Certified and accurate SDP bounds for the ACOPF problem

Antoine Oustry, Claudia d'Ambrosio, Leo Liberti, Manuel Ruiz

Certified and accurate SDP bounds for the ACOPF problem

Antoine Oustry*†, Claudia D'Ambrosio*, Leo Liberti* and Manuel Ruiz‡
* LIX CNRS, École polytechnique
Institut Polytechnique de Paris, 91128, Palaiseau, France
† École des Ponts, 77455 Marne-La-Vallée
‡ R&D department, Réseau de transport d'électricité, 92073 La Défense, France

***Abstract*—We propose a new method for improving the bound tightness of the popular semidefinite programming (SDP) relaxation for the ACOPF introduced in [1], [2]. First, we reformulate the ACOPF Lagrangian dual as an unconstrained concave maximization problem with a clique decomposition induced sparse structure. We prove that this new formulation has the same optimal value as the SDP relaxation. We then use the solution of the SDP relaxation as a starting point for a tailored structure-aware bundle method. This post-processing technique significantly improves the tightness of the SDP bounds computed by the state-of-the-art solver MOSEK, as shown by our computational experiments on large-scale instances from PGLib-OPF v21.07. For ten of the tested instances, our post-processing decreases by more than 50% the optimality gap obtained with MOSEK.**

***Index Terms*—AC Optimal Power Flow, Complex semidefinite programming, Nonsmooth optimization.**

I. Introduction

The ACOPF problem is one of the most fundamental optimization problems for the management of electric power systems. This is a cost minimization problem in an AC network, where the generators should supply electricity to loads under generator and line constraints. Due to the nonconvexity of power flow equations, the ACOPF is a difficult optimization problem both in theory [3] and in practice [4]. Although local optimization solvers are able to find good solutions, obtaining a feasible point with a global optimality certificate is still a challenge for national-scale instances. In this quest for global optimization, convex relaxations are powerful tools to efficiently compute lower bounds on the ACOPF's value. A review of various relaxation techniques for the ACOPF problem is available in [2]. Among them, semidefinite programming (SDP) is of strong interest for both theoretical and practical reasons: indeed, the semidefinite relaxation (SDR) is the ACOPF's Lagrangian bidual [1] and provides tight lower bounds [5]. The main drawback of the standard SDR is that it involves a dense $n \times n$ complex matrix as a decision variable, where n is the number of buses. This becomes intractable as soon as n exceeds magnitudes of around 10^3.

To overcome this computational burden, state-of-the-art approaches [6], [7] exploit the sparse structure of the power grid by using a clique decomposition technique. Thanks to a semidefinite completion theorem [8] for matrices with chordal sparsity pattern, it is possible to find an equivalent formulation for the SDR, with many small semidefinite blocks instead of a single and large $n \times n$ matrix. Each of these blocks corresponds to a maximal clique of the chordal sparsity pattern. This clique-based SDR can be solved with a symmetric interior point (IP) solver [9], [10], with a non-symmetric IP solver [11] or with a first order method like the Alternating Direction Method of Multipliers (ADMM) [12]–[14]. Eltved et al. [6] report the solution of the clique-based ACOPF's SDR for test cases with up to $82,000$ buses, in a few hours.

Despite these advances, the clique-based SDR remains difficult to solve for large-scale instances: numerical instabilities may arise when solving this convex optimization problem. In [5] and [15], the authors report numerical instabilities with the commercial IP solver MOSEK [9], which raises a warning for many tested instances. In [6], the authors report that the well-known academic IP solver SeDuMi fails to solve the clique-based SDR in more than 50% of the test cases; and that the ADMM-based solver CDCS often terminates with sizable dual residuals.

Two categories of troubles may arise due to numerical difficulties in solving the SDR. First, they may limit the accuracy of the calculation of the SDR value. Second, obtaining a solution with non-zero primal and/or dual feasibility errors implies that the calculated relaxation value is not *certified* [16]. In this case, one obtains an *approximated* value of the relaxation without knowing if it is an *exact* lower bound on the ACOPF's value.

Main contributions This paper tackles both aforementioned numerical issues with an original approach. For this purpose, we introduce a new formulation for the Lagrangian dual of the ACOPF, whose value equals the value of the SDR. Our new formulation is a concave maximization problem with the following interesting properties: (a) it is unconstrained (b) the objective function is *partially separable*. Based on this formulation, we present how to obtain a certified lower bound from any dual vector, whether feasible or not in the *classical* dual SDR. We solve our new formulation with a structure-exploiting polyhedral bundle method. We use this algorithm as a *post-processing step*, after solving the clique-based SDR with the commercial IP solver MOSEK [9], which is the state-of-the-art solver for ACOPF's SDR according to [2], [6]. Our numerical experiments on instances from PGLib-OPF v21.07 [17] show that this post-processing considerably improves the tightness of the dual bounds. For 13 cases, the post-processing increases the certified dual bound by more than 0.1%. A more accurate dual bound yields a lower optimality gap: for 10 cases, the gap is reduced by more than 50%; for 5 cases among them, the gap is reduced by more than 90%.

Mathematical notations:

- x^+, x^- : positive and negative part of the real number x,
- $\mathsf{Re}(z), \mathsf{Im}(z)$, $z^\star$, $|z|$: Real part, imaginary part, conjugate and magnitude of complex number $z = \mathsf{Re}(z) + \mathbf{i}\,\mathsf{Im}(z)$,
- $\mathbb{H}_n \subset \mathbb{C}^{n\times n}$: the set of Hermitian matrices of size n, subset of $\mathbb{C}^{n\times n}$, the vector space of square complex matrices of size n,
- $\mathbb{H}_n(S) \subset \mathbb{H}_n$, for $S \subset \{1,\dots,n\}$: the set of Hermitian matrices of size n, whose non-zero coefficients all belong to the principal submatrix associated with indices $k \in S$,
- M^{H} : Hermitian transpose of the matrix $M \in \mathbb{C}^{n\times n}$,
- $\mathcal{H}(M) := \frac{1}{2}(M + M^{\mathsf{H}})$ and $\mathcal{Z}(M) := \frac{1}{2}(M - M^{\mathsf{H}})$,
- $\mathsf{Tr}(M)$: Trace of the matrix M,
- $\langle M, N\rangle = \mathsf{Tr}(M^{\mathsf{H}}N)$: Frobenius product between complex matrices $M, N \in \mathbb{C}^{n\times n}$,
- $\lambda_{\min}(H) \in \mathbb{R}$: Minimum eigenvalue of the Hermitian matrix $H \in \mathbb{H}_n$,
- $H \succeq 0$: the matrix H is positive semidefinite (PSD),
- E_{ba} : the matrix of the canonical basis of $\mathbb{C}^{n\times n}$, associated with the pair of indices (b, a).

II. The ACOPF, the ACOPF's SDR and their dual

This section presents the context of this work, i.e. the ACOPF problem with current limits, its SDR, and the state-of-the-art approach to solve it, based on clique decomposition.

