solved, a new solution is obtained and it is then examined for new cut generation. Modified with a set of cuts, the new SOCP problem will be solved again for next iteration, until no cuts will be generated or iteration limit is reached.

IV. Case Studies

All computations are conducted in MATLAB. MATPOWER [19] is used to find upper bound while CVX toolbox [20] and MOSEK 7.1 solver [21] are used to carry out convex programming problem solving. Cycle basis is identified using the algorithm in [18]. We have implemented the proposed LSE algorithm and the SDP separation algorithm of [1].

A. Three-bus test system

This test case comes from NESTA v0.4.0 archive [22]. The system is a three-bus system consisting of one cycle. After five iterations, five SDP cuts are added to the SOCP problem. The solution from SOCP relaxation of AC OPF is now SDP feasible. Table I lists the gap before adding SDP cuts and after adding SDP cuts. The objective function value of the minimization problem that generates SDP cuts stands for the distance from z_0 to the SDP feasible region $\mathcal{S}$ of a cycle in the cycle basis. Fig. 2(a) shows the distance for ten iterations. It can be seen that after a few iterations, the strengthened SOCP solution z_0 is in the SDP feasible region $\mathcal{S}$.

TABLE I
PERCENTAGE GAP OF A 3-BUS CASE

Test case	MATPOWER ($/h)	SOCP	SDP cuts
nesta_case3_lmbd	5812.6	1.67	1.27

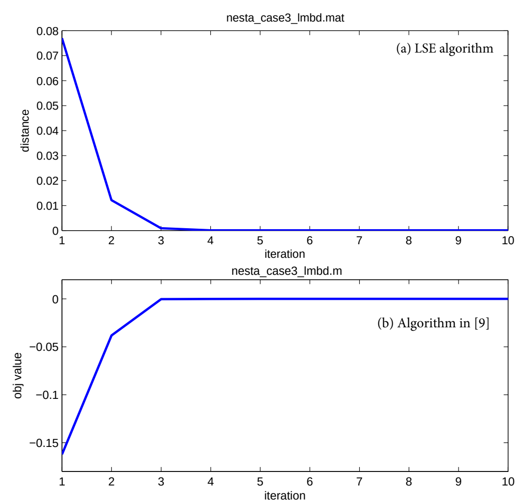

Fig. 2. Three-bus test case. This system has one cycle. (a) z_0 to $\mathcal{S}$ distance over iteration; (b) the value of (20) of the separation algorithm in [1].

As a comparison, we also presented the objective function of the minimization problem (20) used in [1] over iterations. We can see that the two algorithms have comparable performance.

B. Five-bus PJM system

This test case also comes from the NESTA archive. The topology is shown in Fig. 3. The five-bus system has two cycles in its cycle basis. The first cycle consists of buses 1, 2, 3, 4 and branch 1, 2, 4, 5. The second cycle consists of buses 1, 4, 5 and branches 2, 3 and 6. At each iteration, two cuts will be generated. After five iterations, the gap reduces from 14.48% to 9.08%.

TABLE II
PERCENTAGE GAP OF A 5-BUS CASE

Test case	MATPOWER ($/h)	SOCP	SDP cuts
nesta_case5_pjm	17552	14.48	9.08

Fig. 4(a) gives the plots of the distances of each cycle's z_0 to the SDP feasible region over iterations. Fig. 4(b) gives v^*

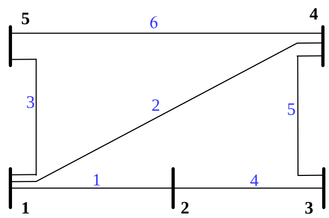

Fig. 3. Five-bus test case with two cycles in its cycle basis. Cycle 1: nodes 1, 2, 3, 4, branches 1, 2, 4, 5; Cycle 2: nodes 1, 4, 5, branches 2, 3, 6.

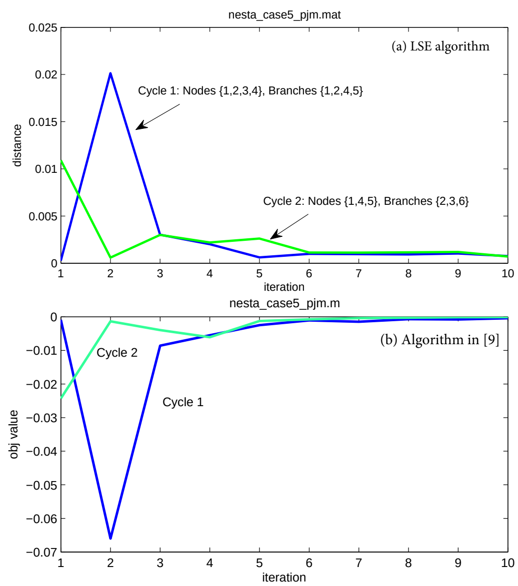

Fig. 4. Five-bus test case. This case has two cycles in the cycle basis. (a) z_0 to $\mathcal{S}$ distance over iteration; (b) the value of (20) of the separation algorithm in [1].

of (20) using the method in [1]. The performance of the two methods are comparable.

