A tight compact quadratically constrained convex relaxation of the Optimal Power Flow problem

Amélie Lambert

A tight compact quadratically constrained convex relaxation of the Optimal Power Flow problem

Amélie Lambert

Cnam-CEDRIC, 292 Rue St Martin FR-75141 Paris Cedex 03
amelie.lambert@cnam.fr

Abstract

In this paper, we consider the Optimal Power Flow (OPF) problem which consists in determining the power production at each bus of an electric network by minimizing the production cost. Our contribution is an exact solution algorithm for the OPF problem. It consists in a spatial branch-and-bound algorithm based on a compact quadratically-constrained convex relaxation. It is computed by solving the semidefinite rank relaxation of OPF once at the root node of the algorithm. An important result is that the optimal value of our compact relaxation is equal to the rank relaxation value. Then, at every sub-nodes of our branch-and-bound, the lower bound is obtained by solving a quadratic convex problem instead of an SDP. Another contribution is that we add only $\mathcal{O}(n+m)$ variables that model the squares of the initial variables, where n is the number of buses in the power system, and m the number of transmission lines to construct our relaxation. Then, since the relations between the initial and auxiliary variables are non-convex, we relax them to get a quadratic convex relaxation. Finally, in our branch-and-bound algorithm, we only have to force a reduced number of equalities to prove global optimality. This quadratic convex relaxation approach is here tailored to the OPF problem, but it can address any application whose formulation is a quadratic optimization problem subject to quadratic equalities and ring constraints. Our first experiments on instances of the OPF problem show that our new algorithm `Compact OPF` (`COPF`) is more efficient than the standard solvers and other quadratic convex relaxation based methods we compare it with.

Key words: Global optimization, Optimal Power Flow, Quadratic Convex Relaxation, Semidefinite programming, Quadratically constrained Quadratic Programming

A tight compact quadratically constrained convex relaxation of the Optimal Power Flow problem

Amélie Lambert

Cnam-CEDRIC, 292 Rue St Martin FR-75141 Paris Cedex 03
amelie.lambert@cnam.fr

1. Presentation of the Optimal Power Flow problem

The class of power system optimization problems known as Optimal Power Flow (OPF) is a family of applications of constrained nonlinear optimization that aims to determine the power production at different buses of an electric network by an minimizing objective function. Each of these optimization problems involves power flow equations that are generally described by Ohm's and Kirchhoff's laws. As reflected in recent surveys (Skolfield & Escobedo [2022]; Frank & Rebennack [2016]), the links between the fields of power system engineering and operations research have strengthened in recent years, and OPF is now a problem widely studied by both communities. Several OPF variants are common in both industry and research. In particular, it can include additional constraints, such as those due to Prohibited Operating Zones (POZ) inducing discontinuity in the formulations (see Pinheiro et al. [2022]). Another variant includes power system contingency constraints that take into account an event disabling one or more generators or power system transmission lines. Most approaches that deal with this case are heuristic, such as Brown & Moreno-Centeno [2023]. Another closely related problem is the Unit Commitment (UC) which determines the activation of production units so that total operating costs are minimized. It is modeled as a large-scale, multi-period, non-linear optimization problem. This problem in its deterministic form can be handled by branch-and-cut approaches (Zheng et al. [2016]), or in an uncertain setting by chance constraints programming (Guo et al. [2021]). Finally, with the introduction of renewable energy sources, other extensions address the network reconfiguration problem (see for instance Cavalheiro et al. [2018]). In practice, almost all OPF problems are solved using a linear approximation of the power flow, which can be significantly inaccurate. The search for reliable OPF solution methods therefore remains extremely relevant to the power systems community.

In this paper, we consider the OPF problem that minimizes a production cost. The electrical transmission network is modeled by a mutli-graph $G = (\mathcal{B}, \mathcal{L})$, where each network point belongs to the set $\mathcal{B}$ of nodes (i.e the set of buses), and their connections (i.e. the set of transmission lines) are modeled by the set of arcs $\mathcal{L}$. Every line is represented by two anti-parallel arcs, and $\mathcal{L}$ is partitioned in two sets $\mathcal{L}_0$, $\mathcal{L}_1$, with $|\mathcal{L}_0| = |\mathcal{L}_1|$. We denote by $n = |\mathcal{B}|$ the number of buses, and $m = |\mathcal{L}|$ the number of lines. We assume that there is an electric demand at each node also called load. We distinguish two classes of nodes: $\mathcal{B} = \mathcal{B}_g \cup \mathcal{B}_d$, where $\mathcal{B}_g$ is the set of nodes that generates and flows the power (the generator nodes), and $\mathcal{B}_d$ is the set of nodes that

only flows the power (the consuming nodes). The aim of the OPF problem is to satisfy demand of all buses while minimizing the total production costs of the generators such that the solution obeys Ohm's law and Kirchhoff's law, and follows the physical limits of the electrical transmission network.

This problem is naturally formulated with complex variables. Let $\mathbf{Y} \in \mathbb{C}^{n\times n}$ be the admittance matrix, which has component $\mathbf{Y}_{ij} = \mathbf{G}_{ij} + \mathrm{j}\mathbf{B}_{ij}$ for each line (i,j) of the network, and $\mathbf{G}_{ii} = \mathbf{g}_{ii} - \sum_{i\neq j}\mathbf{G}_{ij}$, $\mathbf{B}_{ii} = \mathbf{b}_{ii} - \sum_{i\neq j}\mathbf{B}_{ij}$, where $\mathbf{g}_{ii}$ (resp. $\mathbf{b}_{ii}$) is the shunt conductance (resp. susceptance) at bus i, and $\mathrm{j}^2 = -1$. Let p_i, q_i be the real and reactive power output of the generator node i, and $\mathbf{p}_i, \mathbf{q}_i$ the given real and reactive power output of the load node i. For each line $(i,j) \in \mathcal{L}$, we consider the complex power in the rectangular form: $S_{ij} = s^r_{ij} + \mathrm{j}s^c_{ij}$, and for each bus $i \in \mathcal{B}$, the complex voltage in the rectangular form: $V_i = e_i + \mathrm{j}f_i$ where $|V_i|^2 = e_i^2 + f_i^2$ is the voltage magnitude, and we denote by $\delta(i)$ the set of adjacent nodes of bus i. With the above notation, the OPF problem can be expressed by the well known rectangular formulation of Torres & Quintana [1998]:

$$
(OPF)\begin{cases}
\min h(p) = \displaystyle\sum_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}p_i^2 + \mathbf{c}_i p_i\Big) & \\
\text{s.t.} & \\
\quad p_i - \mathbf{p}_i = \mathbf{g}_{ii}(e_i^2+f_i^2) + \displaystyle\sum_{j\in\delta(i)} s^r_{ij} & i\in\mathcal{B}_g \quad (1)\\
\quad -\mathbf{p}_i = \mathbf{g}_{ii}(e_i^2+f_i^2) + \displaystyle\sum_{j\in\delta(i)} s^r_{ij} & i\in\mathcal{B}_d \quad (2)\\
\quad q_i - \mathbf{q}_i = -\mathbf{b}_{ii}(e_i^2+f_i^2) + \displaystyle\sum_{j\in\delta(i)} s^c_{ij} & i\in\mathcal{B}_g \quad (3)\\
\quad -\mathbf{q}_i = -\mathbf{b}_{ii}(e_i^2+f_i^2) + \displaystyle\sum_{j\in\delta(i)} s^c_{ij} & i\in\mathcal{B}_d \quad (4)\\
\quad s^r_{ij} = -\mathbf{G}_{ij}(e_i^2+f_i^2) + \mathbf{G}_{ij}(e_ie_j + f_if_j) - \mathbf{B}_{ij}(e_if_j - e_jf_i) & (i,j)\in\mathcal{L} \quad (5)\\
\quad s^c_{ij} = \mathbf{B}_{ij}(e_i^2+f_i^2) - \mathbf{B}_{ij}(e_ie_j + f_if_j) - \mathbf{G}_{ij}(e_if_j - e_jf_i) & (i,j)\in\mathcal{L} \quad (6)\\
\quad \underline{\mathbf{v}}_i \le e_i^2+f_i^2 \le \overline{\mathbf{v}}_i & i\in\mathcal{B} \quad (7)\\
\quad (s^r_{ij})^2 + (s^c_{ij})^2 \le \overline{\mathbf{S}}_{ij} & (i,j)\in\mathcal{L} \quad (8)\\
\quad \underline{\mathbf{p}}_i \le p_i \le \overline{\mathbf{p}}_i & i\in\mathcal{B}_g \quad (9)\\
\quad \underline{\mathbf{q}}_i \le q_i \le \overline{\mathbf{q}}_i & i\in\mathcal{B}_g \quad (10)\\
\quad (p,q)\in(\mathbb{R}^{|\mathcal{B}_g|},\mathbb{R}^{|\mathcal{B}_g|}),\ (e,f)\in(\mathbb{R}^n,\mathbb{R}^n),\ (s^r,s^c)\in(\mathbb{R}^m,\mathbb{R}^m) & \quad (11)
\end{cases}
$$

where $\mathbf{C} \in \mathbf{S}^+_{|\mathcal{B}_g|}$ is a diagonal and semidefinite matrix, $\mathbf{c} \in \mathbb{R}^{|\mathcal{B}_g|}$ is the vector of linear costs of the power injection at each generator node, $(\underline{\mathbf{v}}, \overline{\mathbf{v}}) \in (\mathbb{R}^n, \mathbb{R}^n)$ are the bounds on the voltage magnitude, $\overline{\mathbf{S}}_{ij}$ the thermal limit of line (i,j), and $(\underline{\mathbf{p}}, \overline{\mathbf{p}}, \underline{\mathbf{q}}, \overline{\mathbf{q}}) \in (\mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|})$. This formulation has $2(n+|\mathcal{B}_g|+m)$ variables, $2n$ quadratic equalities that enforces the active and reactive power balances at each node (Constraints (1)-(4)), $2m$ quadratic equalities that define the power at each line (Constraints (5)-(6)), $2n$ quadratic inequalities

that models the voltage magnitude (Constraints (7)), m quadratic inequalities that ensure the thermal line limits (Constraints (8)), and $4|\mathcal{B}_g|$ box constraints (Constraints (9)-(10)).

The first results of the literature for solving the OPF problem were focused on optimal local solutions, mostly by adapting interior point methods, see, e.g., Wu et al. [1993]; Torres & Quintana [1998]; Jabr et al. [2002]; Wang et al. [2007]; Bostan & Li [2003]. In the context of global optimization, one requires furthermore to determine lower bounds on the OPF problem, and second-order cone programming (SOCP) or semidefinite programming (SDP) relaxations were first used to calculate them (see Bai & Wei [2009]; Bai et al. [2008]; Jabr [2006]; Lavaei & Low [2012]; Zohrizadeh et al. [2020a,b]). The most used SDP relaxation, also named the rank relaxation, leads to very tight lower bounds on the OPF problem. In particular, it was proven in Gan et al. [2015] that this relaxation is exact for a restricted class of problems and under some assumptions. In the case where it is not exact, the rank relaxation can be used within a branch-and-cut algorithm to prove global optimality, as it is done with a complex formulation of the OPF in Chen et al. [2017]. Several other methods use semidefinite programming to reach global optimality. The approach called `SDP-BT` Gopinath et al. [2020] tightened the rank relaxation with RLT inequalities, and then iterates with Optimality Based Bound Tightening (OBBT) techniques until reaching either an optimality threshold or a maximum number of iteration. Another approach is to strength the rank relaxation following the ideas of the hierarchy of Lasserre [2001] that can be applied to any polynomial optimisation problem . This approach was specialized in the context of the OPF problem in Josz et al. [2015], and showed its efficiency to solve small-size problems. It was also used in Molzahn & Hiskens [2015] to strengthen the lower bounds for larger problems. Unfortunately, in practice, using interior point methods for solving large SDP relaxations, with increasing sizes at each rank of the hierarchy, is intractable for large networks. Several specialized algorithms that exploit the sparsity of power networks were thus proposed to solve the rank relaxation (see Jabr [2006, 2012]; Madani et al. [2015]; Molzahn & Hiskens [2015]; Molzahn et al. [2013]).

More recently, several cheaper computable convex relaxations were introduced for the OPF problem. For instance, linear and quadratic envelopes for trigonometric functions in the polar formulation of the OPF problem are constructed in Coffrin et al. [2016]; Coffrin et al. [2017]; Coffrin & Hentenryck [2014], and strong SOCP relaxations were introduced in Kocuk et al. [2016, 2018]. These bounds, that can be computed in polynomial time, may then be used within a spatial branch-and-bound framework to solve the OPF problem to global optimality. Another interesting work presented in Lu et al. [2018] introduces a tight piecewise convex relaxation of the (OPF) problem which is then embedded into an adaptive multivariate partitioning algorithm for globally solving it.

