UNIVERSITÀ DI PAVIA

Università della Svizzera italiana

Optimal Power Flow problem: a study on Jabr relaxation

The HEXAGON Workshop on power grids

Ambrogio Maria Bernardelli   Stefano Gualandi   Gabor Riccardi

19 June 2024

Overview

Problem definition

The problem

Let us take a network modeled as a graph $(\mathcal{B}, \mathcal{L})$, where $\mathcal{B}$ represents the set of buses and $\mathcal{L}$ represents the set of lines. For every bus k we have a (possibly empty) set of generators $\mathcal{G}(k)$ located at bus k. The problem consists of meeting the energy demand at every bus, and doing so with the lowest possible energy generation cost.

The problem

Let us take a network modeled as a graph $(\mathcal{B}, \mathcal{L})$, where $\mathcal{B}$ represents the set of buses and $\mathcal{L}$ represents the set of lines. For every bus k we have a (possibly empty) set of generators $\mathcal{G}(k)$ located at bus k. The problem consists of meeting the energy demand at every bus, and doing so with the lowest possible energy generation cost.

More precisely, we have the following variables:

- for each bus k we have a complex voltage $V_k = |V_k|e^{j\delta_k}$;
- for each branch km we have two variables S_{km} and S_{mk}, the complex power injected into the branch at k and at m, respectively;
- for each generator g there is power generation $P_g^G + jQ_g^G$.

These variables are subjected to five classes of constraints.

Polar coordinates formulation

$$\inf_{\substack{P_g^G, Q_g^G, \delta_k, \\ |V_k|, S_{km}}} \sum_{g \in \mathcal{G}} F_g(P_g^G, Q_g^G) \tag{1a}$$

s.t.

AC power flow laws:

$$S_{km} = (G_{kk} - jB_{kk})|V_k|^2 + (G_{km} - jB_{km})|V_k||V_m| \cdot (\cos(\theta_{km}) + j\sin(\theta_{km})) \qquad \forall km \in \mathcal{L}, \tag{1b}$$

Flow balance constraints:

$$\sum_{km \in L} S_{km} + P_k^L + jQ_k^L = \sum_{g \in \mathcal{G}(k)} P_g^G + j \sum_{g \in \mathcal{G}(k)} Q_g^G \qquad \forall k \in \mathcal{B}, \tag{1c}$$

Branch limits, generator limits, voltage bounds:

$$|S_{km}|^2 \leq U_{km} \qquad \forall km \in \mathcal{L}, \tag{1d}$$

$$P_g^{\min} \leq P_g^G \leq P_g^{\max},\ Q_g^{\min} \leq Q_g^G \leq Q_g^{\max} \qquad \forall g \in \mathcal{G}, \tag{1e}$$

$$V_k^{\min} \leq |V_k| \leq V_k^{\max} \qquad \forall k \in \mathcal{B}, \tag{1f}$$

$$\theta_{km}^{\min} \leq \theta_{km} \leq \theta_{km}^{\max} \qquad \forall km \in \mathcal{L}. \tag{1g}$$

Variable substitution

One can introduce auxiliary variables to tackle the problem of having sine and cosine functions:

$$
\begin{aligned}
c_{km} &= |V_k||V_m| \cdot \cos(\theta_{km}) && \forall km \in \mathcal{L}, \\
s_{km} &= |V_k||V_m| \cdot \sin(\theta_{km}) && \forall km \in \mathcal{L}, \\
c_{kk} &= |V_k|^2 && \forall k \in \mathcal{B}.
\end{aligned}
$$

Variable substitution

One can introduce auxiliary variables to tackle the problem of having sine and cosine functions:

$$
\begin{aligned}
c_{km} &= |V_k||V_m| \cdot \cos(\theta_{km}) && \forall km \in \mathcal{L}, \\
s_{km} &= |V_k||V_m| \cdot \sin(\theta_{km}) && \forall km \in \mathcal{L}, \\
c_{kk} &= |V_k|^2 && \forall k \in \mathcal{B}.
\end{aligned}
$$

Substituting such variables in the model without adding their definitions gives us a first relaxed model.