C. 30-bus test case

NESTA's IEEE 30-bus system has been reported in [1], [9] to have a large gap for SOCP relaxations and quadratic relaxations. This system has 12 cycles in its cycle basis. SOCP relaxation gives a set of decision variables. With the set, 12 minimum distance problems are solved and 12 SDP cuts are generated. The SCOP problem with 12 cuts is solved again and the gap is reduced significantly to 1.71%. After five iterations, the gap is reduced to 0.29%. Fig. 5(a) gives the 12 distances over iteration. Fig. 5(b) gives the 12 values over iteration using the method of [1]. For this instance, the method of [1] shows better performance: in five iterations, the objective functions of (20) become zero while the distances are still not zero.

D. Comparison of the LSE method, SDP duality method in [1]

The proposed LSE method is compared with the SDP separation method in [1]. Table III shows the optimality gap and computing time for the two methods. We can see that the performance of the two types of SDP cuts are overall similar. Computing time is also at the same scale.

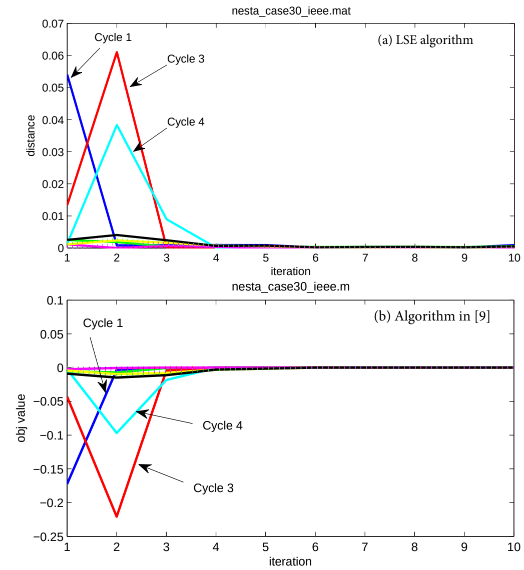

Fig. 5. 30-bus test case. This case has 12 cycles in the cycle basis. (a) z_0 to $\mathcal{S}$ distance over iteration; (b) the value of (20) of the separation algorithm in [1].

E. More Computational Results

For the 3-bus instance and the 5-bus instance, we have increased the iteration time to be 50 and examined the sum of distances related to cycles in the cycle basis. The sums of distances are shown in Fig. 6. We see exponential decrease in the sum of distances within 20 iterations. This indicates that SOCP solution may eventually be feasible for the SDP problem after a sequence of SDP cuts.

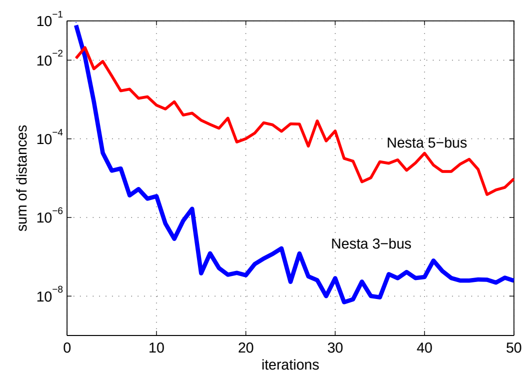

Fig. 6. Sum of distances over iterations of the 3-bus system and the 5-bus system.

A number of Nesta cases have been tested and the results are listed in Table IV. Gaps are compared for SOCP relaxation and SOCP relaxation with LSE-based SDP cuts.

Remarks: For all the case studies, we have shown the effectiveness of gap reduction after adding SDP cuts to the SOCP relaxation problems.

Computing Complexity

The computing time for SOCP relaxation problem solving and SOCP relaxation with SDP cuts is also given. All cases except the 2224-bus case were solved for five iterations for the enhanced SOCP relaxation with SDP cuts.