Exact algorithms have also been proposed for a variant of problem (OPF) where the thermal line limits (Constraints (8)) are not considered. In this case, variables s^r_{ij} and s^c_{ij} can be replaced in Constraints (1)-(4) by their expression given by Constraints (5)-(6), which reduce the total number of variables and constraints of the formulation. In Phan [2012], a sub-gradient algorithm is proposed to solve the Lagrangian dual of (OPF), which is used as a bound in a global solution approach. In Foster [2013], the spatial branch-and-bound is

based on a piecewise linear approximation. Another exact solution approach called `RC-OPF` was proposed in Godard et al. [2019], that is a specialization of the Mixed-Integer Quadratic Convex Reformulation (`MIQCR`) approach proposed in Elloumi & Lambert [2019] that applies to general quadratic problems. It consists in a branch-and-bound algorithm based on a quadratic convex relaxation of the OPF problem. This convex relaxation, whose value reaches that of the rank relaxation, is computed in a pre-processing step by solving the rank relaxation. Then, at each node of the branch-and-bound tree, the bound is obtained by solving a convex problem with a quadratic convex objective function and linear constraints. The size of the relaxation is a quadratic function of the number of initial variables, since it relies on the introduction of one additional variable for each possible product of the original variables. Unfortunately, the spatial branch-and-bound relies on the relaxation of a large number of non-convex equalities and, as illustrated in Section 5, in practice it can be very time consuming even for medium sized-instances.

In this work, we propose a two-stage algorithm for globally solving (OPF) following the same steps as the method `RC-OPF` of Godard et al. [2019], or the more general `MIQCR` approaches (see Billionnet et al. [2012, 2016]; Elloumi & Lambert [2019]). The first step consists in computing a quadratic convex relaxation that reaches the value of the rank relaxation, and the second one in performing a spatial branch-and-bound based on the latter relaxation. The difference with the approaches of the literature is that we introduce a quadratic convex relaxation of linear size with respect to the size of the considered problem while previous approaches had a quadratic size. For this purpose, we define a family of quadratic relaxations with quadratic constraints instead of the linear constraints used in `RC-OPF`. The idea is to rewrite the problem (OPF) into a parametric family of equivalent problems, where the objective function and the initial constraints are convex. To do so, we use quadratic terms that vanish on the feasible domain weighted by parameters together with auxiliary variables that model the squares of the initial variables.The family of convex relaxations is then obtained by replacing the relations between auxiliary and initial variables by a convex quadratic envelope introduced in Lu et al. [2018]. Then, we propose to compute the best parameters that maximize the value of the associated bound. We first prove that these parameters can be computed from the optimal dual solutions of the semidefinite rank relaxation, and we then show that the optimal value of the resulting relaxation reaches the value of the rank relaxation. Finally, we perform a spatial branch-and-bound based on our compact relaxation whose optimal value is a lower bound valid not only at the root node, but also in all sub-nodes of the tree. Our main contribution is thus an exact solution algorithm for problem (OPF), called `Compact Quadratic Convex Reformulation (COPF)`, that relies on a quadratic convex relaxation whose size is a linear function of the number of initial variables. An additional contribution is that `COPF` addresses the thermal line limits (Constraints (8)) which was not possible with the `RC-OPF` method since the size of resulting quadratic relaxation makes the method numerically impractical.

From a more general point of view, a key advantage of method `COPF` over traditional convexifications (linearizations or convex quadratic relaxations) is that in our spatial branch-and-bound we need to force a significantly smaller number of non-convex equalities to prove global optimality. This is because we add a

smaller number of auxiliary variables that model the squares of the initial variables. As a consequence our branch-and-bound can be faster than those based on complete linearization. Finally, observe that even if our new method is in this paper specialized for solving the OPF problem, the results proved for the first phase of the algorithm do not rely on this application, and they are therefore valid for a more generic class of problems that are quadratic optimization problems subject to quadratic equalities and ring constraints.

The paper is organized as follows. We start by describing the first phase of algorithm `COPF` in Sections 2 and 3. First, we introduce a new family of compact quadratic convex relaxations to (OPF). Further, in Section 3, we show how to calculate the tightest quadratic convex relaxation within this family. Then, in Section 4, we describe the main features of our spatial branch-and-bound which constitutes the second phase of the algorithm. Finally, in Section 5, after by a brief recall of method `RC-OPF` and of the variant it applies, we present a computational comparison of the two approaches. Then, we evaluate `COPF` on instances of problem (OPF) and we compare it with the method `SDP-BT` of Gopinath et al. [2020]. Section 6 draws a conclusion.

2. Building a compact family of quadratic convex relaxations

In this section, we introduce a new family of quadratic convex relaxations to (OPF). We start by rewriting the constraints as a convex set by introducing auxiliary variables and non-convex constraints. Since our aim is to build a relaxation that leads to a tight lower bound, we then rewrite the objective function as a parameterized function. For this, the basic idea is to add to the original objective function $h(y)$, quadratic terms that vanish on the new convex set and that are weighted by vector parameters. We will further carefully choose the values of these parameters such that i) our relaxation is convex, and ii) the optimal value of the relaxation is as large as possible.

2.1. Convexification of the constraints

To build a compact family of quadratic convex relaxations, we start by observing that the structure of the formulation (OPF) is specific. First, only variables p are involved into the objective function. Moreover, the matrix $\mathbf{C}$ is diagonal and positive semidefinite, hence the objective function is convex and separable. It follows that the non convexities only come from the quadratic constraints (1)-(7). Moreover, in the constraints, variables e and f are only involved into quadratic forms, variables p and q only in linear forms, and variables s^r and s^c appear both in linear and quadratic forms.

Starting from the latter observations, our idea is to build an equivalent problem to (OPF), where the original constraints are convexified thanks to $2(n+m)$ auxiliary variables $z = (z^e, z^f) \in \mathbb{R}^{2n}$, and $w =$

$(w^{s^r}, w^{s^c}) \in \mathbb{R}^{2m}$ that model the squares of the initial variables e, f, s^r, and s^c, respectively:

$$\begin{cases} z_i^e = e_i^2 & i \in \mathcal{B} \quad (12) \\ z_i^f = f_i^2 & i \in \mathcal{B} \quad (13) \\ w_{ij}^{s^r} = (s_{ij}^r)^2 & (i,j) \in \mathcal{L} \quad (14) \\ w_{ij}^{s^c} = (s_{ij}^c)^2 & (i,j) \in \mathcal{L} \quad (15) \end{cases}$$

Using these new variables it is easy to rewrite Constraints (1)-(4), (7) and (8) into a linear form, and we get:

$$\begin{cases} p_i - \mathbf{p}_i = \mathbf{g}_{ii}(z_i^e + z_i^f) + \sum_{j \in \delta(i)} s_{ij}^r & i \in \mathcal{B}_g \quad (16) \\ -\mathbf{p}_i = \mathbf{g}_{ii}(z_i^e + z_i^f) + \sum_{j \in \delta(i)} s_{ij}^r & i \in \mathcal{B}_d \quad (17) \\ q_i - \mathbf{q}_i = -\mathbf{b}_{ii}(z_i^e + z_i^f) + \sum_{j \in \delta(i)} s_{ij}^c & i \in \mathcal{B}_g \quad (18) \\ -\mathbf{q}_i = -\mathbf{b}_{ii}(z_i^e + z_i^f) + \sum_{j \in \delta(i)} s_{ij}^c & i \in \mathcal{B}_d \quad (19) \\ \underline{\mathbf{v}}_i \leq z_i^e + z_i^f \leq \overline{\mathbf{v}}_i & i \in \mathcal{B} \quad (20) \\ w_{ij}^{s^r} + w_{ij}^{s^c} \leq \overline{\mathbf{S}}_{ij} & (i,j) \in \mathcal{L} \quad (21) \end{cases}$$

Note that even if the original constraints (8) are already convex, and since we have introduced auxiliary variables w, it is easy linearize them obtaining Constraints (21). These new constraints are obviously redundant with constraints (8), but we will see in the next section that they are necessary to ensure that the optimal value of our relaxation reaches the value of the rank relaxation.

We now focus on the convexification of the quadratic equality constraints (5)-(6). For this, our first step is to transform each equality into two inequalities. We thus introduce for all $(i,j) \in \mathcal{L}$, Constraints (22)-(25) that obviously describe the same feasible set as Constraints (5)-(6):

$$\begin{cases} s_{ij}^r + \mathbf{G}_{ij}(e_i^2 + f_i^2) - \mathbf{G}_{ij}(e_i e_j + f_i f_j) + \mathbf{B}_{ij}(e_i f_j - e_j f_i) \leq 0 & (i,j) \in \mathcal{L} \quad (22) \\ -s_{ij}^r - \mathbf{G}_{ij}(e_i^2 + f_i^2) + \mathbf{G}_{ij}(e_i e_j + f_i f_j) - \mathbf{B}_{ij}(e_i f_j - e_j f_i) \leq 0 & (i,j) \in \mathcal{L} \quad (23) \\ s_{ij}^c - \mathbf{B}_{ij}(e_i^2 + f_i^2) + \mathbf{B}_{ij}(e_i e_j + f_i f_j) + \mathbf{G}_{ij}(e_i f_j - e_j f_i) \leq 0 & (i,j) \in \mathcal{L} \quad (24) \\ -s_{ij}^c + \mathbf{B}_{ij}(e_i^2 + f_i^2) - \mathbf{B}_{ij}(e_i e_j + f_i f_j) - \mathbf{G}_{ij}(e_i f_j - e_j f_i) \leq 0 & (i,j) \in \mathcal{L} \quad (25) \end{cases}$$

Then, recall that Inequalities (22)-(25) are only quadratic on variables e and f. To make them convex, we apply the smallest eigenvalue method introduced in Hammer & Rubin [1970]. Denote by A_{ij}^r (resp. A_{ij}^c) the sub-matrix of the Hessian of the $(i,j)^{th}$ Constraints (5) (resp. (6)) that corresponds to the quadratic terms involving variables e and f only. Let $\lambda(A_{ij}^r)$ be the smallest eigenvalue of matrix A_{ij}^r, and $\mathrm{d}(\lambda(A_{ij}^r))$ be the diagonal matrix where each diagonal term equals $\lambda(A_{ij}^r)$. The basic idea for rewriting Inequalities (22)-(25) as convex functions is to add to them weighted quadratic terms that vanish when Constraints (12)-(13) are satisfied. By choosing suitable parameters, we can make the Hessian matrix positive semidefinite, while preserving the value of the constraint on the feasible domain.

More precisely, we add to each inequality the quadratic term $-\lambda(A^r_{ij})\sum_{k\in\mathcal{B}}(e_k^2 + f_k^2 - z_k^e - z_k^f)$ (resp. $-\lambda(-A^r_{ij})\sum_{k\in\mathcal{B}}(e_k^2+f_k^2-z_k^e-z_k^f)$), and the Hessian matrix of the constraint becomes $\Big[A^r_{ij} - \mathrm{d}(\lambda(A^r_{ij}))\Big]$ (resp. $\Big[-A^r_{ij} - \mathrm{d}(\lambda(-A^r_{ij}))\Big]$). Since A^r_{ij} is a symmetric matrix with real eigenvalues, adding $-\lambda(A^r_{ij})$ on each of its diagonal term shifts the value of its smallest eigenvalue by the same quantity, and thus turns it non-negative. As a consequence, the new Hessian is a positive semidefinite matrix and the resulting constraint is convex. Moreover, the value of the convexified function remains the same as soon as for all $k \in \mathcal{B}$, $e_k^2+f_k^2-z_k^e-z_k^f = 0$, or equivalently when Equalities (12)-(13) are satisfied. Applying the same process to Constraints (24)-(25), we obtain the set $\mathcal{S}$ of convex constraints:

$$\mathcal{S} = \begin{cases} s^r_{ij} + \mathbf{G}_{ij}(e_i^2+f_i^2) - \mathbf{G}_{ij}(e_ie_j+f_if_j) + \mathbf{B}_{ij}(e_if_j-e_jf_i) - \lambda(A^r_{ij})\sum_{k\in\mathcal{B}}\Big(e_k^2+f_k^2-z_k^e-z_k^f\Big) \le 0 & (i,j)\in\mathcal{L} \\ -s^r_{ij} - \mathbf{G}_{ij}(e_i^2+f_i^2) + \mathbf{G}_{ij}(e_ie_j+f_if_j) - \mathbf{B}_{ij}(e_if_j-e_jf_i) - \lambda(-A^r_{ij})\sum_{k\in\mathcal{B}}\Big(e_k^2+f_k^2-z_k^e-z_k^f\Big) \le 0 & (i,j)\in\mathcal{L} \\ s^c_{ij} - \mathbf{B}_{ij}(e_i^2+f_i^2) + \mathbf{B}_{ij}(e_ie_j+f_if_j) + \mathbf{G}_{ij}(e_if_j-e_jf_i) - \lambda(A^c_{ij})\sum_{k\in\mathcal{B}}\Big(e_k^2+f_k^2-z_k^e-z_k^f\Big) \le 0 & (i,j)\in\mathcal{L} \\ -s^c_{ij} + \mathbf{B}_{ij}(e_i^2+f_i^2) - \mathbf{B}_{ij}(e_ie_j+f_if_j) - \mathbf{G}_{ij}(e_if_j-e_jf_i) - \lambda(-A^c_{ij})\sum_{k\in\mathcal{B}}\Big(e_k^2+f_k^2-z_k^e-z_k^f\Big) \le 0 & (i,j)\in\mathcal{L} \end{cases}$$

By replacing Constraints (1)-(8) by Constraints (12)-(21), together with set $\mathcal{S}$ in (OPF), we then obtain an equivalent problem to (OPF) where the initial constraints are rewritten as convex functions.