A. Problem formulation

A power grid is as a network of buses interconnected by lines. It is modelled [18] as an oriented multi-graph $\mathcal{N} = (\mathcal{B}, \mathcal{L})$ with sizes $n = |\mathcal{B}|$ and $l = |\mathcal{L}|$. The orientation of a line is an arbitrary but necessary convention to uniquely define the admittance matrix (see below) modelling this line. A line $\ell \in \mathcal{L}$ is described by a triplet (b, a, h) s.t. $b \in \mathcal{B}$ is the "'from" bus (denoted by f), $a \in \mathcal{B}$ is the "to" bus (denoted by t) and $h \in \mathbb{N}$ is an index, which is non-null in case there are several parallel lines between b and a. Electricity generating units are located in several buses in the network. We denote by $\mathcal{G}_b$ the set of generators located at bus $b \in \mathcal{B}$. The set of all generators is $\mathcal{G} := \cup_{b\in\mathcal{B}}\, \mathcal{G}_b$, whose cardinality is denoted by $m = |\mathcal{G}|$. For any generator $g \in \mathcal{G}$, we denote by $b(g) \in \mathcal{B}$ the bus where the generator g is located. For each $g \in \mathcal{G}$, the generated power $S_g \in \mathbb{C}$ is a decision variable subject to

$$\underline{P}_g \le \mathsf{Re}(S_g) \le \overline{P}_g, \tag{1}$$

$$\underline{Q}_g \le \mathsf{Im}(S_g) \le \overline{Q}_g, \tag{2}$$

for parameters $\underline{P}_g, \overline{P}_g, \underline{Q}_g, \overline{Q}_g \in \mathbb{R}$. For brevity, we define $\underline{S}_g = \underline{P}_g + \mathbf{i}\,\underline{Q}_g$, $\overline{S}_g = \overline{P}_g + \mathbf{i}\,\overline{Q}_g$ and we write (1)-(2) as $S_g \in \left[\underline{S}_g, \overline{S}_g\right]_{\mathbb{C}}$. The generators' cost functions are traditionally assumed to depend only on the active power and to be quadratic convex [18]. Consistently with this latter assumption and for all $g \in \mathcal{G}$, we define $c_{0g} \in \mathbb{R}$, $c_{1g} \in \mathbb{R}$ and $c_{2g} \in \mathbb{R}_+$, s.t. the cost associated with active power p is $c_{0g} + c_{1g}p + c_{2g}p^2$. Since the offsets c_{0g} play no role in the optimization, we assume without loss of generality that $c_{0g} = 0$, for better readability. However, in the numerical experiments, we take these offsets into account. We denote by $\mathcal{G}^1$ the set of generators g with purely linear cost ($c_{2g} = 0$), and $\mathcal{G}^2 := \mathcal{G} \setminus \mathcal{G}^1$, which should not be confused with the set $\mathcal{G}_b$ of generators attached to bus b. For each bus $b \in \mathcal{B}$, the voltage $V_b \in \mathbb{C}$ is a decision variable subject to

$$\underline{V}_b \le |V_b| \le \overline{V}_b, \tag{3}$$

for parameters $\underline{V}_b, \overline{V}_b \in \mathbb{R}_+$. The other parameters related to bus b are a shunt admittance $Y_b^s \in \mathbb{C}$ and a load $S_b^{\mathsf{load}} = P_b^{\mathsf{load}} + \mathbf{i}Q_b^{\mathsf{load}}$ with $P_b^{\mathsf{load}}, Q_b^{\mathsf{load}} \in \mathbb{R}$. So as to model the power conservation at each bus, we also need to introduce quantities describing the electrical characteristics of the lines. For each line $\ell = (b, a, h) \in \mathcal{L}$, we know its (non-symmetric) admittance matrix $Y_\ell \in \mathbb{C}^{2\times 2}$, which follows from a Π-line model of the line [18]. The coefficients of this matrix are denoted by $Y_\ell^{\mathsf{ff}}, Y_\ell^{\mathsf{ft}}, Y_\ell^{\mathsf{tf}}, Y_\ell^{\mathsf{tt}}$. With this notation, we define

$$\begin{aligned} M_b := Y_b^s E_{bb} &+ \sum_{\substack{\ell\in\mathcal{L}\\ \ell=(b,a,h)}} Y_\ell^{\mathsf{ff}} E_{bb} + Y_\ell^{\mathsf{ft}} E_{ba} \\ &+ \sum_{\substack{\ell\in\mathcal{L}\\ \ell=(a,b,h)}} Y_\ell^{\mathsf{tt}} E_{bb} + Y_\ell^{\mathsf{tf}} E_{ba}, \end{aligned}$$

for each bus $b \in \mathcal{B}$. Then, the equation modelling the power conservation at $b \in \mathcal{B}$ is

$$\sum_{g\in\mathcal{G}_b} S_g = S_b^{\mathsf{load}} + \left\langle M_b, VV^{\mathsf{H}}\right\rangle. \tag{4}$$

Finally, we consider current flow limits on lines. For any line $\ell = (b, a, h) \in \mathcal{L}$, given a limit $\overline{I}_\ell \in \mathbb{R}_+$, we have

$$|Y_\ell^{\mathsf{ff}} V_b + Y_\ell^{\mathsf{ft}} V_a| \le \overline{I}_\ell, \tag{5}$$

$$|Y_\ell^{\mathsf{tt}} V_a + Y_\ell^{\mathsf{tf}} V_b| \le \overline{I}_\ell. \tag{6}$$

We treat this type of branch flow limits because they are the ones that Réseau de Transport d'Electricité, the French Transmission System Operator, deals with in its daily operations. Nevertheless, our approach could easily be transposed to other types of constraints such as apparent power flow limits. In summary, the ACOPF problem with current flow constraints is the following non-convex optimization problem:

$$\begin{aligned} \min \quad & \sum_{g\in\mathcal{G}} c_{1g}\,\mathsf{Re}(S_g) + c_{2g}\,\mathsf{Re}(S_g)^2 \\ \text{s.t.} \quad & (1) - (6) \\ & V \in \mathbb{C}^n,\ S \in \mathbb{C}^m. \end{aligned} \tag{OPF}$$

For any line $\ell = (b, a, h) \in \mathcal{L}$, we also introduce the matrices $N_{\ell\mathsf{f}} := |Y_\ell^{\mathsf{ff}}|^2 E_{bb} + Y_\ell^{\mathsf{ff}} (Y_\ell^{\mathsf{ft}})^\star E_{ab} + (Y_\ell^{\mathsf{ff}})^\star Y_\ell^{\mathsf{ft}} E_{ba} + |Y_\ell^{\mathsf{ft}}|^2 E_{aa}$ and $N_{\ell\mathsf{t}} := |Y_\ell^{\mathsf{tt}}|^2 E_{aa} + Y_\ell^{\mathsf{tt}} (Y_\ell^{\mathsf{tf}})^\star E_{ba} + (Y_\ell^{\mathsf{tt}})^\star Y_\ell^{\mathsf{tf}} E_{ab} + |Y_\ell^{\mathsf{tf}}|^2 E_{bb}$. Hence, constraints (5) and (6) reads $\left\langle N_{\ell\mathsf{f}}, VV^{\mathsf{H}}\right\rangle \le \overline{I}_\ell^2$ and $\left\langle N_{\ell\mathsf{t}}, VV^{\mathsf{H}}\right\rangle \le \overline{I}_\ell^2$.