TABLE III

COMPARISON OF OPTIMALITY GAP PERCENTAGE AND RUNTIME RESULTS OF LSE METHOD, SDP DUALITY METHOD, AND THE COMBINATION OF TWO CUTS. A_l IS SYMMETRIC.

			SOCP with SDP cuts over iterations						runtime (s)	
case	UB ($/hr)	gap_SOCP	1	2	3	4	5	min_gap	SOCP	with Cuts
LSE method										
nesta_case3_lmbd.m	5812.64	1.67	1.31	1.27	1.27	1.27	1.27	1.27	0.54	3.68
nesta_case4_gs.m	156.43	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.32	2.53
nesta_case5_pjm.m	17551.89	14.48	14.19	11.45	9.17	9.12	9.08	9.08	0.41	3.58
nesta_case6_ww.m	3119.26	0.13	0.09	0.02	0.02	0.02	0.02	0.02	0.59	6.92
nesta_case9_wscc.m	5296.69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.56	4.15
nesta_case14_ieee.m	243.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	1.02	10.69
nesta_case30_as.m	800.14	0.05	0.01	0.00	0.02	0.00	0.00	0.00	2.04	21.08
nesta_case30_fsr.m	574.97	0.23	0.25	0.26	0.27	0.19	0.16	0.16	2.13	21.92
nesta_case30_ieee.m	203.30	15.93	1.96	0.41	0.40	0.13	0.10	0.10	2.09	20.84
nesta_case57_ieee.m	1140.69	0.05	0.01	0.01	0.01	0.01	0.01	0.01	3.72	52.38
nesta_case118_ieee.m	3696.72	2.10	1.74	1.54	1.52	1.52	1.52	1.52	8.42	101.52
nesta_case162_ieee_dtc.m	4142.39	2.47	2.33	2.30	2.30	2.29	2.28	2.28	12.88	198.38
nesta_case300_ieee.m	16539.60	1.09	0.64	0.68	0.69	0.69	0.69	0.66	20.21	233.28
Method in [1]										
nesta_case3_lmbd.m	5812.64	1.67	1.33	1.26	1.26	1.26	1.26	1.26	0.28	2.69
nesta_case4_gs.m	156.43	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.45	3.87
nesta_case5_pjm.m	17551.89	14.49	14.24	11.14	8.95	8.87	8.70	8.70	0.54	5.59
nesta_case6_ww.m	3119.26	0.13	0.07	0.02	0.04	0.04	0.02	0.02	0.69	11.02
nesta_case9_wscc.m	5296.69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.67	5.35
nesta_case14_ieee.m	243.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	1.14	15.15
nesta_case30_as.m	800.14	0.05	0.01	0.01	0.01	0.02	0.02	0.01	1.97	26.57
nesta_case30_fsr.m	574.97	0.23	0.24	0.22	0.13	0.12	0.10	0.10	1.97	26.31
nesta_case30_ieee.m	203.30	15.93	2.12	0.21	0.02	0.02	0.00	0.00	1.98	28.31
nesta_case57_ieee.m	1140.69	0.05	0.01	0.00	0.00	0.00	0.00	0.00	3.55	52.64
nesta_case118_ieee.m	3696.72	2.10	1.73	1.49	1.43	1.42	1.41	1.41	7.78	113.21
nesta_case162_ieee_dtc.m	4142.39	2.47	2.32	2.24	2.19	2.15	2.15	2.15	12.44	148.09
nesta_case300_ieee.m	16539.60	1.09	0.61	0.50	0.41	0.39	0.38	0.38	19.51	188.17

For the 1354-bus system, there are 357 cycles in the cycle basis. Once we have a solution from SOCP relaxation of the AC OPF, 357 SDP problems are solved to check if the portion of the solution that is related to a particular cycle is SDP feasible or not. The feasibility is indicated by the norm 2 distance of the SOCP solution to the SDP feasible region or the objective function of the minimization problem in (22). Shall we choose a very small tolerance number, e.g., 10^{-10}, 357 cuts will be generated and added to the set of constraints of the SOCP relaxation problem at every iteration.

The 2224-bus case is solved first by SOCP relaxation. The 581 SDP problems are solved to check of the portion of the solution related to certain cycle is SDP feasible or not. With 581 cuts generated, the SOCP relaxation problem has all cuts added and is solved again. This leads to a reduction of gap at 10 percent. Note that for large-scale cases, Mosek solver reported numerical inaccuracy issue.