2.2. A new parameterized objective function

In the following, we first rewrite the convex objective function $h(y)$ as a parameterized function that has the same value as $h(y)$ on the feasible domain of our convex relaxation. To do this, we simply add weighted quadratic functions that vanish on the feasible domain. Our basic idea, is then to compute parameters that give the larger possible objective function value, in order to get a tight relaxation, and in particular we aim at reaching the value of the rank relaxation of (OPF).

We now focus on the rewriting of the objective function into a parameterized convex function. For simplicity, we use in the rest of the paper the notation $x = (e, f) \in \mathbb{R}^{2n}$, $s = (s^r, s^c) \in \mathbb{R}^{2m}$, and $y = (p, q) \in \mathbb{R}^{2|\mathcal{B}_g|}$, to get the following compact notation of the initial equality constraints (1)-(6):

$$\langle A_k, xx^\top\rangle + a_k^{y\top} y + a_k^{s\top} s = b_k \quad k \in \mathcal{C}$$

where $\mathcal{C} = (\mathcal{B}_g, \mathcal{B}_g, \mathcal{B}_d, \mathcal{B}_d, \mathcal{L}, \mathcal{L})$, with $|\mathcal{C}| = 2(n+m)$, and $\forall\, k \in \mathcal{C}$, $A_k \in \mathbf{S}_{2n}$ is the Hessian matrix of constraint k (i.e. matrices A_k^{rg}, A_k^{rd}, A_k^{cg}, A_k^{cd}, A_k^r, and A_k^c for Constraints (1)-(6) respectively), $(a_k^y, a_k^s) \in (\mathbb{R}^{2|\mathcal{B}_g|}, \mathbb{R}^{2m})$ are the vectors of linear coefficients of constraint k, and $b \in \mathbb{R}^{2(n+m)}$, where coefficient b_k is the the right-hand side of constraint k.

We now introduce three vector parameters $(\phi, \gamma, \delta) \in (\mathbb{R}^{2(n+m)}, \mathbb{R}^n, \mathbb{R}^m)$, and we build the following parameterized objective function:

$$h_{\phi,\gamma,\delta}(x,y,s,z,w) = h(y) + \sum_{k\in\mathcal{C}}\Big(\phi_k(\langle A_k, xx^\top\rangle + a_k^{y\top}y + a_k^{s\top}s - b_k)\Big) + \sum_{i\in\mathcal{B}}\gamma_i\Big(x_i^2 + x_{i+n}^2 - z_i - z_{i+n}\Big)$$
$$+ \sum_{(i,j)\in\mathcal{L}}\delta_{ij}\Big((s_{ij})^2 + (s_{i+m,j+m})^2 - w_{ij} - w_{i+m,j+m}\Big)$$

where $h(y)$ is the initial objective function, and we recall that for all $i \in \mathcal{B}$, $x_i = e_i$, $x_{i+n} = f_i$, $z_i = z_i^e$, $z_{i+n} = z_i^f$, and for all $(i,j) \in \mathcal{L}$, $s_{ij} = s_{ij}^r$, $s_{i+m,j+m} = s_{ij}^c$, $w_{ij} = w_{ij}^{s^r}$, and $w_{i+m,j+m} = w_{ij}^{s^c}$.

Observe that there exist parameters (ϕ, γ, δ) such that $h_{\phi,\gamma,\delta}$ is a convex function. Indeed, as mentioned above, function $h(y)$ is convex and separable. Now, the two additional terms are linear in y, z and w, separable in s, and quadratic in x. By taking $\forall\, k \in \mathcal{C}$, $\bar{\phi}_k = 0$, and $\forall i \in \mathcal{B}$, $\bar{\gamma}_i \geq 0$, and $\forall (i,j) \in \mathcal{L}$, $\bar{\delta}_{ij} \geq 0$, the associated function $h_{\bar{\phi},\bar{\gamma},\bar{\delta}}(x,y,s,z,w)$ is obviously convex.

By denoting $\forall k \in \mathcal{C}$, $\lambda_k = \lambda_{min}(A_k)$ and $\lambda'_k = \lambda_{min}(-A_k)$, we are now able to build $(OPF_{\phi,\gamma,\delta})$, a compact parameterized family of equivalent formulations to (OPF):

$$(OPF_{\phi,\gamma,\delta})\begin{cases} \min h_{\phi,\gamma,\delta}(x,y,s,z,w) & \\ \text{s.t.} & \\ \quad (9)(10)(12)-(21) & \\ \quad \langle A_k, xx^\top\rangle + a_k^{y\top}y + a_k^{s\top}s - \lambda_k\sum_{i=1}^{2n}(x_i^2 - z_i) \leq b_k & k\in\mathcal{C} \quad (26)\\ \quad \langle -A_k, xx^\top\rangle - a_k^{y\top}y - a_k^{s\top}s - \lambda'_k\sum_{i=1}^{2n}(x_i^2 - z_i) \leq -b_k & k\in\mathcal{C} \quad (27)\\ \quad x = (e,f) \in \mathbb{R}^{2n},\ s = (s^r, s^c) \in \mathbb{R}^{2m},\ y = (p,q) \in \mathbb{R}^{2|\mathcal{B}_g|} & \quad (28)\\ \quad z = (z^e, z^f) \in \mathbb{R}^{2n},\ w = (w^{s^r}, w^{s^c}) \in \mathbb{R}^{2m} & \quad (29) \end{cases}$$

where Constraints (26)-(27) are Constraints of set $\mathcal{S}$ written with the compact notation. It is easy to see that problem $(OPF_{\phi,\gamma,\delta})$ is equivalent to problem (OPF), in the sense that any optimal solution from one is an optimal solution from the other. Indeed, $h_{\phi,\gamma,\delta}(x,y,s,z,w) = h(y)$ when Constraints (16)-(19), (26), (27) and (12)-(15) are satisfied.

Since we can choose parameters $(\bar{\phi}, \bar{\gamma}, \bar{\delta})$ such that the objective function $h_{\bar{\phi},\bar{\gamma},\bar{\delta}}$ is a convex function, the only constraints that remain non-convex are Constraints (12)-(15), i.e. $z = x^2$ and $w = s^2$. Thus, for given parameters $(\bar{\phi}, \bar{\gamma}, \bar{\delta})$ that make function $h_{\bar{\phi},\bar{\gamma},\bar{\delta}}$ convex, we can derive a family of compact quadratic convex relaxation of $(OPF_{\phi,\gamma,\delta})$ by relaxing the latter equalities into a convex set. For this, we need upper and lower bounds on each variable x_i and s_{ij}. For variables x_i, some trivial initial bounds can easily be deduced from Constraints (7), i. e. $-\sqrt{\overline{\mathbf{v}}_i} \leq x_i \leq \sqrt{\overline{\mathbf{v}}_i}$, and $-\sqrt{\overline{\mathbf{v}}_i} \leq x_{i+n} \leq \sqrt{\overline{\mathbf{v}}_i}$. We denote by $(\ell, u) \in (\mathbb{R}^{2n}, \mathbb{R}^{2n})$ these bounds (i.e. $\ell = (-\sqrt{\overline{\mathbf{v}}}, -\sqrt{\overline{\mathbf{v}}})$ and $u = (\sqrt{\overline{\mathbf{v}}}, \sqrt{\overline{\mathbf{v}}})$). To deduce bound for variables s_{ij}, we use

Constraints (8), i. e. $-\sqrt{\overline{\mathbf{S}}_{ij}} \leq s_{ij} \leq \sqrt{\overline{\mathbf{S}}_{ij}}$, and we denote by $(\ell', u') \in (\mathbb{R}^{2m}, \mathbb{R}^{2m})$ these bounds (i.e. $\ell' = (-\sqrt{\overline{\mathbf{S}}}, -\sqrt{\overline{\mathbf{S}}})$ and $u' = (\sqrt{\overline{\mathbf{S}}}, \sqrt{\overline{\mathbf{S}}})$). Then, instead of using the complete linearization obtained by the McCormicks envelopes (McCormick [1976]) to relax the non-convex equalities (12)-(15), we use its convex hull representation already used in Lu et al. [2018]. More precisely, we first rewrite them into two equivalent inequalities, and keep in our formulation the convex ones, i.e. $z \geq x^2$ and $w \geq s^2$. We thus use the following a convex quadratic envelope:

$$\mathcal{D} = (x, s, z, w) \begin{cases} z_i \leq (u_i + \ell_i)x_i - u_i\ell_i \\ z_i \geq x_i^2 \\ w_{ij} \leq (u'_{ij} + \ell'_{ij})s_{ij} - u'_{ij}\ell'_{ij} \\ w_{ij} \geq s_i^2 \end{cases}$$

Finally, we get $(\overline{OPF}_{\phi,\gamma,\delta})$, a compact family of quadratic convex relaxations to (OPF):

$$(\overline{OPF}_{\phi,\gamma,\delta}) \begin{cases} \min h_{\phi,\gamma,\delta}(x, y, s, z, w) \\ \text{s.t. } (9)(10)(16) - (21)(26) - (29) \\ \quad (x, s, z, w) \in \mathcal{D} \end{cases} \tag{30}$$

For any parameters ϕ, γ, and δ such that $h_{\phi,\gamma,\delta}$ is a convex function, problem $(\overline{OPF}_{\phi,\gamma,\delta})$ is a convex QCQP and can thus be solved in polynomial time. Our aim is to embed it within a spatial branch-and-bound algorithm. Performing such an algorithm highly depends on the quality of the bound at the root node. Moreover, we know that for many instances of the OPF, the rank relaxation provides a tight lower bound. This is why in the rest of the paper, we focus on the computation of a quadratic convex relaxation whose value equals to the optimal value of the rank relaxation.

3. Computing a strong quadratic convex relaxation

We are now interested in the best parameters $(\phi^*, \gamma^*, \delta^*)$ that maximize the optimal value of $(\overline{OPF}_{\phi,\gamma,\delta})$ while making the parameterized function $h_{\phi,\gamma,\delta}$ convex. We prove that these best parameters can be deduced from the dual optimal solution of the rank relaxation of (OPF). For simplicity, we denote by $\gamma' = (\gamma, \gamma) \in \mathbb{R}^{2n}$, and $\delta' = (\delta, \delta) \in \mathbb{R}^{2m}$. With this notation, we can rewrite function $h_{\phi,\gamma,\delta}$ as follows:

$$h_{\phi,\gamma,\delta'}(x, y, s, z, w) = \sum_{i \in \mathcal{B}_g} \left(\mathbf{C}_{ii} y_i^2 + \mathbf{c}_i y_i \right) + \langle \sum_{k \in \mathcal{C}} \phi_k A_k + \mathrm{d}(\gamma'), xx^\top \rangle + \langle d(\delta'), ss^\top \rangle + \sum_{k \in \mathcal{C}} \phi_k (a_k^{y\top} y + a_k^{s\top} s - b_k) - \gamma'^\top z - \delta'^\top w$$

where $\mathrm{d}(v)$ is the diagonal matrix of whose i^{th}-diagonal coefficient equals v_i. We formally pose the problem we aim to solve as follows:

$$(P) \begin{cases} \max\limits_{\phi,\gamma,\delta} \quad v(\overline{OPF}_{\phi,\gamma,\delta}) \\ \quad \sum\limits_{k \in \mathcal{C}} \phi_k A_k + \mathrm{d}(\gamma') \succeq 0 \\ \quad \mathrm{d}(\delta') \succeq 0 \\ \quad \gamma' = (\gamma, \gamma), \ \delta' = (\delta, \delta) \end{cases}$$

where $v(\overline{OPF}_{\phi,\gamma,\delta})$ is the optimal value of problem $(\overline{OPF}_{\phi,\gamma,\delta})$.