B. The SDR and the clique-based SDR

The SDR, also known as “rank relaxation”, is classically derived by replacing the rank-one matrix VV^{H} in (OPF) by a Hermitian and PSD matrix W of unspecified rank. Hence, the SDR of (OPF) is the following convex optimization problem

$$\begin{array}{llll}\min & \sum_{g\in\mathcal{G}} c_{1g}\,\mathsf{Re}(S_g) + c_{2g}\,\mathsf{Re}(S_g)^2 & & \\ \text{s.t.} & \underline{V}_b^2 \le \langle E_{bb}, W\rangle \le \overline{V}_b^2 & \forall b\in\mathcal{B} & (\mathsf{SDR}) \\ & \sum_{g\in\mathcal{G}_b} S_g = S_b^{\mathsf{load}} + \langle M_b, W\rangle & \forall b\in\mathcal{B} & \\ & \langle N_{\ell\mathsf{s}}, W\rangle \le \overline{I_\ell}^2 & \forall(\ell,\mathsf{s})\in\mathcal{L}\times\{\mathsf{f},\mathsf{t}\} & \\ & W\succeq 0 & & \\ & W\in\mathbb{H}_n,\ S\in\prod_{g\in\mathcal{G}}[\underline{S}_g,\overline{S}_g]_{\mathbb{C}} & & \end{array}$$

This formulation becomes intractable for large-scale instances [2]. A standard approach to avoid this computational burden consists in introducing a chordal extension of the undirected graph induced by $\mathcal{N}$, and a *clique tree* $\mathcal{T}$ of this chordal extension [19]. For any node $k\in\mathcal{T}$, the set $\mathcal{B}_k\subset\mathcal{B}$ is the corresponding clique of the chordal extension, and n_k is its cardinality. Choosing an arbitrary root $r\in\mathcal{T}$, we designate by $\mathsf{p}(k)\in\mathcal{T}$ the parent node of any node $k\in\mathcal{T}$, with the convention $\mathsf{p}(r)=r$. We define the set $\mathcal{J}_k := \{(b,a)\in(\mathcal{B}_k\cap\mathcal{B}_{\mathsf{p}(k)})^2 : b\le a\}$. For any $b\in\mathcal{B}$, we introduce some matrices $M_{bk}\in\mathbb{H}_n(\mathcal{B}_k)$ for $k\in\mathcal{T}$, so as to write $M_b = \sum_{k\in\mathcal{T}|b\in\mathcal{B}_k} M_{bk}$. We underline that the non-zero coefficients in matrix M_{bk} all belong to the principal submatrix associated with clique $\mathcal{B}_k$. For any line $\ell=(b,a,h)\in\mathcal{L}$, we define $k(\ell)\in\mathcal{T}$ s.t. $\{b,a\}\subset\mathcal{B}_k$. With this notation the clique-based SDR, denoted by (cSDR), reads

$$\begin{array}{lll}\min & \sum_{g\in\mathcal{G}} c_{1g}\,\mathsf{Re}(S_g) + c_{2g}\,\mathsf{Re}(S_g)^2 & \\ \text{s.t.} & \underline{V}_b^2 \le \langle E_{bb}, W_k\rangle \le \overline{V}_b^2 & \forall k\in\mathcal{T},\forall b\in\mathcal{B}_k \\ & \sum_{g\in\mathcal{G}_b} S_g = S_b^{\mathsf{load}} + \sum_{k\in\mathcal{T}|b\in\mathcal{B}_k}\langle M_{bk}, W_k\rangle & \forall b\in\mathcal{B} \\ & \langle N_{\ell\mathsf{s}}, W_{k(\ell)}\rangle \le \overline{I_\ell}^2 & \forall(\ell,\mathsf{s})\in\mathcal{L}\times\{\mathsf{f},\mathsf{t}\} \\ & \langle E_{ba}, W_k\rangle = \langle E_{ba}, W_{\mathsf{p}(k)}\rangle & \forall k\in\mathcal{T},\forall(b,a)\in\mathcal{J}_k \\ & W_k\succeq 0 & \forall k\in\mathcal{T} \\ & W_k\in\mathbb{H}_n(\mathcal{B}_k) & \forall k\in\mathcal{T} \\ & S\in\prod_{g\in\mathcal{G}}[\underline{S}_g,\overline{S}_g]_{\mathbb{C}}. & \end{array}$$

The advantage of formulation (cSDR) is that the total number of non-zero coefficients in the matrices $(W_k)_{k\in\mathcal{T}}$ is $\sum_{k\in\mathcal{T}} n_k^2$, whereas the matrix variable W in (SDR) involves n^2 non-zero coefficients. If the cliques are of limited size ($n_k\ll n$), then this decomposition is particularly relevant. This is often the case in ACOPF instances, since power grids are known to have low tree-width [2]. Since $|\mathcal{T}|\le n$ by property of chordal graphs, we deduce that for graphs with bounded tree-width, $\sum_{k\in\mathcal{T}} n_k^2$ scales in $O(n)$ rather than $O(n^2)$.

Proposition 1. *The SDP problems* (SDR) *and* (cSDR) *share the same value.*

Proof. This follows from the PSD completion theorem in [8], for Hermitian matrices with chordal sparsity pattern. ☐

In the following, we will use the integers $K_1 := \sum_{k\in\mathcal{T}} n_k$, $K_2 := \sum_{k\in\mathcal{T}} |\mathcal{J}_k|$ and $N := K_1 + 2(n+l+K_2)$.

C. The SDP dual of the clique-based SDR

The dual formulation of (cSDR) includes Linear Matrix Inequalities (LMI) involving a family of $\mathbb{R}$-linear matrix operators $\mathcal{A}_k:\mathbb{R}^N\to\mathbb{H}_n(\mathcal{B}_k)$ for $k\in\mathcal{T}$. The operator $\mathcal{A}_k$ is defined s.t. for all $\theta=(\alpha,\beta,\gamma,\eta,\nu,\mu)\in\mathbb{R}^N$,

$$\begin{aligned}\mathcal{A}_k(\theta) := &\sum_{b\in\mathcal{B}_k} \alpha_{bk}E_{bb} + \beta_b\mathcal{H}(M_{bk}) + \mathbf{i}\gamma_b\mathcal{Z}(M_{bk}) \\ &+ \sum_{(\ell,\mathsf{s})\in\mathcal{L}\times\{\mathsf{f},\mathsf{t}\}} \eta_{\ell\mathsf{s}}N_{\ell\mathsf{s}} + \sum_{(b,a)\in\mathcal{J}_k} \nu_{bak}\mathcal{H}(E_{ba}) + \mathbf{i}\mu_{bak}\mathcal{Z}(E_{ba}) \\ &- \left(\sum_{d\in\mathcal{C}|\mathsf{p}(d)=k}\ \sum_{(b,a)\in\mathcal{J}_d} \nu_{bad}\mathcal{H}(E_{ba}) + \mathbf{i}\mu_{bad}\mathcal{Z}(E_{ba})\right).\end{aligned}$$

We point out that the Hermitian part and anti-Hermitian part operators $\mathcal{H}$ and $\mathcal{Z}$, defined in Introduction, appear when projecting the complex equalities of the primal problem on their real and imaginary parts. Having introduced the operators $\mathcal{A}_k$ for $k\in\mathcal{T}$, the dual of (cSDR), which we denote by (dualcSDR), then reads

$$\begin{array}{lll}\max & \sum_{k\in\mathcal{T},\,b\in\mathcal{B}_k} \underline{V}_b^2\underline{\alpha}_{bk} - \overline{V}_b^2\overline{\alpha}_{bk} + \sum_{b\in\mathcal{B}} P_b^{\mathsf{load}}\beta_b + Q_b^{\mathsf{load}}\gamma_b & \\ & +\sum_{g\in\mathcal{G}} \underline{P}_g\underline{y}_g - \overline{P}_g\overline{y}_g + \underline{Q}_g\underline{z}_g - \overline{Q}_g\overline{z}_g & \\ & -\sum_{g\in\mathcal{G}^2} \frac{(c_{1g}+\overline{y}_g-\underline{y}_g-\beta_{b(g)})^2}{4c_{2g}} - \sum_{\ell\in\mathcal{L}} \overline{I_\ell}^2(\eta_{f\ell}+\eta_{t\ell}) & \\ \text{s.t.} & \mathcal{A}_k(\overline{\alpha}-\underline{\alpha},\beta,\gamma,\eta,\nu,\mu)\succeq 0 & \forall k\in\mathcal{T} \\ & \overline{y}_g - \underline{y}_g = \beta_{b(g)} - c_{1g} & \forall g\in\mathcal{G}^1 \\ & \overline{z}_g - \underline{z}_g = \gamma_{b(g)} & \forall g\in\mathcal{G} \\ & (\underline{y},\overline{y},\underline{z},\overline{z})\in\mathbb{R}_+^{4m} & \\ & (\underline{\alpha},\overline{\alpha})\in\mathbb{R}_+^{2K_1} & \\ & (\beta,\gamma,\eta,\nu,\mu)\in\mathbb{R}^{2(n+l+K_2)}. & \end{array}$$

III. New formulation for the dual ACOPF

This section introduces the central element of our approach, i.e. a new formulation for the dual problem of the ACOPF.