Compared to SOCP relaxation, solving time for SOCP relaxation with SDP cuts is mainly dependent on the number of cycles and the size of the cycles in the cycle basis. For example, the computing time of nesta_case162_ieee_dtc is comparable with that of nesta_case300_ieee since both of them have about $110 \sim 120$ cycles in their cycle basis. Though the number of nodes in the 300-bus case is about twice of that in the 161-bus system, the computing time just has a slight increase.

V. Conclusion

This paper presents an LSE-based approach to find affine inequalities for SOCP relaxation of AC OPF. The affine inequalities serve as SDP cuts to reduce the feasible region and get rid of SOCP solutions outside of the feasible region of SDP relaxation. Least square estimation-based SDP cuts have been demonstrated to be effective to exclude infeasible solutions and enhance SOCP relaxation of OPF. This method has been tested on variety of cases to demonstrate its effectiveness.

Ackledgement

The authors would like to acknowledge Dr. Carleton Coffrin for providing the NESTA test cases. The authors wish to acknowledge anonymous reviewers for providing insightful comments.

References

[1] B. Kocuk, S. S. Dey, and X. A. Sun, "Strong socp relaxations for the optimal power flow problem," *Operations Research*, vol. 64, no. 6, pp. 1177–1196, 2016.

[2] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, "Semidefinite programming for optimal power flow problems," *International Journal of Electrical Power & Energy Systems*, vol. 30, no. 6, pp. 383–392, 2008.

[3] J. Lavaei and S. H. Low, "Zero duality gap in optimal power flow problem," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 92–107, 2012.

[4] D. K. Molzahn and I. A. Hiskens, "Moment-based relaxation of the optimal power flow problem," in *Power Systems Computation Conference (PSCC), 2014*. IEEE, 2014, pp. 1–7.

[5] C. Josz, J. Maeght, P. Panciatici, and J. C. Gilbert, "Application of the moment-sos approach to global optimization of the opf problem," *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 463–470, 2015.

[6] B. Ghaddar, J. Marecek, and M. Mevissen, "Optimal power flow as a polynomial optimization problem," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 539–546, 2016.

[7] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *IEEE Transactions on Automatic Control*, vol. 60, no. 1, pp. 72–87, 2015.

TABLE IV

QUALITY AND RUNTIME RESULTS OF AC POWER FLOW RELAXATIONS. FIVE ITERATIONS WERE CONDUCTED FOR THE ENHANCED SOCP WITH SDP CUTS. UB -UPPER BOUND COMPUTED FROM MATPOWER. UNSYMMETRICAL A_lS ARE USED.

case	UB ($/hr)	Optimality gap(%) SOCP	Optimality gap(%) SOCP_SDP	Runtime (seconds) SOCP	Runtime (seconds) SOCP_SDP	No of cycles
Typical Operating Conditions						
nesta_case3_lmbd	5812.64	1.67	1.27	0.47	3.21	1
nesta_case4_gs	156.43	0.00	0.00	0.51	3.51	1
nesta_case5_pjm	17551.89	14.48	9.08	0.57	4.56	2
nesta_case6_ww	3119.26	0.14	0.02	0.74	7.80	6
nesta_case9_wscc	5296.69	0.00	0.00	0.71	4.98	1
nesta_case14_ieee	243.00	0.10	0.00	1.16	11.49	7
nesta_case30_as	800.14	0.06	0.01	2.13	20.68	12
nesta_case30_fsr	574.97	0.26	0.20	2.18	20.88	12
nesta_case30_ieee	203.30	15.93	0.29	2.13	20.75	12
nesta_case39_epri	96447.55	0.04	0.06	2.44	20.33	8
nesta_case57_ieee	1140.69	0.05	0.00	3.78	51.98	22
nesta_case118_ieee	3669.70	2.09	1.51	8.53	100.31	62
nesta_case162_ieee_dtc	4142.39	2.47	2.28	12.60	192.19	119
nesta_case300_ieee	16539.60	1.04	0.64	19.95	225.78	110
nesta_case1354_pegase	74019.2	17.52*	12.41*	95.9	2944.36	357
nesta_case2224_edin	37893	63.35*	53.00*	231.41	694.29†	581
Congested Operating Conditions (API)						
nesta_case3_lmbd__api	367.74	9.41	9.34	0.55	3.47	1
nesta_case4_gs__api	767.27	0.78	0.20	0.51	3.66	1
nesta_case5_pjm__api	2998.54	0.44	0.18	0.60	4.85	2
nesta_case6_ww__api	236.66	0.30	0.04	0.76	7.84	6
nesta_case9_wscc__api	656.60	0.04	0.03	0.60	4.56	1
nesta_case14_ieee__api	318.82	0.07	0.00	1.10	11.62	7
nesta_case30_as__api	561.84	3.52	3.01	2.03	23.10	12
nesta_case30_fsr__api	347.87	42.23	41.82	2.04	22.97	12
nesta_case30_ieee__api	409.75	0.33	0.06	2.06	23.13	12
nesta_case39_epri__api	7468.40	3.93	2.67	2.35	21.69	8
nesta_case57_ieee__api	1424.93	0.06	0.01	3.82	57.83	22
nesta_case118_ieee__api	6299.81	41.63	40.84	8.81	106.55	62
nesta_case162_ieee_dtc__api	6026.84	0.71	0.53	12.13	185.60	119
nesta_case300_ieee__api	22753.19	1.01	0.58	19.31	222.48	110
Small Angle Difference Conditions (SAD)						
nesta_case3_lmbd__sad	5992.72	4.62	4.23	1.37	3.79	1
nesta_case4_gs__sad	324.02	51.72	51.72	0.33	2.63	1
nesta_case5_pjm__sad	26423.33	43.17	39.58	0.42	3.57	2
nesta_case6_ww__sad	3119.26	0.14	0.02	0.59	6.67	6
nesta_case9_wscc__sad	5590.09	5.25	5.24	0.55	4.00	1
nesta_case14_ieee__sad	243.00	0.10	0.00	0.99	10.16	7
nesta_case30_as__sad	878.02	8.92	8.88	1.93	19.37	12
nesta_case30_fsr__sad	575.39	0.35	0.27	1.97	19.41	12
nesta_case30_ieee__sad	203.30	15.92	0.09	1.93	19.41	12
nesta_case39_epri__sad	96515.91	0.11	0.11	2.24	19.11	8
nesta_case118_ieee__sad	3997.32	10.01	9.48	8.13	96.08	62
nesta_case162_ieee_dtc__sad	4192.13	3.58	3.40	12.07	185.71	119
nesta_case300_ieee__sad	16564.70	1.10	0.70	19.13	219.80	110