We state in Theorem 1 that the optimal value of (P) equals the optimal value of the following the so-called rank relaxation of (OPF):

$$
(SDP)\begin{cases}
\min h(Y,y) = \sum\limits_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}Y_{ii} + \mathbf{c}_i y_i\Big) & \\
\text{s.t.} & \\
\quad \langle A_k, X\rangle + a_k^{y\top} y + a_k^{s\top} s = b_k & \forall k\in\mathcal{C} \quad (31)\\
\quad X_{ii} + X_{i+n,i+n} \le \overline{\mathbf{v}}_i & i\in\mathcal{B} \quad (32)\\
\quad -X_{ii} - X_{i+n,i+n} \le -\underline{\mathbf{v}}_i & i\in\mathcal{B} \quad (33)\\
\quad W_{k+1,k+1} + W_{k+m+1,k+m+1} \le \overline{\mathbf{S}}_k & k=(i,j)\in\mathcal{L} \quad (34)\\
\quad \underline{\mathbf{p}}_i \le p_i \le \overline{\mathbf{p}}_i & i\in\mathcal{B}_g \quad (35)\\
\quad \underline{\mathbf{q}}_i \le q_i \le \overline{\mathbf{q}}_i & i\in\mathcal{B}_g \quad (36)\\
\quad \begin{bmatrix} 1 & y^\top & s^\top \\ y & Y & \star \\ s & \star & W \end{bmatrix} \succeq 0 & \quad (37)\\
\quad X \succeq 0 & \quad (38)\\
\quad y=(p,q)\in\mathbb{R}^{2|\mathcal{B}_g|},\ s\in\mathbb{R}^{2m},\ (Y,X,W)\in(\mathbf{S}_{2|\mathcal{B}_g|},\mathbf{S}_{2n},\mathbf{S}_{2m}) &
\end{cases}
$$

where Constraints (31) are the compact form of Constraints (1)-(6).

$$
\begin{cases}
p_i - \mathbf{p}_i = \mathbf{g}_{ii}(X_{ii}+X_{i+n,i+n}) + \sum\limits_{k=(i,j):j\in\delta(i)} s_k & i\in\mathcal{B}_g\\
-\mathbf{p}_i = \mathbf{g}_{ii}(X_{ii}+X_{i+n,i+n}) + \sum\limits_{k=(i,j):j\in\delta(i)} s_k & i\in\mathcal{B}_d\\
q_i - \mathbf{q}_i = -\mathbf{b}_{ii}(X_{ii}+X_{i+n,i+n}) + \sum\limits_{k=(i,j):j\in\delta(i)} s_{k+m} & i\in\mathcal{B}_g\\
-\mathbf{q}_i = -\mathbf{b}_{ii}(X_{ii}+X_{i+n,i+n}) + \sum\limits_{k=(i,j):j\in\delta(i)} s_{k+m} & i\in\mathcal{B}_d\\
s_k = -\mathbf{G}_{ij}(X_{ii}+X_{i+n,i+n}) + \mathbf{G}_{ij}(X_{ij}+X_{i+n,j+n}) - \mathbf{B}_{ij}(X_{i,j+n}-X_{j,i+n}) & k=(i,j)\in\mathcal{L}\\
s_{k+m} = \mathbf{B}_{ij}(X_{ii}+X_{i+n,i+n}) - \mathbf{B}_{ij}(X_{ij}+X_{i+n,j+n}) - \mathbf{G}_{ij}(X_{i,j+n}-X_{j,i+n}) & k=(i,j)\in\mathcal{L}
\end{cases}
$$

Theorem 1 *The optimal value of (P) equals the optimal value of (SDP).*

Proof.

$\diamond$ To prove that $v(P)\le v(SDP)$, we show that for any feasible solution $(\bar\phi,\bar\gamma,\bar\delta)$ to (P), we have $v(\overline{OPF}_{\bar\phi,\bar\gamma,\bar\delta})\le v(SDP)$, which in turn implies that $v(P)\le v(SDP)$ since the right hand side is constant. Let $(\bar y,\bar s,\bar Y,\bar X,\bar W)$ be a feasible solution of (SDP), and build the solution $(x=0,y=\bar y,s=\bar s,z=\mathrm{D}(\bar X),w=\mathrm{D}(\bar W))$, where $\mathrm{D}(M)$ is the vector composed of the diagonal terms of matrix M. We prove that: *i)* (x,y,s,z,w) is feasible for $(\overline{OPF}_{\bar\phi,\bar\gamma,\bar\delta})$, and *ii)* its objective value is smaller or equal than $v(SDP)$. Since both $(\overline{OPF}_{\bar\phi,\bar\gamma,\bar\delta})$ and (SDP) are minimization problems, $v(\overline{OPF}_{\bar\phi,\bar\gamma\bar\delta})\le v(SDP)$ follows.

i) We show that $(0,\bar y,\bar s,\mathrm{D}(\bar X),\mathrm{D}(\bar W))$ is feasible for $(\overline{OPF}_{\bar\phi,\bar\gamma,\bar\delta})$. Obviously, Constraints (9)-(10), and (16)-(21) are satisfied. We now prove that Constraints (26), (27), and (30) are satisfied.

(a) *Constraints (26).* We start by observing that, since $x = 0$, Constraints (26) can be rewritten as $b_k - a_k^{y\top}\bar{y} - a^{s\top}\bar{s} - \lambda_k \sum_{i=1}^{2n} \bar{X}_{ii} \geq 0$, and by Constraints (31), we have:

$$b_k - a_k^{y\top}\bar{y} - a^{s\top}\bar{s} - \lambda_k \sum_{i=1}^{2n} \bar{X}_{ii} = \langle A_k, \bar{X}\rangle - \lambda_k \sum_{i=1}^{2n} \bar{X}_{ii} = \langle A_k - \mathrm{diag}(\lambda_k), \bar{X}\rangle \geq 0$$

by Constraint (38) and since $A_k - \mathrm{diag}(\lambda_k) \succeq 0$

(b) *Constraints (27).* Similarly, since $x = 0$, Constraints (27) can be rewritten as $-b_k + a_k^{y\top}\bar{y} + a^{s\top}\bar{s} - \lambda'_k \sum_{i=1}^{2n} \bar{X}_{ii}$, and with Constraints (31) we have:

$$-b_k + a_k^{y\top}\bar{y} + a^{s\top}\bar{s} - \lambda'_k \sum_{i=1}^{2n} \bar{X}_{ii} = \langle -A_k, \bar{X}\rangle - \lambda'_k \sum_{i=1}^{2n} \bar{X}_{ii} = \langle -A_k - \mathrm{diag}(\lambda'_k), \bar{X}\rangle \geq 0$$

by Constraints (38) and since $-A_k - \mathrm{diag}(\lambda'_k) \succeq 0$

(c) *Constraints (30).* Since $x = 0$, the set $\mathcal{D}$ becomes:

$$\begin{cases} \bar{X}_{ii} \leq \overline{\mathbf{v}}_i \\ \bar{X}_{ii} \geq 0 \end{cases}$$

Since $X \succeq 0$, we have $\bar{X}_{ii} \geq 0$. The first inequality comes from $\bar{X}_{i+n,i+n} \geq 0$ and Constraints (32).

ii) Let us now compare the objective values of the two problems to show that $v(\overline{OPF}_{\bar{\phi},\bar{\gamma},\bar{\delta}}) \leq v(SDP)$. Since $x = 0$, the objective function of $(\overline{OPF}_{\bar{\phi},\bar{\gamma},\bar{\delta}})$ can be rewritten as:

$$h_{\bar{\phi},\bar{\gamma},\bar{\delta}}(0,\bar{y},\bar{s},\mathrm{D}(\bar{X}),\mathrm{D}(\bar{W})) = \sum_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}\bar{y}_i^2 + \mathbf{c}_i\bar{y}_i\Big) + \langle d(\bar{\delta}'), \bar{s}\bar{s}^\top\rangle + \sum_{k\in\mathcal{C}}\bar{\phi}_k(a_k^{y\top}\bar{y} + a^{s\top}\bar{s} - b_k) - \bar{\gamma}'^{\top}\mathrm{D}(\bar{X}) - \delta'^{\top}\mathrm{D}(\bar{W})$$

We prove below that $\Delta = h_{\bar{\phi},\bar{\gamma},\bar{\delta}}(0,\bar{y},\bar{s},\mathrm{D}(\bar{X}),\mathrm{D}(\bar{W})) - h(\bar{Y},\bar{y}) \leq 0$

$$\Delta = \sum_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}\bar{y}_i^2 + \mathbf{c}_i\bar{y}_i\Big) + \langle d(\bar{\delta}'), \bar{s}\bar{s}^\top\rangle + \sum_{k\in\mathcal{C}}\bar{\phi}_k(a_k^{y\top}\bar{y} + a^{s\top}\bar{s} - b_k) - \bar{\gamma}'^{\top}\mathrm{D}(\bar{X}) - \delta'^{\top}\mathrm{D}(\bar{W}) - \sum_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}\bar{Y}_{ii} + \mathbf{c}_i\bar{y}_i\Big)$$

$$\Delta = \sum_{i\in\mathcal{B}_g}\mathbf{C}_{ii}(\bar{y}_i^2 - \bar{Y}_{ii}) + \langle d(\bar{\delta}'), \bar{s}\bar{s}^\top\rangle + \sum_{k\in\mathcal{C}}\bar{\phi}_k(a_k^{y\top}\bar{y} + a^{s\top}\bar{s} - b_k) - \bar{\gamma}'^{\top}\mathrm{D}(\bar{X}) - \delta'^{\top}\mathrm{D}(\bar{W})$$

By constraints (31), and by definition of $\bar{\gamma}'$ and $\bar{\delta}'$, we have:

$$\Delta = \sum_{i\in\mathcal{B}_g}\mathbf{C}_{ii}(\bar{y}_i^2 - \bar{Y}_{ii}) + \langle d(\bar{\delta}'), \bar{s}\bar{s}^\top\rangle - \sum_{k\in\mathcal{C}}\bar{\phi}_k\langle A_k, \bar{X}\rangle - \langle \mathrm{d}(\bar{\gamma}'), \bar{X}\rangle - \langle \mathrm{d}(\bar{\delta}'), \bar{W}\rangle$$

$$\Delta = \sum_{i\in\mathcal{B}_g}\mathbf{C}_{ii}(\bar{y}_i^2 - \bar{Y}_{ii}) + \langle d(\bar{\delta}'), \bar{s}\bar{s}^\top - \bar{W}\rangle - \langle \sum_{k\in\mathcal{C}}\bar{\phi}_k A_k + \mathrm{d}(\bar{\gamma}'), \bar{X}\rangle$$

We claim that $\Delta \leq 0$ since each of its 3 terms are non positive:

- $\sum_{i\in\mathcal{B}_g} \mathbf{C}_{ii}(\bar{y}_i^2 - \bar{Y}_{ii}) \leq 0$. By Constraint (37), we know that matrix $\begin{bmatrix} 1 & y^\top \\ y & Y \end{bmatrix} \succeq 0$, which implies that all its 2×2 minors are non negative, and in particular, we have $\forall i,\ \bar{Y}_{ii} - \bar{y}_i^2 \geq 0$. Since moreover $\mathbf{C}_{ii} \geq 0$, we get the result.
- $\langle d(\bar{\delta}'), \bar{s}\bar{s}^\top - \bar{W}\rangle = \sum_{i=1}^{2m} \bar{\delta}'_i(\bar{s}_i^2 - \bar{W}_{ii}) \leq 0$. By Constraint (37), we have $\forall i,\ \bar{W}_{ii} - \bar{s}_i^2 \geq 0$. Moreover, by feasibility of (P), we have $d(\bar{\delta}') \succeq 0$ and we get the result.
- $-\langle \sum_{k\in\mathcal{C}} \bar{\phi}_k A_k + \mathrm{d}(\bar{\gamma}'), \bar{X}\rangle \leq 0$. By Constraint (38), we have $X \succeq 0$. Moreover, by feasibility of (P), we have $\sum_{k\in\mathcal{C}} \bar{\phi}_k A_k + \mathrm{d}(\bar{\gamma}') \succeq 0$ and we get the result.

$\diamond$ Let us secondly prove that $v(P) \geq v(SDP)$. We suppose here that the original problem (OPF) is feasible, and by Proposition 1 of Godard et al. [2019], we know that strong duality holds. Let (D) be the dual of (SDP), under these hypothesis, we have $v(SDP) = v(D)$. To prove that $v(P) \geq v(SDP)$, we equivalently prove that $v(P) \geq v(D)$ where (D) is the dual of (SDP):

$$
(D)\begin{cases}
\max \sum_{i\in\mathcal{B}_g} (\underline{\theta}_i^p \underline{\mathbf{p}}_i + \underline{\theta}_i^q \underline{\mathbf{q}}_i - \overline{\theta}_i^p \overline{\mathbf{p}}_i - \overline{\theta}_i^q \overline{\mathbf{q}}_i) + \sum_{i\in\mathcal{B}} (\underline{\gamma}_i \underline{\mathbf{v}}_i - \overline{\gamma}_i \overline{\mathbf{v}}_i) - \sum_{k\in\mathcal{L}} \delta_k \overline{\mathbf{S}}_k - \sum_{k\in\mathcal{C}} \phi_k b_k - \rho \\
\text{s.t.} \\
M = \begin{bmatrix} \rho & \frac{1}{2}(c + \overline{\theta} - \underline{\theta} + \sum_{k\in\mathcal{C}} \phi_k a_k^y)^\top & (\sum_{k\in\mathcal{C}} \phi_k a_k^s)^\top & \mathbf{0}_{2n}^\top \\ \frac{1}{2}(c + \overline{\theta} - \underline{\theta} + \sum_{k\in\mathcal{C}} \phi_k a_k^y) & C & \mathbf{0}_{2|\mathcal{B}_g|,2m} & \mathbf{0}_{2|\mathcal{B}_g|,2n} \\ \sum_{k\in\mathcal{C}} \phi_k a_k^s & \mathbf{0}_{2m,2|\mathcal{B}_g|} & \mathrm{d}(\delta') & \mathbf{0}_{2m,2n} \\ \mathbf{0}_{2n} & \mathbf{0}_{2n,2|\mathcal{B}_g|} & \mathbf{0}_{2n,2m} & \sum_{k\in\mathcal{C}} \phi_k A_k + \mathrm{d}(\gamma') \end{bmatrix} \succeq 0 \quad (39) \\
\overline{\theta} = (\overline{\theta}^p, \overline{\theta}^q),\ \underline{\theta} = (\underline{\theta}^p, \underline{\theta}^q),\ \gamma' = ((\overline{\gamma} - \underline{\gamma}), (\overline{\gamma} - \underline{\gamma})),\ \delta' = (\delta, \delta) \quad (40) \\
C = \begin{bmatrix} \mathbf{C} & \mathbf{0}_{|\mathcal{B}_g|,|\mathcal{B}_g|} \\ \mathbf{0}_{|\mathcal{B}_g|,|\mathcal{B}_g|} & \mathbf{0}_{|\mathcal{B}_g|,|\mathcal{B}_g|} \end{bmatrix} \in \mathbf{S}_{2|\mathcal{B}_g|} \quad (41) \\
c = (\mathbf{c}, \mathbf{0}_{|\mathcal{B}_g|}),\ a_k = (a_k^y, a_k^s) \quad (42) \\
(\phi, \overline{\gamma}, \underline{\gamma}, \delta, \overline{\theta}^p, \underline{\theta}^p, \overline{\theta}^q, \underline{\theta}^q, \rho) \in (\mathbb{R}^{|\mathcal{C}|}, \mathbb{R}^n, \mathbb{R}^n, \mathbb{R}^m, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R}^{|\mathcal{B}_g|}, \mathbb{R})
\end{cases}
$$

where $\phi, \overline{\gamma}, \underline{\gamma}, \delta, (\overline{\theta}^p, \underline{\theta}^p), (\overline{\theta}^q, \underline{\theta}^q), \rho$ are the dual variables associated to Constraints (31)-(37) respectively. We notice here that the dual variable $\rho \in \mathbb{R}$ is associated to the constraint which sets the value of the first diagonal term of the matrix M to 1. We denote by $\mathbf{0}_n$ ($\mathbf{0}_{n,n}$ respectively) the n-dimensional ($n\times n$-dimensional resp.) vector (matrix resp.) where each coefficient equals 0.