Note that by doing so we manage to remove sine and cosine functions but we also lose crucial relations between the new variables.

A first relaxed model

$$\inf_{\substack{P_g^G, Q_g^G, c_{km}, \\ s_{km}, S_{km}, P_{km}, Q_{km}}} \quad F(x) := \sum_{g \in \mathcal{G}} F_g(P_g^G) \tag{2a}$$

Subject to:

$$P_{km} = G_{kk} c_{kk} + G_{km} c_{km} + B_{km} s_{km} \qquad \forall km \in \mathcal{L}, \tag{2b}$$

$$Q_{km} = -B_{kk} c_{kk} - B_{km} c_{km} + G_{km} s_{km} \qquad \forall km \in \mathcal{L}, \tag{2c}$$

$$S_{km} = P_{km} + jQ_{km} \qquad \forall km \in \mathcal{L}, \tag{2d}$$

$$\sum_{km \in L} S_{km} + P_k^L + jQ_k^L = \sum_{g \in \mathcal{G}(k)} P_g^G + j \sum_{g \in \mathcal{G}(k)} Q_g^G \qquad \forall k \in \mathcal{B}, \tag{2e}$$

$$P_{km}^2 + Q_{km}^2 \leq U_{km} \qquad \forall km \in \mathcal{L}, \tag{2f}$$

$$V_k^{\min^2} \leq c_{kk} \leq V_k^{\max^2} \qquad \forall k \in \mathcal{B}, \tag{2g}$$

$$P_g^{\min} \leq P_g^G \leq P_g^{\max}, \; Q_g^{\min} \leq Q_g^G \leq Q_g^{\max} \qquad \forall g \in \mathcal{G}, \tag{2h}$$

$$c_{kk} \geq 0 \qquad \forall k \in \mathcal{B}, \tag{2i}$$

$$V_k^{\max} V_m^{\max} \geq c_{km} \geq 0 \qquad \forall km \in \mathcal{L}, \tag{2j}$$

$$- V_k^{\max} V_m^{\max} \leq s_{km} \leq V_k^{\max} V_m^{\max} \qquad \forall km \in \mathcal{L}, \tag{2k}$$

$$c_{km} = c_{mk}, \; s_{km} = -s_{mk} \qquad \forall km \in \mathcal{L}. \tag{2l}$$

Jabr (I)

Equality

To link the c and s variables we make use of the following equality:

$$c_{km}^2 + s_{mk}^2 = c_{kk} c_{mm} \quad \forall km \in \mathcal{L}. \tag{3}$$

We will denote by Jabr equality ACOPF relaxation the model (2) together with constraints (3).

Jabr (I)

Equality

To link the c and s variables we make use of the following equality:

$$c_{km}^2 + s_{mk}^2 = c_{kk}c_{mm} \quad \forall km \in \mathcal{L}. \tag{3}$$

We will denote by Jabr equality ACOPF relaxation the model (2) together with constraints (3).

These nonconvex couplings constraints can be relaxed as follows.

Inequality

$$c_{km}^2 + s_{mk}^2 \leq c_{kk}c_{mm} \quad \forall km \in \mathcal{L}. \tag{4}$$

Jabr (II)

Note that inequality

$$c_{km}^2 + s_{mk}^2 \leq c_{kk} c_{mm}$$

can be rewritten as

$$c_{km}^2 + s_{mk}^2 + \left(\frac{c_{mm} - c_{kk}}{2} \right)^2 \leq \left(\frac{c_{mm} + c_{kk}}{2} \right)^2,$$

which represents a rotated SOCP cone in $\mathbb{R}^4$. Note also that the cone (4) is the convex hull of (3).

Jabr (II)

Note that inequality

$$c_{km}^2 + s_{mk}^2 \leq c_{kk} c_{mm}$$

can be rewritten as

$$c_{km}^2 + s_{mk}^2 + \left(\frac{c_{mm} - c_{kk}}{2} \right)^2 \leq \left(\frac{c_{mm} + c_{kk}}{2} \right)^2,$$

which represents a rotated SOCP cone in $\mathbb{R}^4$. Note also that the cone (4) is the convex hull of (3).