A. Some concave functions of interest

In order to reformulate (dualcSDR) as an unconstrained problem, we introduce some basic functions that are terms in the objective function of (dualcSDR) or penalizations of the constraints:

- for each generator $g\in\mathcal{G}^1$, we introduce the concave functions p_g, q_g s.t. for all $x\in\mathbb{R}$,
$$\begin{aligned}p_g(x) &:= \underline{P}_g(x-c_{1g})^- - \overline{P}_g(x-c_{1g})^+,\\ q_g(x) &:= \underline{Q}_g\,x^- - \overline{Q}_g\,x^+,\end{aligned}$$
- for each generator $g\in\mathcal{G}^2$, we introduce the concave functions p_g, q_g s.t. for all $x\in\mathbb{R}$,
$$p_g(x) := \begin{cases} -\underline{P}_g(x-c_{1g}-c_{2g}\underline{P}_g) & \text{if } x-c_{1g}\le 2c_{2g}\underline{P}_g,\\ -\frac{(x-c_{1g})^2}{4c_{2g}} & \text{if } x-c_{1g}\in[\,2c_{2g}\underline{P}_g, 2c_{2g}\overline{P}_g\,],\\ -\overline{P}_g(x-c_{1g}-c_{2g}\overline{P}_g) & \text{if } x-c_{1g}\ge 2c_{2g}\overline{P}_g,\end{cases}$$
$$q_g(x) := \underline{Q}_g\,x^- - \overline{Q}_g\,x^+,$$

- for each bus $b \in \mathcal{B}$, we define the concave functions v_b, p_b, q_b s.t. for all $x \in \mathbb{R}$,
$$v_b(x) := \underline{V}_b^2\, x^- - \overline{V}_b^2\, x^+,$$
$$p_b(x) := P_b^{\mathsf{load}}\, x + \sum_{g \in \mathcal{G}_b} p_g(x),$$
$$q_b(x) := Q_b^{\mathsf{load}}\, x + \sum_{g \in \mathcal{G}_b} q_g(x),$$
- for each line $\ell \in \mathcal{L}$, we define the function h_ℓ s.t. for all $x \in \mathbb{R}$, $h_\ell(x) := -\overline{I}_\ell^2\, x^+$.

Finally, we introduce for each $k \in \mathcal{T}$ the concave multivariate function f_k, s.t. for all $\theta = (\alpha, \beta, \gamma, \eta, \nu, \mu) \in \mathbb{R}^N$,
$$f_k(\theta) := \min \left\{ \rho_k\, \lambda_{\min}(\mathcal{A}_k(\theta)), 0 \right\},$$
where $\rho_k := \sum_{b \in \mathcal{B}_k} \overline{V}_b^2$. We underline that the linear operator $\mathcal{A}_k$ only depends on variables related to the clique index k; hence, so does function f_k (see Table I).

B. A sparse and unconstrained dual formulation

Our new formulation for the ACOPF's dual reads
$$\begin{aligned} \max \quad & \sum_{k \in \mathcal{T}} f_k(\alpha, \beta, \gamma, \eta, \nu, \mu) + \sum_{k \in \mathcal{T},\, b \in \mathcal{B}_k} v_b(\alpha_{bk}) \\ & + \sum_{b \in \mathcal{B}} p_b(\beta_b) + \sum_{b \in \mathcal{B}} q_b(\gamma_b) + \sum_{(\ell,\mathsf{s}) \in \mathcal{L} \times \{\mathsf{f},\mathsf{t}\}} h_\ell(\eta_{\ell\mathsf{s}}) \\ \text{s.t.} \quad & (\alpha, \beta, \gamma, \eta, \nu, \mu) \in \mathbb{R}^N \end{aligned} \tag{dualOPF}$$

We denote the objective function of this unconstrained concave maximization by $F : \mathbb{R}^N \to \mathbb{R}$. Since it is a sum of functions, each involving just one or a limited number of variables, problem (dualOPF) is said partially separable.

Theorem 1. *The problem* (dualOPF) *has the same value as problem* (SDR).

Proof. We define $\mathbb{X} := \{(W_k)_{k \in \mathcal{T}} \mid \forall k \in \mathcal{T},\ (W_k \in \mathbb{H}_n(\mathcal{B}_k)) \wedge (W_k \succeq 0) \wedge (\mathsf{Tr}(W_k) \leq \rho_k)\}$. We underline that the constraint $\mathsf{Tr}(W_k) \leq \rho_k$ is redundant in (cSDR) since it follows from the constraints $\forall b \in \mathcal{B}_k, \langle E_{bb}, W_k \rangle \leq \overline{V}_b^2$, but it plays a role in the derivation of the dual formulation. We define the Lagrangian $\mathcal{L}$ s.t. for any primal vectors $S \in \mathbb{C}^m$, $\mathbf{W} \in \mathbb{X}$, $u^1 \in \mathbb{R}^{K_1}$ and $u^2 \in \mathbb{R}^{2l}$ and for any dual vector $\theta = (\alpha, \beta, \gamma, \eta, \nu, \mu) \in \mathbb{R}^N$,
$\mathcal{L}(S, \mathbf{W}, u^1, u^2, \theta) := \sum_{g \in \mathcal{G}} c_{1g}\ \mathsf{Re}(S_g) + c_{2g}\ \mathsf{Re}(S_g)^2 + \sum_{k \in \mathcal{T}, b \in \mathcal{B}_k} \alpha_{bk}\big(\langle E_{bb}, W_k\rangle - u^1_{bk}\big) + \sum_{b \in \mathcal{B}} \beta_b\big(P_b^{\mathsf{load}} + \langle \mathcal{H}(M_{bk}), W_k\rangle - \sum_{g \in \mathcal{G}_b} \mathsf{Re}(S_g)\big) + \sum_{b \in \mathcal{B}} \gamma_b\big(Q_b^{\mathsf{load}} + \langle \mathbf{i}\mathcal{Z}(M_{bk}), W_k\rangle - \sum_{g \in \mathcal{G}_b} \mathsf{Im}(S_g)\big) + \sum_{(\ell,\mathsf{s}) \in \mathcal{L} \times \{\mathsf{f},\mathsf{t}\}} \eta_{\ell\mathsf{s}}\big(\big\langle N_{\ell\mathsf{s}}, W_{k(\ell)}\big\rangle - u^2_{\ell\mathsf{s}}\big) + \sum_{k \in \mathcal{T}, (b,a) \in \mathcal{J}_k} \nu_{bak}\big(\langle \mathcal{H}(E_{ba}), W_k\rangle - \big\langle \mathcal{H}(E_{ba}), W_{\mathsf{p}(k)}\big\rangle\big) + \sum_{k \in \mathcal{T}, (b,a) \in \mathcal{J}_k} \mu_{bak}\big(\langle \mathbf{i}\mathcal{Z}(E_{ba}), W_k\rangle - \big\langle \mathbf{i}\mathcal{Z}(E_{ba}), W_{\mathsf{p}(k)}\big\rangle\big)$.
We leave it to the reader to notice that problem (cSDR) can be written as
$$\min_{S, \mathbf{W}, u^1, u^2}\ \max_{\theta}\ \mathcal{L}(S, \mathbf{W}, u^1, u^2, \theta),$$
with constraints $S \in \prod_g \left[\underline{S}_g, \overline{S}_g\right]_{\mathbb{C}}$, $\mathbf{W} \in \mathbb{X}$, $u^1 \in \prod_{bk} [\underline{V}_b^2, \overline{V}_b^2]$ and $u^2 \in \prod_{\ell\mathsf{s}} [0, \overline{I}_\ell^2]$ and with $\theta \in \mathbb{R}^N$. We also point out that (dualOPF) can be written as the corresponding max-min problem. Since (i) the Lagrangian is linear w.r.t. primal variables and also linear w.r.t dual variables (ii) both minimization and maximization sets are convex (iii) the minimization set is compact; we can apply Sion min-max theorem [20] and deduce that the two values are equals, hence $\mathsf{val}(\mathsf{cSDR}) = \mathsf{val}(\mathsf{dualOPF})$. Applying Proposition 1, we conclude that $\mathsf{val}(\mathsf{SDR}) = \mathsf{val}(\mathsf{dualOPF})$. □