$\star$ - solver reported numerical accuracy warnings, —,† - one iteration.

[8] C. Coffrin and P. Van Hentenryck, "A linear-programming approximation of ac power flows," *INFORMS Journal on Computing*, vol. 26, no. 4, pp. 718–734, 2014.

[9] C. Coffrin, H. L. Hijazi, and P. Van Hentenryck, "The qc relaxation: A theoretical and computational study on optimal power flow," *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 3008–3018, 2016.

[10] R. A. Jabr, "Radial distribution load flow using conic programming," *Power Systems, IEEE Transactions on*, vol. 21, no. 3, pp. 1458–1459, 2006.

[11] H. Hijazi, C. Coffrin, and P. Van Hentenryck, "Polynomial sdp cuts for optimal power flow," *arXiv preprint arXiv:1510.08107*, 2015.

[12] S. Boyd and L. Vandenberghe, "Localization and cutting-plane methods," *From Stanford EE 364b lecture notes*, 2011.

[13] A. J. Wood and B. F. Wollenberg, *Power generation, operation, and control*. John Wiley & Sons, 2012.

[14] S. H. Low, "Convex relaxation of optimal power flow, part ii: Exactness," *arXiv preprint arXiv:1405.0814*, 2014.

[15] R. A. Jabr, "A conic quadratic format for the load flow equations of meshed networks," *Power Systems, IEEE Transactions on*, vol. 22, no. 4, pp. 2285–2286, 2007.

[16] B. Gärtner and J. Matousek, *Approximation algorithms and semidefinite programming*. Springer Science & Business Media, 2012.

[17] E. De Klerk, *Aspects of semidefinite programming: interior point algorithms and selected applications*. Springer Science & Business Media, 2006, vol. 65.

[18] K. Mehlhorn and D. Michail, "Implementing minimum cycle basis algorithms," *Journal of Experimental Algorithmics (JEA)*, vol. 11, pp. 2–5, 2007.

[19] 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," *Power Systems, IEEE Transactions on*, vol. 26, no. 1, pp. 12–19, 2011.

[20] M. Grant, S. Boyd, and Y. Ye, "Cvx: Matlab software for disciplined convex programming," 2008.

[21] A. Mosek, "The mosek optimization toolbox for matlab manual," *Version 7.1 (Revision 28)*, 2015.

[22] C. Coffrin, D. Gordon, and P. Scott. (2014) NESTA, the NICTA energy system test case archive. [Online]. Available: http://arxiv.org/abs/1411.0359