We will denote by Jabr inequality ACOPF relaxation the model (2) together with constraints (4).

Trees and cycles

Trees

Can we take advantage of the network structure to ensure exactness of a relaxed model?

Trees

Can we take advantage of the network structure to ensure exactness of a relaxed model?

More specifically, does a multisource radial network require other constraints other than the Jabr equality?

Trees

Can we take advantage of the network structure to ensure exactness of a relaxed model?

More specifically, does a multisource radial network require other constraints other than the Jabr equality?

Lemma 1.

If $(\mathcal{B}, \mathcal{L})$ is a multisource radial network, then the Jabr equality ACOPF relaxation is exact [Jab06].

Trees

Can we take advantage of the network structure to ensure exactness of a relaxed model?

More specifically, does a multisource radial network require other constraints other than the Jabr equality?

Lemma 1.

If $(\mathcal{B}, \mathcal{L})$ is a multisource radial network, then the Jabr equality ACOPF relaxation is exact [Jab06].

Why do we need a tree structure for the exactness of the model?

Loop constraints

Definition 1 (Loop constraint).

Given a cycle $\mathcal{C}$ on nodes $\{k_1, \dots, k_n\}$, we define the loop constraint on $\mathcal{C}$ as the following

$$\sum_{j=0}^{\lfloor n/2 \rfloor} \sum_{\substack{A \subset [n] \\ |A| = 2j}} (-1)^j \prod_{h \in A} s_{k_h k_{h+1}} \prod_{l \in A^c} c_{k_l k_{l+1}} = \prod_{i=1}^{n} c_{k_i k_i}, \tag{5}$$

with $A^c := [n] \setminus A$.

Loop constraints

Definition 1 (Loop constraint).

Given a cycle $\mathcal{C}$ on nodes $\{k_1, \ldots, k_n\}$, we define the loop constraint on $\mathcal{C}$ as the following

$$\sum_{j=0}^{\lfloor n/2 \rfloor} \sum_{\substack{A \subset [n] \\ |A|=2j}} (-1)^j \prod_{h \in A} s_{k_h k_{h+1}} \prod_{l \in A^c} c_{k_l k_{l+1}} = \prod_{i=1}^{n} c_{k_i k_i}, \tag{5}$$

with $A^c := [n] \setminus A$.

Lemma 2.

The Jabr equality ACOPF relaxation together with the additional loop constraint (5) written for every cycle of $(\mathcal{B}, \mathcal{L})$ is exact.

Constraint redundancy

Definition 2 (Cycle space).

The (binary) cycle space of an undirected graph is the set of its even-degree subgraphs.

Definition 3 (Cycle basis).

A cycle basis of an undirected graph is a set of simple cycles that forms a basis of the cycle space of the graph.

Constraint redundancy

Definition 2 (Cycle space).

The (binary) cycle space of an undirected graph is the set of its even-degree subgraphs.

Definition 3 (Cycle basis).

A cycle basis of an undirected graph is a set of simple cycles that forms a basis of the cycle space of the graph.

Lemma 3.

It is sufficient to write (5) for every cycle in a cycle basis of $(\mathcal{B}, \mathcal{L})$.

Linearizations

3-cycles and 4-cycles

We first focus on short cycles, namely, cycles made up of 3 or 4 nodes [KDS16]. Note that, in this case, the polynomials constituiting the loop constraints are cubic polynomials and quartic polynomials, respectively.

3-cycles and 4-cycles

We first focus on short cycles, namely, cycles made up of 3 or 4 nodes [KDS16]. Note that, in this case, the polynomials constituiting the loop constraints are cubic polynomials and quartic polynomials, respectively.

In this particular case, it is possible to reduce the degree of the polynomials thanks to the following result.

Lemma 4.

Given a cycle of length 3 or 4, if the Jabr equality is satisifed on all the branches of the cycle, the loop constraint (5) can be replaced exactly by two bilinear constraints.

3-cycles and 4-cycles

We first focus on short cycles, namely, cycles made up of 3 or 4 nodes [KDS16]. Note that, in this case, the polynomials constituiting the loop constraints are cubic polynomials and quartic polynomials, respectively.