This proof makes (dualOPF) appear as the dual of (cSDR). Yet, it may also be seen as the dual problem of (OPF). Indeed, adding the constraint $\forall k \in \mathcal{T}, \mathsf{rank}(W_k) = 1$ in the definition of $\mathbb{X}$, we obtain a max-min problem that also corresponds to (dualOPF). However, we cannot apply the Sion theorem [20] anymore in this case, due to the non-convexity of the rank constraints: we only have $\mathsf{val}(\mathsf{OPF}) \geq \mathsf{val}(\mathsf{dualOPF})$.

C. Lower-bound certification

We consider a realistic computational framework in rational numbers. The parameters of problem (OPF) are thus supposed to have rational real and imaginary parts. We consider the question of computing a certified lower bound, that is, computing a value $v \in \mathbb{Q}$ such that $\mathsf{val}(\mathsf{OPF}) \geq v$. A disadvantage of solving the primal-dual pair (cSDR), (dualcSDR) of SDP problems is that obtaining a certified dual bound requires the exact feasibility of the dual solution. In practice, one may obtain a slightly infeasible solution of (dualcSDR) which invalidates the certification.

Our new unconstrained formulation (dualOPF) helps addressing this issue. We take any vector $\theta \in \mathbb{Q}^N$, produced by any algorithm maximizing $F(\theta)$. The inequality $\mathsf{val}(\mathsf{OPF}) \geq F(\theta)$ holds. Nevertheless, we have to acknowledge that, because of the eigenvalue functions in $f_k(\theta)$ terms, the value of $F(\theta)$ can only be approximated. Despite this, we need to compute an exact lower bound on $f_k(\theta)$, so as to compute an exact lower bound on $F(\theta)$. For any $M \in \mathbb{H}_n$ with rational coefficients, we define $\lambda_{\mathsf{LB}}(M) := \min_{1 \leq i \leq n} M_{ii} - \sum_{j \neq i} |M_{ij}|$, which is a lower bound on $\lambda_{\min}(M)$ due to Gershgorin's circle theorem. We point out that computing the magnitude $|M_{ij}| = \sqrt{\mathsf{Re}(M_{ij})^2 + \mathsf{Im}(M_{ij})^2}$ involves computing a square root, which is not necessarily rational. Yet, we can always round up the output of the square root algorithm to an arbitrary precision, to obtain $s \in \mathbb{Q}$, and check that $s^2 \geq \mathsf{Re}(M_{ij})^2 + \mathsf{Im}(M_{ij})^2$. Such square roots upper bounds will guarantee that the computed value $\underline{\lambda}_{\mathsf{LB}}(M) \in \mathbb{Q}$ is lower than $\lambda_{\mathsf{LB}}(M)$ and thus than $\lambda_{\min}(M)$. By eigendecomposition, we compute a matrix $U \in \mathbb{H}_n$ and a diagonal matrix $D \in \mathbb{R}^{n \times n}$, both with rational coefficients and s.t. $\mathcal{A}_k(\theta) \approx U D U^{\mathsf{H}}$. We define then $\underline{f_k}(\theta | U, D) := \rho_k \min\{\min_i(D_{ii}) + \underline{\lambda}_{\mathsf{LB}}(\mathcal{A}_k(\theta) - U D U^{\mathsf{H}}), 0\}$, which is computable through basic arithmetic operations in rational numbers and
$$f_k(\theta) \geq \underline{f_k}(\theta | U, D). \tag{7}$$
Replacing $f_k(\theta)$ by $\underline{f_k}(\theta | U, D)$ in the expression of $F(\theta)$, we can compute an *exact* lower bound on $\mathsf{val}(\mathsf{OPF})$. Indeed, the functions p_b, q_b, v_b, h_ℓ are computable in rational numbers since they involve elementary arithmetic operations.

Variable	Corresponding component of $\mathcal{A}_k^\dagger(UU^\mathsf{H})$
α_{bk}, for $b \in \mathcal{B}_k$	$\langle E_{bb}, UU^\mathsf{H}\rangle$
β_b, γ_b, for $b \in \mathcal{B}_k$	$\langle \mathcal{H}(M_{bk}), UU^\mathsf{H}\rangle$, $\langle \mathbf{i}\mathcal{Z}(M_{bk}), UU^\mathsf{H}\rangle$
$\eta_{\ell\mathsf{s}}$, for $(\ell, \mathsf{s}) \in \mathcal{L} \times \{\mathsf{f}, \mathsf{t}\}$ s.t. $k(\ell) = k$	$\langle N_{\ell\mathsf{s}}, UU^\mathsf{H}\rangle$
ν_{bak}, μ_{bak} for $(b, a) \in \mathcal{J}_k$	$\langle \mathcal{H}(E_{ba}), UU^\mathsf{H}\rangle$,$\langle \mathbf{i}\mathcal{Z}(E_{ba}), UU^\mathsf{H}\rangle$
ν_{bad}, μ_{bad} for $d \in \mathcal{T}$ s.t. $\mathsf{p}(d) = k$ and for $(b, a) \in \mathcal{J}_d$	$-\langle \mathcal{H}(E_{ba}), UU^\mathsf{H}\rangle$,$-\langle \mathbf{i}\mathcal{Z}(E_{ba}), UU^\mathsf{H}\rangle$
All other coordinates	0

TABLE I
COMPUTATION OF VECTOR $\mathcal{A}_k^\dagger(UU^\mathsf{H})$, FOR GIVEN $k \in \mathcal{T}$ AND $U \in \mathbb{C}^n$

IV. NONSMOOTH OPTIMIZATION ALGORITHM

The objective function of (dualOPF) being nonsmooth, we solve this new formulation for the dual ACOPF with a tailored nonsmooth optimization (NSO) algorithm belonging to the family of proximal bundle methods (PBM) [21].

A. Computing supergradients

Generally speaking, NSO algorithms minimize (resp. maximize) convex (resp. concave) functions based on an *oracle* capable of computing the value of this function and a subgradient (resp. supergradient). In the present case, problem (dualOPF) amounts to maximize a concave function, which is itself a sum of concave functions. Note that v_b, p_b, q_b, h_ℓ are univariate functions with closed-form supergradients, not detailed here for brevity. We focus now on the computation of supergradients of the concave functions f_k.