Problems (D) and (P) are both maximization problems. To prove that $v(P) \geq v(D)$, we start with an optimal solution of (D), from which we build a feasible solution of (P) whose objective value is greater than $v(D)$.

Let $(\phi^*, \overline{\gamma}^*, \underline{\gamma}^*, \delta^*, \overline{\theta}^{p*}, \underline{\theta}^{p*}, \overline{\theta}^{q*}, \underline{\theta}^{q*}, \rho^*)$ be an optimal solution of (D), we build the following solution $(\phi = \phi^*, \gamma = (\overline{\gamma}^* - \underline{\gamma}^*), \delta = \delta^*)$ that is obviously feasible for (P), i.e. $\sum_{k\in\mathcal{C}} \phi_k^* A_k + \mathrm{d}(\gamma'^*) \succeq 0$, and $\mathrm{d}(\delta'^*) \succeq 0$, by Constraint (39). The objective value of this solution is equal to $v(\overline{OPF}_{\phi^*, \overline{\gamma}^* - \underline{\gamma}^*, \delta^*})$. To prove that $v(\overline{OPF}_{\phi^*, \overline{\gamma}^* - \underline{\gamma}^*, \delta^*}) \geq v(D)$, we prove that for any feasible solution (x, y, s, z, w) of $(\overline{OPF}_{\phi^*, \overline{\gamma}^* - \underline{\gamma}^*, \delta^*})$, the associated objective value is not smaller than the optimal value of (D). Denote by Δ the difference between

the objective values, we prove below that $\Delta \geq 0$.

$$\Delta = \sum_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{ii}y_i^2 + \mathbf{c}_i y_i\Big) + \langle d(\delta'^*), ss^\top\rangle + \langle\sum_{k\in\mathcal{C}}\phi_k^* A_k + \mathrm{d}(\gamma'^*), xx^\top\rangle + \sum_{k\in\mathcal{C}}\phi_k^*(a_k^{y\top}y + a_k^{s\top}s - b_k) - \gamma'^{*\top}z - \delta'^{*\top}w$$
$$- \sum_{i\in\mathcal{B}_g}(\underline{\theta}_i^{p*}\underline{\mathbf{p}}_i + \underline{\theta}_i^{q*}\underline{\mathbf{q}}_i - \overline{\theta}_i^{p*}\overline{\mathbf{p}}_i - \overline{\theta}_i^{q*}\overline{\mathbf{q}}_i) - \sum_{i\in\mathcal{B}}(\underline{\gamma}_i^*\underline{\mathbf{v}}_i - \overline{\gamma}_i^*\,\overline{\mathbf{v}}_i) + \sum_{k\in\mathcal{L}}\delta_k^*\,\overline{\mathbf{S}}_k + \sum_{k\in\mathcal{C}}\phi_k^* b_k + \rho^*$$

$$\Delta = \langle C, yy^\top\rangle + \langle d(\delta'^*), ss^\top\rangle + \langle\sum_{k\in\mathcal{C}}\phi_k^* A_k + \mathrm{d}(\gamma'^*), xx^\top\rangle - \sum_{i\in\mathcal{B}_g}(\underline{\theta}_i^{p*}\underline{\mathbf{p}}_i + \underline{\theta}_i^{q*}\underline{\mathbf{q}}_i - \overline{\theta}_i^{p*}\overline{\mathbf{p}}_i - \overline{\theta}_i^{q*}\overline{\mathbf{q}}_i) + \sum_{k\in\mathcal{C}}\phi_k^*(a_k^{y\top}y + a_k^{s\top}s)$$
$$+ \sum_{i\in\mathcal{B}_g}\mathbf{c}_i y_i - \sum_{i\in\mathcal{B}}(\underline{\gamma}_i^*\underline{\mathbf{v}}_i - \overline{\gamma}_i^*\,\overline{\mathbf{v}}_i) + \sum_{k\in\mathcal{L}}\delta_k^*\,\overline{\mathbf{S}}_k - \sum_{i\in\mathcal{B}}(\overline{\gamma}_i^* - \underline{\gamma}_i^*)(z_i^e + z_i^f) - \sum_{k\in\mathcal{L}}\delta_k^*(w_k^{s^r} + w_k^{s^c}) + \rho^*$$

By Constraints (9) and (10) and since all coefficients $\overline{\theta}_i^{p*}$, $\underline{\theta}_i^{p*}$, $\overline{\theta}_i^{q*}$, $\underline{\theta}_i^{q*}$ are non-negative, we get:

$$\Delta = \langle C, yy^\top\rangle + \langle d(\delta'^*), ss^\top\rangle + \langle\sum_{k\in\mathcal{C}}\phi_k^* A_k + \mathrm{d}(\gamma'^*), xx^\top\rangle - \sum_{i\in\mathcal{B}_g}(\underline{\theta}_i^{p*}p_i + \underline{\theta}_i^{q*}q_i - \overline{\theta}_i^{p*}p_i - \overline{\theta}_i^{q*}q_i) + \sum_{k\in\mathcal{C}}\phi_k^*(a_k^{y\top}y + a_k^{s\top}s)$$
$$+ \sum_{i\in\mathcal{B}_g}\mathbf{c}_i y_i + \sum_{i\in\mathcal{B}}\big(\overline{\gamma}_i^*(\overline{\mathbf{v}}_i - z_i^e - z_i^f) + \underline{\gamma}_i^*(z_i^e + z_i^f - \underline{\mathbf{v}}_i)\big) + \sum_{k\in\mathcal{L}}\big(\delta_k^*(\overline{\mathbf{S}}_k - w_k^{s^r} - w_k^{s^c}) + \rho^*$$

By Constraints (20) and (21) and and since all coefficients $\overline{\gamma}_i^*$, $\underline{\gamma}_i^*$, δ_k^* are non-negative, we get:

$$\Delta \geq \langle C, yy^\top\rangle + \langle d(\delta'^*), ss^\top\rangle + \langle\sum_{k\in\mathcal{C}}\phi_k^* A_k + \mathrm{d}(\gamma'^*), xx^\top\rangle + (\overline{\theta} - \underline{\theta})^\top y + \sum_{k\in\mathcal{C}}\phi_k^* a_k^{y\top}y + c^\top y + \sum_{k\in\mathcal{C}}\phi_k^* a_k^{s\top}s + \rho^*$$
$$\Delta \geq \langle C, yy^\top\rangle + \langle d(\delta'^*), ss^\top\rangle + \langle\sum_{k\in\mathcal{C}}\phi_k^* A_k + \mathrm{d}(\gamma'^*), xx^\top\rangle + (\overline{\theta} - \underline{\theta} + \sum_{k\in\mathcal{C}}\phi_k^* a_k^y + c)^\top y + \sum_{k\in\mathcal{C}}\phi_k^* a_k^{s\top}s + \rho^*$$
$$\Delta \geq \begin{pmatrix}1\\ y\\ s\\ x\end{pmatrix}^\top M \begin{pmatrix}1\\ y\\ s\\ x\end{pmatrix}$$

Finally, by Constraint (39)

$$\Delta \geq 0$$

□

Theorem 1 states that we can compute parameters $(\phi^*, \gamma^*, \delta^*)$ such that $(\overline{OPF}_{\phi^*,\gamma^*,\delta^*})$ is a convex problem and $v(\overline{OPF}_{\phi^*,\gamma^*,\delta^*}) = v(SDP)$. In Corollary 1, we give a characterization of $(\phi^*, \gamma^*, \delta^*)$.

Corollary 1 *If (OPF) is feasible, an optimal solution $(\phi^*, \gamma^* = (\overline{\gamma}^* - \underline{\gamma}^*), \delta^*)$ of (P) can be computed by solving (SDP). In particular, ϕ^* is the vector of optimal dual variables associated to Constraints (31), $\overline{\gamma}^*, \underline{\gamma}^*$, the vectors of optimal dual variables associated to Constraints (32) and (33) respectively, and δ^* the vector of optimal dual variables associated to Constraints (34).*

Proof. We know by Constraint (39) that $(\phi^*, \gamma^* = (\overline{\gamma}^* - \underline{\gamma}^*), \delta^*)$ is feasible for (P), and in the proof of Theorem 1, we proved that the associated value of (P) reaches $v(SDP)$. □

Our idea is now to perform a spatial branch-and-bound algorithm based on the relaxation $(\overline{OPF}_{\phi^*,\gamma^*,\delta^*})$ that has the same optimal value as (SDP). Usually, and as proposed in Chen et al. [2017], branch-and-bound algorithms that take advantage of the quality of the rank relaxation bound, solve at each node an SDP relaxation. In our method, it is sufficient to solve (SDP) once at the root node, then, in all other nodes, the computation of the bound will be done by solving a QCQP. Thus, a fundamental difference with our approach is that we solve a QCQP at each sub-node instead of an SDP, which can be faster. To further strengthen the value of the bound calculated at the sub-nodes of the tree, we add to our relaxation the convex quadratic inequalities of the initial formulation (Constraints (8) and (43)). Note however that these constraints do not tighten the bound at the root node since they are already considered in (SDP). To sum up, the quadratic convex relaxation we use in our branch-and-bound is thus the following problem:

$$(\overline{OPF}^*)\begin{cases} \min h_{\phi^*,\gamma^*,\delta^*}(x,y,s,z,w) \\ \text{s.t. } (8)(9)(10)(16)-(21)(26)-(30) \\ \quad e_i^2+f_i^2 \leq \overline{\mathbf{v}}_i \qquad\qquad i \in \mathcal{B} \end{cases} \tag{43}$$

A practical advantage of `COPF` is that we only have to enforce $2n$ equalities during the branch-and-bound to prove global optimality, instead of $(2(n+m))^2$ in methods based on complete linearization. To see this, observe that from an optimal solution $(\bar{x},\bar{y},\bar{s},\bar{z},\bar{w})$ of the relaxation $(\overline{OPF}^*)$ satisfying $\bar{x}^2=\bar{z}$, the solution $(\bar{x},\bar{y},\bar{s},\bar{z},\bar{s}^2)$ is the optimal solution of the current branch. Indeed, Constraints (1)–(7), (9)–(10) of (OPF) are satisfied since $\bar{x}^2=\bar{z}$, and since Constraints (8) are included in $(\overline{OPF}^*)$, the solution $(\bar{x},\bar{y},\bar{s},\bar{z},\bar{s}^2)$ is thus feasible for (OPF) with a value $h(\bar{y})=h_{\phi^*,\gamma^*,\delta^*}(\bar{x},\bar{y},\bar{s},\bar{z},\bar{s}^2)$.

Another contribution is that since any convex QCQP can be reformulated as a SOCP, it means that we can derive from $(\overline{OPF}^*)$ a SOCP relaxation that reaches the value of the semidefinite rank relaxation. Of course, to calculate this SOCP relaxation, one must first solve the rank relaxation. However, this SCOP relaxation can be embedded within a spatial branch-and-bound where at each node a SOCP is solved rather than an SDP, which can speeds up the global resolution. Finally, the relaxation $(\overline{OPF}^*)$ is valid for other applications than the OPF problem, since it applies to any quadratic problem with quadratic equality constraints and ring constraints.

4. Solving (OPF) to global optimality with a spatial branch-and-bound

In order to solve (OPF) to global optimality we perform a spatial branch-and-bound (see for instance Belotti et al. [2013] for a complete description) based on the relaxation $(\overline{OPF}^*)$. We sum up our global optimization two-stage algorithm `COPF` in Algorithm 1 and further describe its main features.