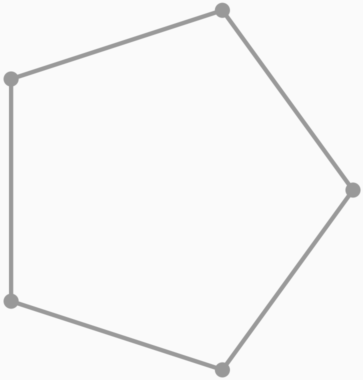
In this particular case, it is possible to reduce the degree of the polynomials thanks to the following result.

Lemma 4.

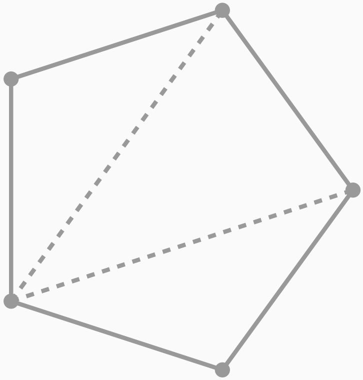
Given a cycle of length 3 or 4, if the Jabr equality is satisifed on all the branches of the cycle, the loop constraint (5) can be replaced exactly by two bilinear constraints.

What about larger cycles?

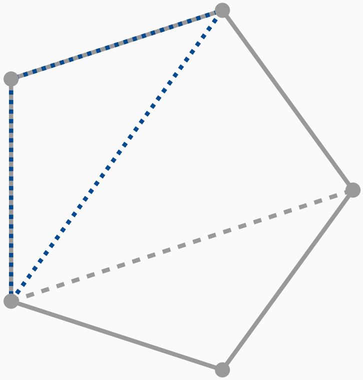
Larger cycles



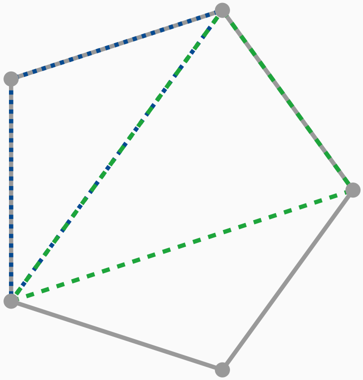
Larger cycles



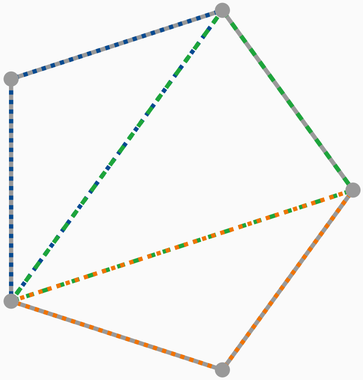
Larger cycles



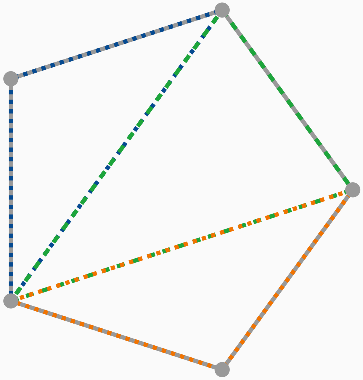
Larger cycles



Larger cycles



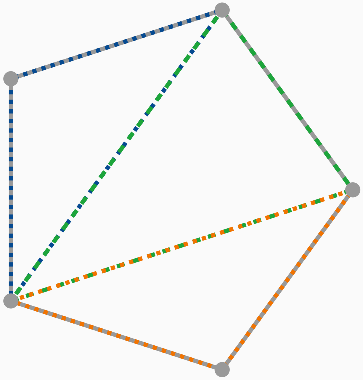
Larger cycles



The idea

Decomposing bigger cycles into smaller cycles by creating artificial branches.

Larger cycles



The idea

Decomposing bigger cycles into smaller cycles by creating artificial branches.

Two types of decomposition

Cycles can either be decomposed into 3-cycles by adding branches $(1, i)$, for $i = 3, \ldots, n$, or into 4-cycles by adding branches $(1, 2i)$, for $i = 2, \ldots, (n-2)/2$. Note that if n is odd, one 3-cycle needs to be added by creating the artificial branch $(1, n-1)$.