Proposition 2. *For any $\mathbb{R}$-linear operator $\phi : \mathbb{R}^N \to \mathbb{H}_n$, the superdifferential of the concave function $\lambda_{\min} \circ \phi$ at $\theta \in \mathbb{R}^N$ is* $\mathsf{conv}\{\phi^\dagger(UU^\mathsf{H}) \mid U \in \mathbb{U},\ \phi(\theta)U = \lambda_{\min}(\phi(\theta))U\}$, *where $\phi^\dagger$ is the adjoint operator of ϕ and $\mathbb{U}$ the unit sphere of $\mathbb{C}^n$.*

Proof. As $\lambda_{\min}(\phi(\theta)) = \min_{U\in\mathbb{U}} \langle \phi(\theta), UU^\mathsf{H}\rangle = \min_{U\in\mathbb{U}} \theta^\top \phi^\dagger(UU^\mathsf{H})$ for all $\theta \in \mathbb{R}^N$, $\lambda_{\min} \circ \phi$ is a minimum of linear functions parametrized by a compact set. Thus, we can apply [22, Th.VI.4.4.2], yielding the result. □

Applying this with $\phi = \mathcal{A}_k$, we deduce that $f_k(\theta)$ and a supergradient can be computed simultaneously: first, we compute the minimum eigenvalue λ of $\mathcal{A}_k(\theta)$ and an associated eigenvector U; if $\lambda > 0$, $f_k(\theta) = 0$ and the null vector is a supergradient of f_k at θ; otherwise $f_k(\theta) = \rho_k\lambda$ and $\mathcal{A}_k^\dagger(UU^\mathsf{H})$ is a supergradient at θ. We emphasize that the computational cost of this operation is limited since

- computing (λ, U) amounts to compute a minimum eigenpair of a $n_k \times n_k$ matrix, since $\mathcal{A}_k(\theta) \in \mathbb{H}_n(\mathcal{B}_k)$,
- as shown in Table I, computing $\mathcal{A}_k^\dagger(UU^\mathsf{H})$ amounts to compute Froebenius products involving matrices with very few non-zero coefficients, all related to clique $\mathcal{B}_k$.

B. Structured cutting-plane model

When designing a PBM, a key choice is the cutting-plane (CP) models used to approximate the objective function. In the present case, the objective function of (dualOPF) has a partially separable structure. We exploit this property to produce a rich and structured CP model, sometimes referred as *disaggregated* CP model in the NSO literature. First, we notice that the functions v_b, q_b and h_ℓ can be described exactly as maxima of two affine functions. Second, for any $k \in \mathcal{T}$ and given a set $\mathcal{S}_k$ of pairs $(\Theta, \Lambda) \in \mathbb{R}^N \times \mathbb{R}^N$ s.t. vector Λ is a supergradient of f_k at Θ, we introduce the CP model

$$\check{f}_k(\theta) = \min_{(\Theta,\Lambda)\in\mathcal{S}_k} f_k(\Theta) + \Lambda^\top(\theta - \Theta),$$

for all $\theta \in \mathbb{R}^N$. Similarly, for any $b \in \mathcal{B}$, given a set $\mathcal{S}_b$ of pairs $(u, s) \in \mathbb{R} \times \mathbb{R}$ s.t. s is a supergradient of p_b at u, we introduce the univariate CP model

$$\check{p}_b(x) = \min_{(u,s)\in\mathcal{S}_b} p_b(u) + s(x - u),$$

for all $x \in \mathbb{R}$. Based on the individual CP models $\check{f}_k$ and $\check{p}_b$, we obtain the following structured CP model of F:

$$\check{F}(\theta) = \sum_{k\in\mathcal{T}} \check{f}_k(\theta) + \sum_{k\in\mathcal{T}, b\in\mathcal{B}_k} v_b(\alpha_{bk}) + \sum_{b\in\mathcal{B}} \check{p}_b(\beta_b) + \sum_{b\in\mathcal{B}} q_b(\gamma_b) + \sum_{(\ell,\mathsf{s})\in\mathcal{L}\times\{\mathsf{f},\mathsf{t}\}} h_\ell(\eta_{\ell\mathsf{s}}),$$

for all $\theta = (\alpha, \beta, \gamma, \eta, \nu, \mu) \in \mathbb{R}^N$. During the iterations of the PBM, the model $\check{F}$ evolves as we add or remove CP. Hence, we denote by $\check{F}_t$ the CP model at iteration t.

C. The bundle algorithm

Algorithm 1 presents the PBM framework used to solve (dualOPF). We point out that $\check{F}_t$ is a polyhedral function; hence the subproblem solved at each iteration is a convex Quadratic Programming (QP) problem. The framework presented in Algorithm 1 must be specified in different ways. First, we stop the algorithm after a maximal number of iterations $K_{\max}$ or a maximal number of consecutive null steps $K_{\max}^{\mathsf{null}}$, or whenever $F(\theta^t)$ is greater than a known upper bound on val(OPF) (primal infeasibility). Second, we update the proximal parameter $\kappa \in [\kappa_{\min}, \kappa_{\max}]$, only when: (a) 2 serious steps are consecutive or separated by at most 1 null step, then $\kappa_{t+1} \leftarrow \max\{r_{\mathsf{down}}\kappa_t, \kappa_{\min}\}$ with $r_{\mathsf{down}} \in]0, 1[$; (b) every batch of 10 consecutive null steps, $\kappa_{t+1} \leftarrow \min\{r_{\mathsf{up}}\kappa_t, \kappa_{\max}\}$ with $r_{\mathsf{up}} > 1$. Third, we keep bounded-size CP models by (i) deleting inactive CP every 5 serious steps (ii) aggregating each individual CP model based on the QP dual solution, as explained in [21], every 10 serious steps.

V. NUMERICAL EXPERIMENTS

In this section we illustrate that the proposed formulation and the PBM can be used to obtain accurate and certified dual bounds on val(OPF). We do not use this algorithm as a standalone solver but as a *post-processing* step after calling the IP solver MOSEK [9]; Tables II and III show that this post-processing enables substantial accuracy improvements for several instances from the library PGLib-OPF v21.07 [17].

Algorithm 1: Proximal bundle method.

Data: $\hat{\theta}^0 \in \mathbb{R}^d$; model $\check{F}_0$; $m \in [0,1]$; $tol, \kappa_0 \in \mathbb{R}_{++}$.
$t \leftarrow 0$, $\delta_0 \leftarrow \infty$;
while $\delta_t > tol$ **do**
  Let θ^t be a solution of

$$\max_{\theta \in \mathbb{R}^N} \check{F}_t(\theta) - \frac{1}{2}\kappa_t \|\theta - \hat{\theta}^t\|^2. \qquad (\mathsf{QP}_t)$$

  Call the oracles to compute the value and the supergradients of the functions f_k and p_b $(\star)$;
  Compute $F(\theta^t)$;
  $\delta_t \leftarrow F_t(\hat{\theta}^t) - \check{F}_t(\theta^t) + \frac{1}{2}\kappa_t \|\theta^t - \hat{\theta}^t\|^2$;
  if $F(\theta^t) \leq F(\hat{\theta}^t) - m\delta_t$ **then**
    `// Serious step`
    $\hat{\theta}^{t+1} \leftarrow \theta^t$;
  end
  else
    `// Null step`
    $\hat{\theta}^{t+1} \leftarrow \hat{\theta}^t$;
  end
  Based on the oracle results $(\star)$, update the models $\check{f}_k$ and $\check{p}_b$ by adding the corresponding cutting planes ;
  Update the model $\check{F}_t$ accordingly to build $\check{F}_{t+1}$;
  Update κ_t to build κ_{t+1} ;
  $t \leftarrow t+1$;
end