In the first phase of `COPF`, we compute $(\overline{OPF}^*)$ as described in Sections 2 and 3. In order to convexify the objective function, we solve the rank relaxation and calculate the parameters $(\phi^*,\gamma^{*'},\delta^*)$ (Step 1). Then, we compute the smallest eigenvalues of the Hessian matrices of the constraints for convexifying them (Steps

Algorithm `COPF`

Phase 1: Building $(\overline{OPF}^*)$

1. $(\phi^*, \gamma^{*'}, \delta^*) \leftarrow$ `Solve`(SDP)
2. $\lambda_k \leftarrow$ `ComputeSmallestEigenValue`(A_k)
3. $\lambda'_k \leftarrow$ `ComputeSmallestEigenValue`$(-A_k)$
4. Build $(\overline{OPF}^*)$

Phase 2: Solving (OPF)

5. $UB \leftarrow$ `ComputeUpperBound()`
6. nodes $\leftarrow 0$ // *Number of nodes*
7. Initialize α // *Parameter between* 0 *and* 1 *for the branching rules*
8. Initialize ϵ // *Relative accuracy of the global solution value*
9. $(x^*, y^*, s^*, \tau^*) \leftarrow$ `BB`$(\overline{OPF}^*, \ell, u)$ // τ^* *denote the optimal solution value of* (OPF)

return (x^*, y^*, s^*, τ^*)

Algorithm 1: The `COPF` algorithm for solving (OPF) to ϵ-global optimality

2 and 3). We finally build $(\overline{OPF}^*)$ (Step 4). The second phase of the algorithm is devoted to the spatial branch-and-bound. We start by determining a feasible solution thanks to an interior point method in order to initialize the global upper bound (Step 5), and we set the parameters of the algorithm: $\alpha \in [0, 1]$ used for computing the value on which we branch, and ϵ the optimality threshold of the branch-and-bound (Steps 6-8). Finally, we run the branch-and-bound (Step 9). In the following we give some details on our implementation of the procedure `BB`$(\overline{OPF}^*, \ell, u)$.

The selection variable strategy

We start by sorting the generator buses in order of decreasing degree, followed by the consuming buses in the same order. We then branch on the variables following this order, always choosing first the real part e of the voltage, then its imaginary part f. Let $(\bar{x}, \bar{y}, \bar{s}, \bar{z}, \bar{w})$ be the solution at the current node, two cases are possible:

1. If $x^2 = z$, then $(\bar{x}, \bar{y}, \bar{s}, \bar{z}, \bar{s}^2)$ is the optimal solution of the considered branch with a value $h(\bar{y})$.
2. Else, following the order of the sorted list of buses, we take the first i^* such that $x_{i^*}^2 \neq z_{i^*}$.

The branching rules

Let $(\bar{x}, \bar{y}, \bar{s}, \bar{z}, \bar{w})$ be the solution of $(\overline{OPF}^*)$ at the current node, i^* the selected variable with a current value $\bar{x}_{i^*}$, and $\alpha \in [0, 1]$ a given parameter. Since the voltage variables are ring constrained, we update the bounds on variable x_{i^*} as follows:

- If $\ell_{i^*} \leq \underline{\mathbf{v}}_i \leq u_{i^*}$, we compute the value $\rho = (1 - \alpha)\frac{u_{i^*} + \ell_{i^*}}{2} + \alpha \underline{\mathbf{v}}_i$,
- Else if $\ell_{i^*} \leq -\underline{\mathbf{v}}_i \leq u_{i^*}$, we compute the value $\rho = (1 - \alpha)\frac{u_{i^*} + \ell_{i^*}}{2} - \alpha \underline{\mathbf{v}}_i$,
- Else, we compute the value $\rho = (1 - \alpha)\frac{u_{i^*} + \ell_{i^*}}{2} + \alpha \bar{x}_{i^*}$,

We then split the feasible domain of x_{i^*} in two partitions:

i) Branch 1: $\ell_{i^*} \leq x_{i^*} \leq \rho$

ii) Branch 2: $\rho \leq x_{i^*} \leq u_{i^*}$.

The node selection strategy

For selecting the next sub-problem we use the "best-first" selection strategy, i.e. we select the node with the highest lower bound.

5. Numerical experiments

To evaluate method `COPF`, we first compare it with the method `RC-OPF` which is also a method based on a convex quadratic relaxation. As `RC-OPF` only supports a variant of problem OPF where the thermal line limits are not considered, we evaluate in a first part `COPF` on this variant. Then, in a second part, we present a comparison of `COPF` with the method `SDP-BT` Gopinath et al. [2020] on instances of problem (OPF).

5.1. Results for instances without thermal line limits

In this section, we make a detailed comparison between methods `COPF` and `RC-OPF` Godard et al. [2019]. We start by outlining `RC-OPF` that applies to the following variant of (OPF):

$$(OPF')\begin{cases} \min h(y) = \sum\limits_{i\in\mathcal{B}_g}\Big(\mathbf{C}_{i,i}y_i^2 + \mathbf{c}_i y_i\Big) & & \\ \text{s.t. } \langle A_k, xx^\top\rangle + a_k^\top y = b_k & k\in\mathcal{C} & (44)\\ \quad \underline{\mathbf{v}}_i \leq e_i^2 + f_i^2 \leq \overline{\mathbf{v}}_i & i\in\mathcal{B} & (45)\\ \quad \underline{\mathbf{p}}_i \leq p_i \leq \overline{\mathbf{p}}_i & i\in\mathcal{B}_g & \\ \quad \underline{\mathbf{q}}_i \leq q_i \leq \overline{\mathbf{q}}_i & i\in\mathcal{B}_g & \\ \quad x=(e,f)\in(\mathbb{R}^n,\mathbb{R}^n),\ y=(p,q)\in(\mathbb{R}^{|\mathcal{B}_g|},\mathbb{R}^{|\mathcal{B}_g|}) & & \end{cases}$$

where Constraints 44 are the compact form of the following set of constraints:

$$\begin{cases} p_i - \mathbf{p}_i = \mathbf{G}_{ii}(e_i^2+f_i^2) + \sum\limits_{j\in\delta(i)}\Big[\mathbf{G}_{ij}(e_ie_j+f_if_j) - \mathbf{B}_{ij}(e_if_j - e_jf_i)\Big] & i\in\mathcal{B}_g\\ -\mathbf{p}_i = \mathbf{G}_{ii}(e_i^2+f_i^2) + \sum\limits_{j\in\delta(i)}\Big[\mathbf{G}_{ij}(e_ie_j+f_if_j) - \mathbf{B}_{ij}(e_if_j - e_jf_i)\Big] & i\in\mathcal{B}_d\\ q_i - \mathbf{q}_i = -\mathbf{B}_{ii}(e_i^2+f_i^2) + \sum\limits_{j\in\delta(i)}\Big[-\mathbf{B}_{ij}(e_ie_j+f_if_j) - \mathbf{G}_{ij}(e_if_j - e_jf_i)\Big] & i\in\mathcal{B}_g\\ -\mathbf{q}_i = -\mathbf{B}_{ii}(e_i^2+f_i^2) + \sum\limits_{j\in\delta(i)}\Big[-\mathbf{B}_{ij}(e_ie_j+f_if_j) - \mathbf{G}_{ij}(e_if_j - e_jf_i)\Big] & i\in\mathcal{B}_d \end{cases}$$

The basic idea is to introduce a matrix variable $X\in\mathbf{S}_{2n}$ that model the products $xx^\top$. With these additional variables, it is easy to linearize Constraints (44) and (45). Then, for a given semi-definite matrix $M\in\mathbf{S}_{2n}^+$, we consider the parameterized convex objective function: $h_M(x,y,X) = h(y) + \langle M, xx^\top - X\rangle$. Obviously function $h_M(x,y,X) = h(y)$ if equalities $X = xx^\top$ are satisfied. To build a family of convex relaxation of (OPF'), we relax the latter non-convex equalities with the McCormick envelopes (McCormick [1976]).

Then, the idea is to compute a positive semi-definite matrix M^* that maximizes the value of the obtained convex relaxation. It is proven in Godard et al. [2019] that M^* can be deduced from the optimal dual variables of the rank relaxation of (OPF'). In particular, by denoting ϕ^* and γ^* the dual optimal variables of Constraints (44) and (45), respectively, $M^* = \sum_{k\in\mathcal{C}} \phi_k^* A_k + \mathrm{d}(\gamma^{'*})$. Finally, we obtain the following quadratic convex relaxation:

$$
(\overline{OPF}^{'*}) \begin{cases} \min f_{M^*}(x,y,X) = \sum_{i\in\mathcal{B}_g} \Big(\mathbf{C}_{ii} p_i^2 + \mathbf{c}_i p_i \Big) + \langle \sum_{r\in\mathcal{C}} \phi_k^* A_k + \mathrm{d}(\gamma^{'*}), xx^\top - X \rangle & \\ \text{s.t. } \langle A_k, X \rangle + a_k^\top y = b_k & k \in \mathcal{C} \\ \quad \underline{\mathbf{v}}_i \leq X_{ii} + X_{i+n,i+n} \leq \overline{\mathbf{v}}_i & i \in \mathcal{B} \\ \quad X_{ij} \leq u_i x_j + \ell_j x_i - u_i \ell_j & (i,j) \in \mathcal{B}^2 \\ \quad X_{ij} \leq \ell_i x_j + u_j x_i - \ell_i u_j & (i,j) \in \mathcal{B}^2 \\ \quad X_{ij} \geq u_i x_j + u_j x_i - u_i u_j & (i,j) \in \mathcal{B}^2 \\ \quad X_{ij} \geq \ell_i x_j + \ell_j x_i - \ell_i \ell_j & (i,j) \in \mathcal{B}^2 \\ \quad x = (e,f) \in \mathbb{R}^{2n},\ y = (p,q) \in \mathbb{R}^{2|\mathcal{B}_g|}, X \in \mathbb{S}_{2n} & \end{cases}
$$

As in method `COPF`, we have the property that the optimal value of $(\overline{OPF}^{'*})$ reaches the optimal value of the rank relaxation of (OPF'). However, the two convex relaxations differ both by their sizes and structures. Indeed, in method `RC-OPF`, we build a linearly constrained quadratic program with $\mathcal{O}(n^2)$ variables and constraints, while in method `COPF`, we compute a quadratically constrained quadratic program with only $\mathcal{O}(n)$ variables and constraints.

Our experiments were carried out on a server with 2 CPU Intel Xeon of 2.3 GHz each of them having 24 threads, and $8 * 16$ GB of RAM with a Linux operating system. We used the semidefinite solver `Mosek` ApS [2019] for solving the rank relaxation of (OPF'). At each node of the spatial branch-and-bound, we used the solver `Mosek` for solving $(\overline{OPF}^{*})$ the QCQP relaxation of method `COPF`, and the solver `Cplex 12.9` for solving $(\overline{OPF}^{'*})$ the QP relaxation of method `RC-OPF`. We used different solvers to solve the convex quadratic relaxation of the two algorithms because the types of relaxations are different (QCQP and QP) and the solvers are the most efficient for solving each type of relaxation (`mosek` for QCQP and `Cplex` for QP). For computing feasible local solutions, we use `Matpower` Zimmerman & Murillo-Sánchez [2020].

In our experiences, we considered medium-sized instances of power networks having 3 to 300 buses. These instances come from the *PG-lib* library (Babaeinejadsarookolaee et al. [2021]), where the thermal line limits are removed from the formulations. We report in Table 1 the characteristics of each instance: its *Name*, and the number of *Buses*, *Generators*, and *Lines* of the considered power network. We indicate in Column *Opt* the best known solution value of each instance. Column $|(y,x)|$ specifies the number of variables of each instance.

Name	*Buses*	*Generators*	*Lines*	*Opt*	$\lvert(y,x)\rvert$
WB2_typ	2	1	1	9.06	6
3_lmbd_typ	3	3	3	5 694.54	12
WB5_typ	5	2	6	13.78	14
5_pjm_typ	5	5	5	14 997.04	20
6ww_typ	6	3	11	3 126.31	18
14_ieee_typ	14	5	20	2 178.08	38
24_ieee_rts_typ	24	33	38	63 352.21	114
30_as_typ	30	6	41	801.55	72
30_ieee_typ	30	6	41	6 592.95	72
39_epri_typ	39	10	46	133 801.71	98
57_ieee_typ	57	7	80	37 589.34	128
73_ieee_rts_typ	73	99	120	189 764.08	344
89_pegase_typ	89	12	210	106 697.06	202
118_ieee_typ	118	54	186	96 881.51	344
162_ieee_dtc_typ	162	12	284	84 785.01	348
179_goc_typ	179	29	263	750 173.90	416
200_activ_typ	200	38	245	27 557.57	476
240_pserc_typ	240	143	448	3 223 503.88	766
300_ieee_typ	300	69	411	546 890.15	738
3_lmbd_api	3	3	3	10 077.59	12
5_pjm_api	5	5	5	73 253.27	20
14_ieee_api	14	5	20	5 688.57	38
24_ieee_rts_api	24	33	38	104 439.96	114
30_as_api	30	6	41	2 770.30	72
30_ieee_api	30	6	41	15 007.92	72
39_epri_api	39	10	46	237 200.66	98
57_ieee_api	57	7	80	49 290.36	128
73_ieee_rts_api	73	99	120	366 984.79	344
118_ieee_api	118	54	186	167 810.79	344
162_ieee_dtc_api	162	12	284	109 391.13	348
179_goc_api	179	29	263	1 670 159.94	416
200_activ_api	200	38	245	26 129.06	476
300_ieee_api	300	69	411	678 486.38	738

Table 1: Characteristics of the considered instances of *PG-lib* library.