Excursus: multilinear optimization (I)

Multilinear problem

$$\min \sum_{I \in \mathcal{I}_0} c_I^0 \prod_{v \in I} x_v \tag{6a}$$

$$\text{s.t.} \sum_{I \in \mathcal{I}_j} c_I^j \prod_{v \in I} x_v \leq b_j \qquad \forall j \in \{1, \dots, m\}, \tag{6b}$$

$$x_v \in [l_v, u_v] \qquad \forall v \in V, \tag{6c}$$

$$\mathcal{I}_0, \mathcal{I}_1, \dots, \mathcal{I}_m \subset V,\ c_I^j, b_j \in \mathbb{R}\ , l, u \in \mathbb{R}^V$$

Excursus: multilinear optimization (I)

Multilinear problem

$$\min \sum_{I \in \mathcal{I}_0} c_I^0 \prod_{v \in I} x_v \tag{6a}$$

$$\text{s.t.} \sum_{I \in \mathcal{I}_j} c_I^j \prod_{v \in I} x_v \leq b_j \qquad \forall j \in \{1, \ldots, m\}, \tag{6b}$$

$$x_v \in [l_v, u_v] \qquad \forall v \in V, \tag{6c}$$

$$\mathcal{I}_0, \mathcal{I}_1, \ldots, \mathcal{I}_m \subset V, \; c_I^j, b_j \in \mathbb{R}\,, l, u \in \mathbb{R}^V$$

Multilinear problem: rewriting

$$\min \sum_{I \in \mathcal{I}_0} c_I^0 z_I \tag{7a}$$

$$\text{s.t.} \sum_{I \in \mathcal{I}_j} c_I^j z_I \leq b_j \qquad \forall j \in \{1, \ldots, m\}, \tag{7b}$$

$$z_I = \prod_{v \in I} x_v \qquad \forall I \in \mathcal{E} := \cup_{j=0}^m \mathcal{I}_j, \tag{7c}$$

$$x_v \in [l_v, u_v] \qquad \forall v \in V. \tag{7d}$$

Excursus: multilinear optimization (II)

It is known that there exists an optimal solution in which each x_v is at its bound, that is, $x_v \in \{l_v, u_v\}$ holds for all $v \in V$. Hence, by an affine transformation we can replace by $x_v \in \{0, 1\}$ for all $x_v \in V$.

Excursus: multilinear optimization (II)

It is known that there exists an optimal solution in which each x_v is at its bound, that is, $x_v \in \{l_v, u_v\}$ holds for all $v \in V$. Hence, by an affine transformation we can replace by $x_v \in \{0, 1\}$ for all $x_v \in V$.

The couple $(V, \mathcal{E})$ gives rise to a multilinear polytope defined as the convex hull

$$\mathrm{ML}(V, \mathcal{E}) := \mathrm{conv}\{(x, z) \in \{0,1\}^V \times \{0,1\}^{\mathcal{E}} \mid z_I = \prod_{v \in I} x_v \ \forall I \in \mathcal{E}\} \qquad (8)$$

of the multilinear set defined as the set of the solutions of (7c) with the constraint of x_v being binary.

Excursus: multilinear optimization (II)

It is known that there exists an optimal solution in which each x_v is at its bound, that is, $x_v \in \{l_v, u_v\}$ holds for all $v \in V$. Hence, by an affine transformation we can replace by $x_v \in \{0,1\}$ for all $x_v \in V$.

The couple $(V, \mathcal{E})$ gives rise to a multilinear polytope defined as the convex hull

$$\mathrm{ML}(V, \mathcal{E}) := \mathrm{conv}\{(x,z) \in \{0,1\}^V \times \{0,1\}^{\mathcal{E}} \mid z_I = \prod_{v \in I} x_v \ \forall I \in \mathcal{E}\} \tag{8}$$

of the multilinear set defined as the set of the solutions of (7c) with the constraint of x_v being binary.

As a side note, all of this can be interpreted in the setting of hypergraphs: in particular with $G = (V, \mathcal{E})$ being an hypergraph.