A. Experimental setting

For all experiments, we used a 64-bit Ubuntu computer with 32 Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz and 32 GB RAM. Algorithm 1 was implemented in Python 3.6. We compute the clique decomposition thanks to the `chompack` package. To manipulate sparse matrices and solve eigenproblems, we use the scientific packages `Scipy` and `numpy`. At each iteration, we solve the QP problem in its dual form, using the open-source solver OSQP [23]. We execute Algorithm 1 with an Armijo parameter $m = 0.01$ and a relative tolerance parameter of 10^{-6}. We also set $K_{\max} = 500$ and $K_{\max}^{\mathsf{null}} = 50$. Based on the isometry between $n \times n$ complex PSD matrices and $2n \times 2n$ real PSD matrices [24], we reformulate problem (cSDR) and solve it with the IP solver MOSEK. We initialize the PBM with a dual solution given by MOSEK; this is why we do not evaluate here the PBM as a standalone solver, but as a post-processing for MOSEK. We point out that (i) MOSEK is considered as the state-of-the-art solver for the ACOPF SDR [2], [6] (ii) we call MOSEK with primal and dual feasibility tolerances (10^{-10}) smaller than tolerance by default (10^{-8}). Therefore, the examples of MOSEK's inaccuracy presented here cannot be attributed to an early stop of MOSEK due to high-tolerance configuration. The considered ACOPF instances are taken from the reference library PGLib-OPF v21.07 [17], with typical operating conditions (TYP) and less than 7,000 buses. The code is available in the following GitHub repository under MIT license: `github.com/aoustry/dualACOPFsolver`. The full results tables and logs are also available at this link.

Instance (case_###)	Est. LB MOSEK	Cert. LB MOSEK	Cert. LB post-proc.	Progress est. LB	Progress cert. LB
3012wp_k	2.56486	2.54175	2.57994	0.59%	1.5%
6495_rte	2.65137	2.61727	2.64956	-0.07%	1.2%
3120sp_k	2.13237	2.09093	2.11465	-0.83%	1.1%
2736sp_k	1.30380	1.29815	1.30739	0.28%	0.71%
2737sop_k	7.75459	7.72214	7.77551	0.27%	0.69%
1354_pegase	1.23135	1.23066	1.23738	0.49%	0.55%
6515_rte	2.66299	2.65224	2.66667	0.14%	0.54%
2746wp_k	1.62822	1.62361	1.63139	0.19%	0.48%
6468_rte	2.05831	2.05267	2.06084	0.12%	0.40%
2746wop_k	1.20676	1.20469	1.20806	0.11%	0.28%
6470_rte	2.21458	2.21198	2.21576	0.053%	0.17%
1951_rte	2.08388	2.08152	2.08489	0.049%	0.16%
1888_rte	1.37253	1.37087	1.37296	0.03%	0.15%
2383wp_k	1.86230	1.86129	1.8629	0.03%	0.09%
4917_goc	1.37226	1.37153	1.37239	0.01%	0.06%
2869_pegase	2.44727	2.44645	2.44793	0.03%	0.06%
3022_goc	5.95436	5.95183	5.95505	0.01%	0.05%
2848_rte	1.28496	1.28476	1.28507	0.01%	0.02%
2312_goc	4.35665	4.35582	4.35683	0.00%	0.02%
4601_goc	8.26171	8.26110	8.26216	0.01%	0.01%
2868_rte	2.00945	2.00933	2.00952	0.00%	0.01%
2000_goc	9.72929	9.72904	9.72956	0.00%	0.01%
2742_goc	2.75607	2.75598	2.75607	0.00%	0.00%
4837_goc	8.71895	8.71883	8.71891	-0.00%	0.00%
2853_sdet	2.03604	2.03604	2.03604	0.00%	0.00%
3970_goc	9.60853	9.60741	9.60741	-0.01%	0.00%
3375wp_k	7.40602	7.39211	7.39211	-0.19%	0.00%

TABLE II
ESTIMATED AND CERTIFIED LOWER BOUNDS COMPUTED BY MOSEK AND BY OUR POST-PROCESSING ALGORITHM (PBM)

B. Numerical results

For cases up to 1,000 buses, MOSEK is very accurate and hence, no significant accuracy improvement is made by the post-processing. We focus now on the instances with more than 1,000 buses. For case4019_goc, case4020_goc and case4661_sdet, we obtained an out-of-memory error during MOSEK execution. The results for other instances are presented in Table II. The *estimated* lower bound ($\mathsf{ELB_M}$) is the value of the (potentially slightly infeasible) dual solution $Z_{\mathsf{M}} = (\underline{y}, \overline{y}, \underline{z}, \overline{z}, \underline{\alpha}, \overline{\alpha}, \beta, \gamma, \eta, \nu, \mu)$ computed by MOSEK. The certified lower bound given by MOSEK ($\mathsf{CLB_M}$) is the value $F(\theta_{\mathsf{M}})$ with $\theta_{\mathsf{M}} := (\overline{\alpha} - \underline{\alpha}, \beta, \gamma, \eta, \nu, \mu)$. We point out that if Z_{M} is feasible and optimal in ($\mathsf{dualcSDR}$), then both values $\mathsf{ELB_M}$ and $\mathsf{CLB_M}$ are equal; case2853_sdet gives an example of such a situation. On the contrary, the spread between $\mathsf{ELB_M}$ and $\mathsf{CLB_M}$, observed for most of the instances in Table II, is due to the slight infeasibility of Z_{M} in ($\mathsf{dualcSDR}$). The certified lower bound given by the post-processing ($\mathsf{CLB_P}$) is the best value obtained by running Algorithm 1 initialized with $\hat{\theta}^0 = \theta_{\mathsf{M}}$. For brevity, these absolute values are expressed in scientific notation without mentioning the scale (e+5 or e+6). The progress of the estimated (resp. certified) lower bound is $\frac{\mathsf{CLB_P} - \mathsf{ELB_M}}{\mathsf{ELB_M}}$ (resp. $\frac{\mathsf{CLB_P} - \mathsf{CLB_M}}{\mathsf{CLB_M}}$), expressed in %. Table II shows that, whether we talk about the progress w.r.t. $\mathsf{ELB_M}$ or $\mathsf{CLB_M}$, the post-processing step yields an improvement in most cases: the best certified lower bound gets higher than both values

Instance (case_###)	UB (IPOPT)	Estim. gap decreased by	Certif. gap decreased by	Time overhead
2737sop_k	7.77719	93%	97%	549%
2746wp_k	1.63171	91%	96%	272%
2746wop_k	1.20826	87%	94%	61%
2736sp_k	1.30800	85%	94%	465%
4601_goc	8.26223	86%	93%	4%
3012wp_k	2.59001	60%	79%	444%
2383wp_k	1.86360	46%	70%	814%
2868_rte	2.00961	43%	66%	86%
1354_pegase	1.24250	54%	57%	2200%
2742_goc	2.75616	3.2%	51%	5%
3120sp_k	2.14655	0.0%	43%	346%
2869_pegase	2.45053	20%	36%	444%
2000_goc	9.73324	6.8%	12%	978%
3022_goc	5.98838	2.0%	8.8%	805%
4917_goc	1.38152	1.4%	8.7%	529%
1888_rte	1.40530	1.3%	6.1%	3300%
4837_goc	8.72020	0.0%	6.1%	24%
2312_goc	4.38452	0.6%	3.5%	240%
2853_sdet	2.04037	0.0%	0.0%	36%
3375wp_k	7.42469	0.0%	0.0%	65%
3970_goc	9.60985	0.0%	0.0%	2%

TABLE III
OPTIMALITY GAP REDUCTION THANKS TO THE POST-PROCESSING STEP

for all 27 cases (in Table II) but 5. We see that the progress w.r.t. MOSEK's output value is more than 0.1% in 8 cases; it is more than 0.5% in one case. If we ask for *certification*, then the progress made by the post-processing step are even better: it is more than 0.1% for 13 cases and more than 0.5% in 7 cases.