To compare the performances of methods `COPF`, `RC-OPF`, and the solver `Baron 19.3.24`, we use a performance profile (see Dolan & Moré [1986]) of the CPU times. A performance profile plots one curve for each solver considered. Each point of a curve gives, for a given factor τ, the proportion of instances whose CPU time was at most τ times greater than the fastest solver. In particular, for $\tau = 1$, we have the proportion of instances on which the solver was the fastest. More precisely, for each instance i and each solver s, we denote by t_{is} the time for solving instance i by solver s, and we define the *performance ratio* as $r_{is} = \frac{t_{is}}{\min\limits_{s} t_{is}}$. Let N be the total number of instances considered, an overall assessment of the performance $P \in [0,1]$ of solver s for a given τ is given by $P(r_{is} \leq \tau) = \frac{1}{N} *$ number of instance such that $r_{is} \leq \tau$. Note that the performances P are computed with respect to the total number of considered instances, even those that were not solved within the time-limit of 1 hour. Since, only 70% of the instances were solved within the time limit the maximum value of P is here 0.7. In Figure 1, we present the performance profile of the CPU times for methods `COPF`, `RC-OPF`, and the solver `Baron 19.3.24` of the instances described in Table 1. We observe that `COPF` and `RC-OPF` significantly outperform the solver `Baron`. In fact `Baron` solves to optimality only 6

instances out of the 33 considered, the largest of which is `pglib_opf_case14_ieee`. The two other approaches are more efficient, since they solve within 1 hours of CPU time, 18 instances for `RC-OPF` and 23 for `COPF`. Moreover, this profile shows that `COPF` is faster than `RC-OPF` for these instances.

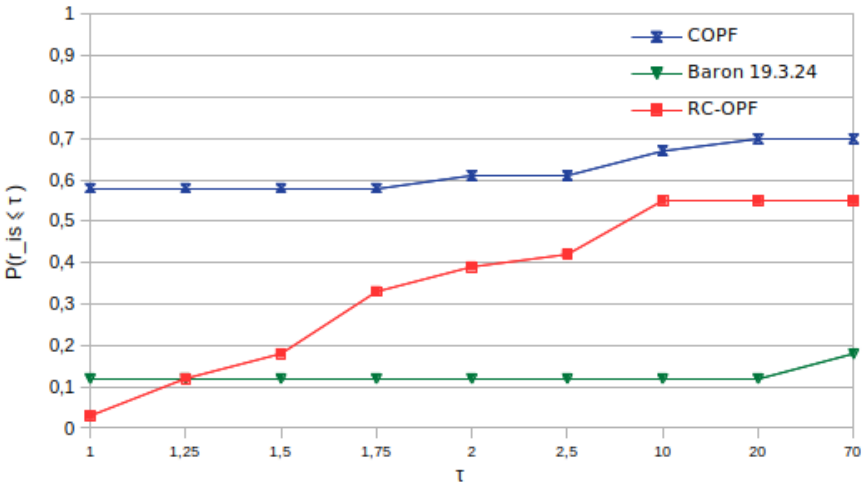

Figure 1: Performance profile of the CPU time for networks with 2 to 300 buses (time limit 1 hour - optimality threshold $\epsilon = 10^-5$).

We present in Table 2, a comparison between methods `COPF` and `RC-OPF`, on instances solved at the root node of both methods. Each line corresponds to one instance and Column *Init gap*: $= \left|\frac{Opt - SDP}{Opt}\right| * 100$, is the initial gap at the root node of the branch-and-bound, where SDP is the optimal value of the rank relaxation of (OPF'), and Opt its best known solution value. Columns $\#vars$ report the number of auxiliary variables in `COPF` or in `RC-OPF`. This is also the number of non-convex equalities to force during the spatial branch-and-bound of each method. Column *Nodes* is the number of nodes visited by the branch-and-bound. A first observation concerns the quality of the initial gap for these 15 instances that confirms the strength of the rank relaxation. For these instances, as expected the difference between methods `RC-OPF` and `COPF` is the size of the relaxations ($\overline{OPF}'^*$), and since the optimality threshold is reached at the root node of the branch-and-bound, the results of both approaches have a similar trend.

Then, we present in Table 3 the results of 17 instances for which the gap is non-zero at the root node. We report in Columns Tm the gap obtained after running the branch-and-bound during T minutes. Note that $T = 60$ minutes corresponds to the time limit. If the instance is solved in less than T minutes, we report between brackets the total CPU time in seconds to solve the instance. If the value is in bold, it means that it is faster, or that it ends with a better gap after 60 minutes of CPU time. These results clearly show that the exact resolution of the instances where the gap is non-zero remains very difficult. In fact, during its branch-and-bound, `RC-OPF` does not decrease the lower bound, despite a large number of explored nodes. This is not the case for `COPF`, which, with a much smaller number of nodes slightly increases the lower bound,

even for the largest instances. This is because the number of auxiliary variables and relaxed equalities $z_i = x_i^2$ is strongly reduced in COPF (by a factor 4 on average). More precisely, out of the 17 considered instances, 5 are solved uniquely by COPF (WB2_typ, WB5_typ, 73ieee_rts_typ, 24_rts_api and 73ieee_rts_api), and 2 have a significantly reduced resolution time (24_iee_rts_typ and 30_as_typ). Overall, this represents a significant improvement on 40% of the instances considered. Let us finally mention that the reformulation time is (always less than 1 minute) significantly shorter than the global resolution time.

Instance		COPF			RC-OPF		
Name	*Init gap*	*#vars*	*CPU time (s)*	*Nodes*	*#vars*	*CPU time (s)*	*Nodes*
3_lmbd_typ	0.000	6	**4**	1	18	5	1
5_pjm_typ	0.000	10	**4**	1	34	6	1
6ww_typ	0.000	12	**3**	1	56	5	1
14_ieee_typ	0.000	28	**3**	1	108	5	1
30_ieee_typ	0.000	60	**4**	1	224	**4**	1
39_epri_typ	0.000	78	**3**	1	262	6	1
89_pegase_typ	0.000	178	**5**	1	1002	6	1
200_activ_typ	0.000	400	**18**	1	1380	23	1
3_lmbd_api	0.000	6	**3**	1	18	5	1
5_pjm_api	0.000	10	**3**	1	34	6	1
14_ieee_api	0.000	28	**3**	1	108	5	1
30_ieee_api	0.000	60	**3**	1	224	5	1
39_epri_api	0.000	78	**3**	1	262	7	1
118_ieee_api	0.000	236	**8**	1	952	9	1
200_activ_api	0.000	400	**22**	1	1380	25	1
300_ieee_api	0.001	600	**52s**	1	2236	69s	1

Table 2: Initial gap, Sizes, CPU times and Nodes for the instances solved at the root node with methods COPF and RC-OPF(optimality threshold $\epsilon = 10^-5$)

Instance		COPF				RC-OPF			
Name	*Init gap*	*#vars*	*15m*	*60m*	*Nodes*	*#vars*	*15m*	*60m*	*Nodes*
WB2_typ	1.947	4	**(7s)**		27	8	1.947	1.947	92939
WB5_typ	28.094	10	11.8	**(1197s)**	9133	34	28.094	28.094	138399
24_ieee_rts_typ	0.012	48	**(7s)**		39	184	(63s)		25
30_as_typ	0.000	60	**(6)**		3	224	(10)		7
57_ieee_typ	0.003	114	0.003	**0.003**	9191	426	0.003	**0.003**	691
73_ieee_rts_typ	0.012	146	**(489s)**		603	578	0.012	0.012	897
118_ieee_typ	0.005	236	0.005	**0.005**	1823	952	0.005	**0.005**	1105
162_ieee_dtc_typ	1.889	324	1.888	**1.888**	1091	1444	1.889	1.889	909
179_goc_typ	0.036	358	0.035	**0.035**	857	1246	0.036	0.036	445
240_pserc_typ	0.270	480	0.269	**0.269**	385	1872	0.270	0.270	1363
300_ieee_typ	0.329	600	0.283	**0.283**	191	2236	0.329	0.329	357
24_ieee_rts_api	0.050	48	**(4s)**		9	184	0.050	0.050	58237
30_as_api	0.026	60	0.025	**0.024**	26513	224	0.026	0.026	37482
57_ieee_api	0.003	114	0.003	**0.003**	8919	426	0.003	0.003	709
73_ieee_rts_api	0.032	146	0.011	**(1688s)**	1997	578	0.032	0.032	635
162_ieee_dtc_api	0.067	324	0.064	**0.064**	1105	1444	0.067	0.067	563
179_goc_api	2.024	358	2.020	**2.020**	785	1246	2.024	2.024	625

Table 3: Initial gap, Sizes, CPU times, Final gaps, and Nodes for the instances with a non-zero initial gap with methods COPF and RC-OPF (time limit 1 hour - optimality threshold $\epsilon = 10^-5$)

5.2. Results for instances including thermal line limits

We now evaluate the performances of `COPF` on instances of (OPF) and compare it with the method `SDP-BT` Gopinath et al. [2020]. The idea of the latter approach is to raise the global lower bound of (OPF), within an iterative procedure, by mixing two approaches: SDP optimization and bound tightening techniques on x variables. More precisely, at each iteration and for each variable x_i, a minimization and a maximization optimization problem that may reduce the domain of x_i are run in parallel. One of the constraints of these problems involves the current lower bound on (OPF). Next, an SDP relaxation whose constraints involve the domain of the x_i is solved, providing an updated lower bound. The process is iterated until either the optimality threshold (with respect to an upper bound calculated at the beginning of the algorithm), or a maximum number of iterations is reached. Method `SDP-BT` can be very efficient since it allows to parallelize the computation of the bounds on variables x_i. On the other hand, it is important to note that it remains a non-exact method, since without a branching procedure, there is no guarantee that, for a given and sufficient small optimality threshold, the method will converge. This is why, in their paper the authors mention as a future work the integration of `SDP-BT` within an adaptive variable partitioning algorithm (see Nagarajan et al. [2019]).

In our experiments, for algorithm `COPF`, we used the semidefinite solver `Mosek` ApS [2019] for solving (SDP) and (OPF^*) and we compute feasible local solutions with the solver `Matpower` Zimmerman & Murillo-Sánchez [2020]. For method `SDP-BT`, we run the code of the authors (available at `https://github.com/coin-or/Gravity/releases/tag/v1.1`) with sub-solvers `Mosek` ApS [2019] and `Ipopt` Wächter & Biegler [2006], with a maximum number of threads of 48, which corresponds to the capacity of the server.

We present in Table 4 the results for instances of (OPF) having 3 to 240 buses where each line corresponds to one instance. Column *Time (s)* is the total CPU time in seconds for solving the instance. If an instance is not solved within the time limit of 1 hour, we report between brackets the final gap $= \left|\frac{Opt - LB}{Opt}\right| * 100$ where LB is the lower bound obtained after 1 hour of CPU time, and Opt its best known solution value. Columns *Iterations* and *Nodes* report the number of iterations in the loop of `SDP-BT`, and of nodes in the b&b of `COPF`, respectively. For this experiences, we use the optimality threshold $\epsilon = 10^{-3}$.

A first observation is that both algorithms are able to reduce the initial gap of all instances considered to less than 1% within the time limit. Next, in every instances with a unique node in `COPF`, the initial gap provided by the solution of (SDP) is smaller than the optimality threshold. These instances are solved very quickly by `SDP-BT` (2 seconds on average) and a little more slowly by `COPF` (23 seconds on average). The difference in the CPU time comes from the `Gravity solver` that is very efficient for solving the rank relaxation. All the other instances have a non-zero initial gap, and even if `COPF` is slightly slower than `SDP-BT` on instance `24_ieee_rts_api` (1.2 times), the limits of the two methods are similar, since they both fail to solve the other 7 instances within the time limit of one hour. Finally, to take advantage of the efficiency of the parallelization while ensuring convergence to the optimal solution, one idea would be to mix the two approaches. To do this, one can run the `OBBT` procedure in parallel, regularly during the course of the branch-

and-bound of algorithm COPF. To illustrate the impact of the combined use of COPF with OBBT, we tested it at the root node for instance 24_iee_ret_api. The results show that the number of nodes is reduced by a factor 5 (we go from 851 at 167 nodes) and the overall resolution time is reduced from 221 seconds to 166 seconds, which is for this instance faster than SDP-BT. The improvement relating to the integration of OBBT will probably not be as effective for larger instances, at least in mono-threading.

	SDP-BT Gopinath et al. [2020]		COPF	
Name	*Time (s)*	*Iterations*	*Time (s)*	*Nodes*
14_ieee_typ	1	1	3	1
24_ieee_rts_typ	1	1	4	1
30_as_typ	1	1	4	1
30_ieee_typ	1	1	4	1
39_epri_typ	1	1	5	1
57_ieee_typ	1	1	6	1
73_ieee_rts_typ	1.5	1	19	1
89_pegase_typ	(0.028%)	1	(0.043%)	619
118_ieee_typ	4.5	1	46	1
162_ieee_dtc_typ	(0.015%)	1	(0.017%)	285
179_goc_typ	4	1	59	1
200_activ_typ	5	1	69	1
240_pserc_typ	(0.012%)	1	(0.013%)	65
14_ieee_api	1	1	3	1
24_ieee_rts_api	176.5	1	221	851
30_ieee_api	1	1	4	1
57_ieee_api	1	1	7	1
73_ieee_rts_api	(0.003%)	1	(0.054%)	1821
118_ieee_api	(0.029%)	1	(0.16%)	513
162_ieee_dtc_api	(0.012%)	1	(0.014%)	319
179_goc_api	(0.005%)	1	(0.006%)	309
200_activ_api	5.5	1	94	1

Table 4: Comparison of the CPU times of methods COPF and SDP-BT (time limit 3600s - optimality threshold $\epsilon = 10^{-3}$ - If the the time limit is reached, we report (g%), where g is the gap after one hour of CPU time.)