A first relaxation

The simplest polyhedral relaxation of $\mathrm{ML}(V, \mathcal{E})$ is the standard relaxation [SW24]

$$
\begin{aligned}
z_I &\le x_v && \forall v \in I \in \mathcal{E}, && \text{(9a)}\\
z_I + \sum_{v \in I}(1 - x_v) &\ge 1 && I \in \mathcal{E}, && \text{(9b)}\\
z_I &\ge 0 && I \in \mathcal{E}, && \text{(9c)}\\
x_v &\in [0,1] && \forall v \in V. && \text{(9d)}
\end{aligned}
$$

Unfortunately, this relaxation is often very weak and so one could be interested in finding additional constraints to tighten the relaxation.

Extended flower inequalities & Co

One possibility is adding the extended flower inequalities [SW24].

Definition 4.

Let $\mathcal{S} := \{\{v\} \mid v \in V\}$, $I \in \mathcal{E}$, and let $J_1, \ldots, J_k \in \mathcal{E} \cup \mathcal{S}$ be such that $J_1 \cup \cdots \cup J_k \supset I$ and $J_i \cap I \neq \emptyset$ hold for $i = 1, \ldots, k$. The extended flower inequality centered at I with neighbors $J_1, \ldots, J_k$ is the inequality

$$z_I + \sum_{i=1}^{k} (1 - z_{J_i}) \geq 1. \tag{10}$$

Another possibility is the recursive (McCormick) linearization [Kha06].

Affine transformations (I)

A first step could consist in applying the standard relaxation to each monomial in every loop constraint

$$\sum_{j=0}^{\lfloor n/2 \rfloor} \sum_{\substack{A \subset [n] \\ |A|=2j}} (-1)^j \prod_{h \in A} s_{k_h k_{h+1}} \prod_{l \in A^c} c_{k_l k_{l+1}} = \prod_{i=1}^{n} c_{k_i k_i}.$$

Affine transformations (I)

A first step could consist in applying the standard relaxation to each monomial in every loop constraint

$$\sum_{j=0}^{\lfloor n/2 \rfloor} \sum_{\substack{A \subset [n] \\ |A|=2j}} (-1)^j \prod_{h \in A} s_{k_h k_{h+1}} \prod_{l \in A^c} c_{k_l k_{l+1}} = \prod_{i=1}^{n} c_{k_i k_i}.$$

We have that $c_{k_i k_i} \geq 0$ and, if we suppose that the angle differences are small, we have that $c_{k_l k_{l+1}} \geq 0$. We only need to define variables

$$c'_{k_i k_i} = \frac{c_{k_i k_i}}{V_{k_i}^{\max 2}}, \quad c'_{k_l k_{l+1}} = \frac{c_{k_l k_{l+1}}}{V_{k_l}^{\max} V_{k_{l+1}}^{\max}}, \quad c'_{k_i k_i}, c'_{k_l k_{l+1}} \in [0, 1]. \tag{11}$$

Affine transformations (II)

The problem is that we can not make hypotesis on the sign of $s_{k_h k_{h+1}}$, and rescaling it in the following way

$$s'_{k_h k_{h+1}} = \frac{s_{k_h k_{h+1}} + V^{\max}_{k_h} V^{\max}_{k_{h+1}}}{2 V^{\max}_{k_h} V^{\max}_{k_{h+1}}}$$

would generate a lot more monomials in (5).

Affine transformations (II)

The problem is that we can not make hypotesis on the sign of $s_{k_h k_{h+1}}$, and rescaling it in the following way

$$s'_{k_h k_{h+1}} = \frac{s_{k_h k_{h+1}} + V_{k_h}^{\max} V_{k_{h+1}}^{\max}}{2 V_{k_h}^{\max} V_{k_{h+1}}^{\max}} \quad \longleftarrow \text{NO}$$