Unsurprisingly, these accuracy gains have a computational cost: calling the PBM after calling MOSEK induces a time overhead. Table III presents this time overhead, expressed as a percentage of MOSEK's computational time. We observe that this relative overhead is very variable depending on the instance, ranging from 2% to 3300%. To confirm that this computational overhead is worth it, we measured how the lower bound rise reduces the optimality gap. Thanks to the solver IPOPT [25], we compute an upper-bound (UB) on $\mathsf{val}(\mathsf{OPF})$ (see Table III). Note that for all the instances present in Table II but not in Table III, IPOPT failed to converge, hence we could not compute an optimality gap. Table III displays then the relative reduction of the estimated gap $\frac{(\mathsf{UB}-\mathsf{ELB_M})-(\mathsf{UB}-\mathsf{CLB_P})}{\mathsf{UB}-\mathsf{ELB_M}} = \frac{\mathsf{CLB_P}-\mathsf{ELB_M}}{\mathsf{UB}-\mathsf{ELB_M}}$ and of the certified gap $\frac{\mathsf{CLB_P}-\mathsf{CLB_M}}{\mathsf{UB}-\mathsf{CLB_M}}$. Table III shows that the accuracy gains allowed by the post-processing are significant, since they are of the order of magnitude of the optimality gap: the post-processing reduces the certified optimality gap by more than 50% for 10 instances. For 5 instances of them, the certified optimality gap is reduced by more than 90%.

VI. Conclusion

This paper shows that for many PGLib-OPF instances, the dual bound computed by the state-of-the-art interior point solver MOSEK can be improved with a nonsmooth optimization algorithm used as a post-processing step, to such extent that the ACOPF optimality gap is significantly reduced. This illustrates, somehow, a form of complementarity between these two algorithms: a first-order method may provide dual progress in a situation where the employed interior point solver stalls. This works also shows the difficulty of achieving a given precision of the objective function value and thus, the importance of manipulating certified lower bounds, which the proposed approach allows. In the context of a global optimization approach, the bounds exactness is critical to the correctness of the master (e.g. branch-and-bound) algorithm. Further steps concern the extension of this approach to the complex sum-of-squares hierarchy and the acceleration of this bundle algorithm by parallelizing the oracles and using a GPU implementation of OSQP. Leveraging warm-starting, we will also employ our approach to compute dual bounds for series of ACOPF problems with small parameter changes.

VII. Acknowledgment

We thank Jean Maeght and Patrick Panciatici for their expert advice on the ACOPF problem.

References

[1] J. Lavaei and S. H. Low, "Zero Duality Gap in Optimal Power Flow Problem," in *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 92-107, 2012.
[2] D. Molzahn and I. Hiskens, "A Survey of Relaxations and Approximations of the Power Flow Equations", *Found. Trends in Elec. Ener. Syst.*, vol. 4, no. 1-2, pp 1-221, 2019.
[3] D. Bienstock and A. Verma, "Strong NP-hardness of AC power flows feasibility," *Oper. Res. Lett.*, vol. 47, pp. 494–501, 2019.
[4] C. Coffrin, H. Hijazi and P. van Hentenryck, "Strengthening the SDP relaxation of AC power flows with convex envelopes, bound tightening, and valid inequalities," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3549-3558, 2017.
[5] C. Josz, S. Fliscounakis, J. Maeght, and P. Panciatici."AC power flow data in MATPOWER and QCQP format: iTesla, RTE snapshots, and PEGASE." arXiv preprint `arXiv:1603.01533`, 2016.
[6] A. Eltved, J. Dahl, and M. S. Andersen, "On the robustness and scalability of semidefinite relaxation for optimal power flow problems," *Optim. Eng.*, vol. 21, no. 2, pp. 375–392, 2020.
[7] D. K. Molzahn, J. T. Holzer, B. C. Lesieutre and C. L. DeMarco, "Implementation of a Large-Scale Optimal Power Flow Solver Based on Semidefinite Programming," *IEEE Trans. Power Syst.*, vol. 28, no.4, pp. 3987–3998, 2013.
[8] R. Gron, C. R. Johnson, E. M. Sá, and H. Wolkowicz, "Positive Definite Completions of Partial Hermitian Matrices," *Linear Algebra Appl.*, vol. 58, pp. 109-–124, 1984.
[9] MOSEK ApS, "The MOSEK Fusion API for Python. Version 9.3.5.," 2019, `https:/docs.mosek.com/9.3/pythonfusion.pdf`.
[10] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones." *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 625-653, 1999.
[11] M. S. Andersen, J. Dahl, and L. Vandenberghe, "Implementation of nonsymmetric interior-point methods for linear optimization over sparse matrix cones," *Math. Prog. Comp.*,vol. 2, no. 3, pp. 167–201, 2010.
[12] R. Madani, A. Kalbat and J. Lavaei, "ADMM for sparse semidefinite programming with applications to optimal power flow problem," in *2015 54th IEEE Conference on Decision and Control*, pp. 5932–5939.
[13] B. O'Donoghue, E. Chu, N. Parikh and S. Boyd, "Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding," *J. Optim Theory Appl.*, vol.169, no. 3, pp. 1042–1068, 2016.
[14] Y. Zheng, G. Fantuzzi, A. Papachristodoulou, P. Goulart and A. Wynn, "Chordal decomposition in operator-splitting methods for sparse semidefinite programs," *Math. Prog.*, vol. 180, no. 1, pp. 489–532, 2020.
[15] J. Wang, V. Magron and J.-B. Lasserre,"Certifying Global Optimality of AC-OPF Solutions via the CS-TSSOS Hierarchy," arXiv preprint `arXiv:2109.10005`, 2021.
[16] H. Waki, M. Nakata and M. Muramatsu, "Strange behaviors of interior-point methods for solving semidefinite programming problems in polynomial optimization," *Comput. Optim. Appl.*, vol. 53, pp. 823—844, 2012.

[17] S. Babaeinejadsarookolaee, et al, "The Power Grid Library for Benchmarking AC Optimal Power Flow Algorithms," 2019, IEEE PES Task Force on Benchmarks for Validation of Emerging Power System Algorithms, available at: http://arxiv.org/abs/1908.02788.

[18] D. Bienstock, M. Escobar, C. Gentile, and L. Liberti, "Mathematical programming formulations for the alternating current optimal power flow problem," *4OR*, vol. 18, no. 3, pp. 249-292, 2020.

[19] L. Vandenberghe, and M. S. Andersen, "Chordal graphs and semidefinite optimization," *Found. Trends Optim.*, vol. 1, no. 4, pp. 241-433, 2015.

[20] H. Komiya, "Elementary proof for Sion's minimax theorem," *Kodai Mathematical Journal*, vol. 11, no. 1, pp. 5–7, 1988.

[21] C. Lemaréchal, and J. Zowe, "A Condensed Introduction to Bundle Methods in Nonsmooth Optimization". In: *Algorithms for Continuous Optimization*, vol. 434, E. Spedicato, Ed. Dordrecht, Springer, 1994.

[22] J.B. Hiriart-Urruty, and C. Lemaréchal, " Convex analysis and minimization algorithms I: Fundamentals," vol. 305, Springer, 2013.

[23] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An operator splitting solver for quadratic programs." *Math. Prog. Comp.*, vol. 12, no. 4, pp. 637–672, 2020.

[24] J.-C. Gilbert and C. Josz, "Plea for a semidefinite optimization solver in complex numbers," Research Report, Inria Paris. 2017, pp.35., 2017.

[25] A. Wächter and L. T. Biegler, "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming," *Math. Prog.*, vol. 106, no. 1, pp. 25-57, 2006.