6. Conclusion

We consider the OPF problem that determines the power production at each bus of an electric network while minimizing a production cost. We introduce a global optimisation algorithm COPF that is based on a new quadratically constrained quadratic relaxation. This relaxation is compact in the sense that it has only $\mathcal{O}(n+m)$ auxiliary variables and constraints, where n is the number of buses of the network, and m the number of transmission lines. We moreover prove that our quadratic relaxation has the same optimal value as the rank relaxation. Finally, to solve (OPF) to global optimality, we perform a spatial branch-and-bound algorithm based on our new quadratic convex relaxation. We report computational results on instances of the literature. These results show that this new approach is more efficient than the method RC-OPF, and competitive with state-of-the-art methods. A future work consists in using Optimality Based Bound Tightening techniques to further improve the behaviour of COPF on the most difficult instances.

References

ApS, M. (2019). *The MOSEK optimization toolbox for MATLAB manual. Version 9.2.*. URL: http://docs.mosek.com/9.0/toolbox/index.html.

Babaeinejadsarookolaee, S., Birchfield, A., Christie, R. D., Coffrin, C., DeMarco, C., Diao, R., Ferris, M., Fliscounakis, S., Greene, S., Huang, R., Josz, C., Korab, R., Lesieutre, B., Maeght, J., Mak, T. W. K., Molzahn, D. K., Overbye, T. J., Panciatici, P., Park, B., Snodgrass, J., Tbaileh, A., Hentenryck, P. V., & Zimmerman, R. (2021). The power grid library for benchmarking ac optimal power flow algorithms. arXiv:1908.02788.

Bai, X., & Wei, H. (2009). Semi-definite programming-based method for security-constrained unit commitment with operational and optimal power flow constraints. *IET Generation, Transmission & Distribution*, *3*, 182–197(15). URL: https://digital-library.theiet.org/content/journals/10.1049/iet-gtd_20070516.

Bai, X., Wei, H., Fujisawa, K., & Wang, Y. (2008). Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, *30*, 383 – 392. URL: http://www.sciencedirect.com/science/article/pii/S0142061507001378. doi:https://doi.org/10.1016/j.ijepes.2007.12.003.

Belotti, P., Kirches, C., Leyffer, S., Linderoth, J., Luedtke, J., & Mahajan, A. (2013). Mixed-integer nonlinear optimization. *Acta Numerica*, *22*, 1–131.

Billionnet, A., Elloumi, S., & Lambert, A. (2012). Extending the QCR method to the case of general mixed integer program. *Mathematical Programming*, *131*, 381–401.

Billionnet, A., Elloumi, S., & Lambert, A. (2016). Exact quadratic convex reformulations of mixed-integer quadratically constrained problems. *Mathematical Programming*, *158*, 235–266. URL: http://dx.doi.org/10.1007/s10107-015-0921-2. doi:10.1007/s10107-015-0921-2.

Bostan, V., & Li, L. (2003). A decision model for reducing active power losses during electric power dispatching. *Computers & Operations Research*, *30*, 833–849. URL: https://www.sciencedirect.com/science/article/pii/S0305054802000394. doi:https://doi.org/10.1016/S0305-0548(02)00039-4. Operation Research in Emerging Economics.

Brown, W. E., & Moreno-Centeno, E. (2023). A data mining transmission switching heuristic for post-contingency ac power flow violation reduction in real-world, large-scale systems. *Computers & Operations Research*, *160*, 106391. URL: https://www.sciencedirect.com/science/article/pii/S0305054823002551. doi:https://doi.org/10.1016/j.cor.2023.106391.

Cavalheiro, E. M., Vergílio, A. H., & Lyra, C. (2018). Optimal configuration of power distribution networks with variable renewable energy resources. *Computers & Operations Research*, *96*, 272–280. URL: https://www.sciencedirect.com/science/article/pii/S0305054817302526. doi:https://doi.org/10.1016/j.cor.2017.09.021.

Chen, C., Atamtürk, A., & Oren, S. (2017). A spatial branch-and-cut method for nonconvex qcqp with bounded complex variables. *Mathematical Programming*, *165*, 549–577. doi:https://doi.org/10.1007/s10107-016-1095-2.

Coffrin, C., & Hentenryck, P. V. (2014). A linear-programming approximation of ac power flows. *INFORMS J. Comput.*, *26*, 718–734.

Coffrin, C., Hijazi, H., & Hentenryck, P. V. (2017). Convex quadratic relaxations for mixed-integer nonlinear programs in power systems. *Mathematical Programming Computation*, *9*, 321–367.

Coffrin, C., Hijazi, H. L., & Van Hentenryck, P. (2016). The qc relaxation: A theoretical and computational study on optimal power flow. *IEEE Transactions on Power Systems*, *31*, 3008–3018. doi:10.1109/TPWRS.2015.2463111.

Dolan, D., & Moré, J. (1986). Benchmarking optimization software with performance profiles. *Mathematical Programming*, *91*, 201–213.

Elloumi, S., & Lambert, A. (2019). Global solution of non-convex quadratically constrained quadratic programs. *Optimization Methods and Software*, *34*, 98–114. URL: http://dx.doi.org/10.1080/10556788.2017.1350675. doi:10.1080/10556788.2017.1350675. arXiv:http://dx.doi.org/10.1080/10556788.2017.1350675.

Foster, J. D. (2013). Mixed-integer quadratically-constrained programming, piecewise-linear approximation and error analysis with applications in power flow. *Diss. University of Newcastle*, .

Frank, S., & Rebennack, S. (2016). An introduction to optimal power flow: Theory, formulation, and examples. *IIE Transactions (Institute of Industrial Engineers)*, *48*, 1172 – 1197. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84981543675&doi=10.1080%2f0740817X.2016.1189626&partnerID=40&md5=44275c2d3ec451b7441a62275ed384d8. doi:10.1080/0740817X.2016.1189626.

Gan, L., Li, N., Topcu, U., & Low, S. H. (2015). Exact convex relaxation of optimal power flow in radial networks. *IEEE Transactions on Automatic Control*, *60*, 72–87. doi:10.1109/TAC.2014.2332712.

Godard, H., Elloumi, S., Lambert, A., Maegth, J., & Ruiz, M. (2019). Global optimality of optimal power flow using quadratic convex optimization. *6th International Conference on Control, Decision and Information Technologies, CODIT*, (pp. 1–6).

Gopinath, S., Hijazi, H., Weisser, T., Nagarajan, H., Yetkin, M., Sundar, K., & Bent, R. (2020). Proving global optimality of acopf solutions. *Electric Power Systems Research*, *189*, 106688. URL: https://www.sciencedirect.com/science/article/pii/S0378779620304910. doi:https://doi.org/10.1016/j.epsr.2020.106688.

Guo, G., Zephyr, L., Morillo, J., Wang, Z., & Anderson, C. L. (2021). Chance constrained unit commitment approximation under stochastic wind energy. *Computers & Operations Research*, *134*, 105398. URL: https://www.sciencedirect.com/science/article/pii/S0305054821001659. doi:https://doi.org/10.1016/j.cor.2021.105398.

Hammer, P., & Rubin, A. (1970). Some remarks on quadratic programming with 0-1 variables. *Revue Française d'Informatique et de Recherche Opérationnelle*, *4*, 67–79.

Jabr, R. A. (2006). Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems*, *21*, 1458–1459. doi:10.1109/TPWRS.2006.879234.

Jabr, R. A. (2012). Exploiting sparsity in sdp relaxations of the opf problem. *IEEE Transactions on Power Systems*, *27*, 1138–1139. doi:10.1109/TPWRS.2011.2170772.

Jabr, R. A., Coonick, A. H., & Cory, B. J. (2002). A primal-dual interior point method for optimal power flow dispatching. *IEEE Transactions on Power Systems*, *17*, 654–662. doi:10.1109/TPWRS.2002.800870.

Josz, C., Maeght, J., Panciatici, P., & Gilbert, J. C. (2015). Application of the moment-sos approach to global optimization of the opf problem. *IEEE Transactions on Power Systems*, *30*, 463–470. doi:10.1109/TPWRS.2014.2320819.

Kocuk, B., Dey, S. S., & Sun, X. A. (2016). Strong socp relaxations for the optimal power flow problem. *Operations Research*, *64*, 1177–1196. URL: https://doi.org/10.1287/opre.2016.1489. doi:10.1287/opre.2016.1489.

Kocuk, B., Dey, S. S., & Sun, X. A. (2018). Matrix minor reformulation and socp-based spatial branch-and-cut method for the ac optimal power flow problem. *Mathematical Programming Computation*, *10*, 557–596.

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

Lavaei, J., & Low, S. H. (2012). Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, *27*, 92–107. doi:10.1109/TPWRS.2011.2160974.

Lu, M., Nagarajan, H., Bent, R., Eksioglu, S. D., & Mason, S. J. (2018). Tight piecewise convex relaxations for global optimization of optimal power flow. In *2018 Power Systems Computation Conference (PSCC)* (pp. 1–7). doi:10.23919/PSCC.2018.8442456.

Madani, R., Sojoudi, S., & Lavaei, J. (2015). Convex relaxation for optimal power flow problem: Mesh networks. *IEEE Transactions on Power Systems*, *30*, 199–211.

McCormick, G. (1976). Computability of global solutions to factorable non-convex programs: Part i - convex underestimating problems. *Mathematical Programming*, *10*, 147–175.

Molzahn, D. K., & Hiskens, I. A. (2015). Sparsity-Exploiting Moment-Based Relaxations of the Optimal Power Flow Problem. *IEEE Transactions on Power Systems*, *30*, 3168–3180. doi:10.1109/TPWRS.2014.2372478.

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

Nagarajan, H., Lu, M., Wanh, S., Bent, R., & Sundar, K. (2019). An adaptive, multivariate partitioning algorithm for global optimization of nonconvex programs. *Journal of Global Optimization*, *74*, 639–675. URL: https://doi.org/10.1007/s10898-018-00734-1. doi:10.1007/s10898-018-00734-1.

Phan, D. T. (2012). Lagrangian duality and branch-and-bound algorithms for optimal power flow. *Operations Research*, *60 (2)*, 275–285. doi:https://doi.org/10.1287/opre.1110.1036.

Pinheiro, R. B., Balbo, A. R., Cabana, T. G., & Nepomuceno, L. (2022). Solving nonsmooth and discontinuous optimal power flow problems via interior-point ℓ_p-penalty approach. *Computers & Operations Research*, *138*, 105607. URL: https://www.sciencedirect.com/science/article/pii/S0305054821003269. doi:https://doi.org/10.1016/j.cor.2021.105607.

Skolfield, J. K., & Escobedo, A. R. (2022). Operations research in optimal power flow: A guide to recent and emerging methodologies and applications. *European Journal of Operational Research*, *300*, 387–404. URL: https://www.sciencedirect.com/science/article/pii/S0377221721008481. doi:https://doi.org/10.1016/j.ejor.2021.10.003.

Torres, G. L., & Quintana, V. (1998). An interior-point method for nonlinear optimal power flow using voltage rectangular coordinates. *IEEE Transactions on Power Systems*, *13*, 1211–1218.

Wächter, A., & Biegler, L. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, *106*, 25–57. URL: https://doi.org/10.1007/s10107-004-0559-y. doi:10.1007/s10107-004-0559-y.

Wang, H., Murillo-Sánchez, C., Zimmerman, R., & Thomas, R. (2007). On computational issues of market-based optimal power flow. *IEEE Transactions on Power Systems*, *22*, 1185–1193.

Wu, Y., Debs, A., & Marsten, R. (1993). A direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows. *IEEE Transactions on Power Systems*, *9*, 876–883.

Zheng, H., Jian, J., Yang, L., & Quan, R. (2016). A deterministic method for the unit commitment problem in power systems. *Computers & Operations Research*, *66*, 241–247. URL: https://www.sciencedirect.com/science/article/pii/S030505481500057X. doi:https://doi.org/10.1016/j.cor.2015.01.012.

Zimmerman, R. D., & Murillo-Sánchez, C. E. (2020). Matpower. URL: https://doi.org/10.5281/zenodo.4074135. doi:10.5281/zenodo.4074135.

Zohrizadeh, F., Josz, C., Jin, M., Madani, R., Lavaei, J., & Sojoudi, S. (2020a). A survey on conic relaxations of optimal power flow problem. *European Journal of Operational Research*, *287*, 391–409. URL: https://www.sciencedirect.com/science/article/pii/S0377221720300552. doi:https://doi.org/10.1016/j.ejor.2020.01.034.

Zohrizadeh, F., Josz, C., Jin, M., Madani, R., Lavaei, J., & Sojoudi, S. (2020b). A survey on conic relaxations of optimal power flow problem. *European Journal of Operational Research*, *287*, 391–409. URL: https://www.sciencedirect.com/science/article/pii/S0377221720300552. doi:https://doi.org/10.1016/j.ejor.2020.01.034.