would generate a lot more monomials in (5). The idea is then to write

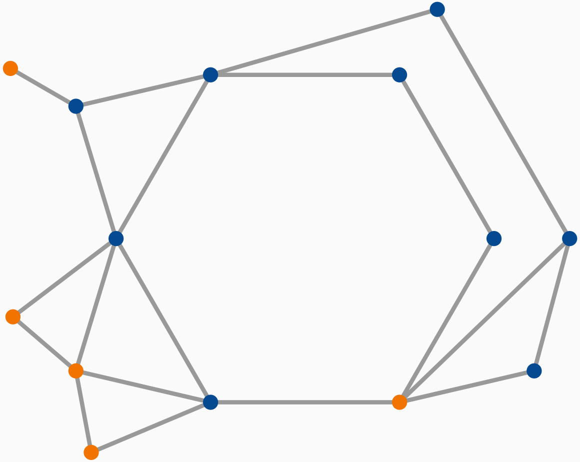
$$s_{k_h k_{h+1}} = V_{k_h}^{\max} V_{k_{h+1}}^{\max} (2\sigma_{k_h k_{h+1}} - 1) u'_{k_h k_{h+1}}, \quad \sigma_{k_h k_{h+1}} \in \{0,1\},\ u_{k_h k_{h+1}} \in [0,1]. \quad (12)$$

At this point $\prod_{ij} s_{ij} = \prod_{ij} V_i^{\max} V_j^{\max} \times \prod_{ij} (2\sigma_{ij} - 1) \times \prod_{ij} u'_{ij}$ and

$$\prod_{ij} (2\sigma_{ij} - 1) = 1 - 2r, \quad \text{where } r + 2m = \sum_{ij} (1 - \sigma_{ij}),\ r \in \{0,1\},\ m \in \mathbb{Z}. \quad (13)$$

Numerical experiment

Preliminary numerical experiment (I)



Network details

- 14 buses, 5 generators, 20 lines;
- minimum cycle basis made up of 7 loops, 5 loops of length 3 and 2 loops of length 6.

Preliminary numerical experiment (II)

	no Jabr	Jabr
no loop	116.090	5.449
loop	84.884	4.717

Table: Norm 2 error of loop constraints (5) with different constraint configurations. Results are scaled by a factor of 10^2.

We studied the impact of the Jabr inequality and our relaxation of the loop constraint on the norm 2 of the vector difference between the lhs and the rhs of (5).

Note that the error on the Jabr equality ($\cong 10^{-9}$) is negligible with respect to the error on the loop constraints when the Jabr inequality is added to the constraints.

Conclusions & future works

Conclusions

- Some models and relaxations for the OPF problem.
- The study of the exactness of the relaxation with respect to the network structure.
- Some multilinear programming techniques and a first attempt to adapt them to the linearization of the loop constraint.

Future works

- Apply the extended flower inequalities or the recursive McCormick linearization to strengthen our formulation.
- Make a comparison between our techniques and the 3-cycle / 4-cycle decomposition.
- Make use of the strong sructure of the loop constraint, investigating the link between multilinear optimization and hypergraph theory [DK17].

Fine.

For other things I do → ambrogiomb.github.io

References

[BEGL20] Bienstock, D., and Escobar, M., and Gentile, C., and Liberti, L. (2020). Mathematical programming formulations for the alternating current optimal power flow problem. *4OR*, 18(3), 249-292.

[DK17] Del Pia, A., and Khajavirad, A. (2017). A polyhedral study of binary polynomial programs. *Mathematics of Operations Research*, 42(2), 389–410.

[Jab06] Jabr, R. A. (2006). Radial distribution load flow using conic programming. *IEEE transactions on power systems*, 21(3), 1458–1459.

[Kha06] Khajavirad, A. (2023). On the strength of recursive McCormick relaxations for binary polynomial optimization. *Operations Research Letters*, 51(2), 146–152.

[KDS16] Kocuk, B. and Dey, S. S. and Sun, X. A. (2016). Strong SOCP relaxations for the optimal power flow problem. *Operations Research*, 64(6), 1177–1196.

[SW24] Schutte, E. and Walter, M. (2024). Relaxation strength for multilinear optimization: McCormick strikes back. *International Conference on Integer Programming and Combinatorial Optimization* (pp. 393-404). Cham: Springer Nature Switzerland.