Electrical and Electronics Engineering
2024-2025
Master Semester 2

Course
Smart grids technologies

Relaxation of the Optimal Power Flow problem

Prof. Mario Paolone
Distributed Electrical Systems Laboratory
École Polytechnique Fédérale de Lausanne (Switzerland)

Outline

The relaxation process

Second order cone programming

SOCP and the branch flow model

The relaxation process

In optimization theory, **relaxation methods are techniques used to simplify complex optimization problems by relaxing, or loosening, constraints**. This allows for easier computation and often leads to **approximate solutions that are still satisfactory for practical purposes**.

Relaxation methods are particularly useful when dealing with non-convex (and/or combinatorial) optimization problems where finding exact solutions is difficult or computationally expensive (sometimes impossible, like in NP-hard problems).

An important property that any relaxation process should have, is to **guarantee that the solution of the relaxed problem satisfies the constraints of the original problem**. As we will see next, such property may not hold.

The relaxation process

Let's consider the standard optimization problem (P), defined for $x \in \mathcal{D} \subset \mathbb{R}^n$, where the set $\mathcal{D}$ is the domain of definition of (P):

$$
\begin{aligned}
&(P): \min f(x) \\
&s.t. \\
&f_i(x) \leq 0, \qquad i = 1{:}\, m \\
&h_j(x) = 0, \qquad j = 1{:}\, p \\
&x \in \mathcal{D} \subset \mathbb{R}^n
\end{aligned}
$$

The **relaxation of** (P) **is to identify a new convex problem** (P^r)

$$
\begin{aligned}
&(P^r): \min f^r(x) \\
&s.t. \\
&f_i^r(x) \leq 0, \qquad i = 1{:}\, m \\
&h^r{}_j(x) = 0, \qquad j = 1{:}\, p \\
&x \in \mathcal{D}^r \subset \mathbb{R}^n
\end{aligned}
$$

where $f^r(x)$ **is convex**, $f_i^r(x) \leq 0$ and $h^r{}_j(x) = 0$ **are convex inequalities and equalities** such that

1. $\mathcal{D}^r \supseteq \mathcal{D}$
2. $f^r(x) \leq f(x), \forall x \in \mathcal{D}$

The relaxation process

$$1.\ \mathcal{D}^r \supseteq \mathcal{D}$$

$$2.\ f^r(x) \leq f(x), \forall x \in \mathcal{D}$$

The first property states that **the domain of definition of (P) is a subset of (P^r)**. The second property states that the objective function of (P) is greater than or equal to the one of (P^r) $\forall x \in \mathcal{D}$.

The consequences are:

1. if x^* is a solution of (P), then $x^* \in \mathcal{D} \subseteq \mathcal{D}^r$ and $f(x^*) \geq f^r(x^*)$, then **the optimal value of (P^r) is a lower bound for (P)**.
2. since $f(x^*) = f^r(x^*)$ $\forall x^* \in \mathcal{D}$, a solution of (P^r) is feasible for (P), then it is the optimal of (P).

The problem that may arise is that, since $\mathcal{D}^r \supseteq \mathcal{D}$, the solution of (P^r) **may produce a solution that is unfeasible for (P)**. Therefore, the usual approach is to **have penalties that guarantee that the relaxation is tight or, in other words, the solution of (P^r) satisfies the constraints of (P) with a very good approximation (ideally none).**

The relaxation process

Example #1

Let us consider the following **mixed integer linear program (MILP)**:

$$\min c^T(x,z)$$
$$s.t.$$
$$F(x,z) \leq g$$
$$A(x,z) = b$$
$$z \in \{0,1\}^q$$
$$x \in \mathbb{R}^n$$

As known, MILP problems are hard to solve in their standard form.

The simplest relaxation is to replace $z \in \{0,1\}^q$ with $z \in [0,1]^q$. In this way, the optimal value of relaxed LP is a lower bound of the MILP.

Furthermore, in order to make the relaxation tight, we may **add to the objective of the relaxed problem a function that is minimised when** $z \to 0,1$.

The relaxation process

Example #2

Let us consider the following problem

$$\min_{x,y} f(x,y)$$
$$s.t.$$
$$x^L \leq x \leq x^U$$
$$y^L \leq y \leq y^U$$
$$x \cdot y = w$$

In this problem, we minimise a cost function $f(x,y)$ supposed to be convex, associated to two generic products x and y, while satisfying min-max bounds of both products and the **interaction among them captured by the bilinear constraint that makes the problem non-convex.**

The relaxation process

Example #2 cont'd

A way to relax and convexify the problem is to use the so-called McCormick relaxation of $x \cdot y = w$ given by four linear inequalities.

Let us introduce two auxiliary variables, say a, b, that are **always positive by construction with respect to the upper and lower bounds** $x^L, x^U, y^L y^U$:

Under-estimators

$$a = (x - x^L), b = (y - y^L)$$

$$a \cdot b \geq 0 \rightarrow (x - x^L) \cdot (y - y^L) = xy - x^L y - xy^L + x^L y^L \geq 0$$

$$w \geq x^L y + xy^L - x^L y^L$$

$$a = (x^U - x), b = (y^U - y)$$

$$a \cdot b \geq 0 \rightarrow (x^U - x) \cdot (y^U - y) = x^U y^U - yx^U - xy^U + xy \geq 0$$

$$w \geq xy^U + yx^U - x^U y^U$$

Over-estimators

$$a = (x^U - x), b = (y - y^L)$$

$$a \cdot b \geq 0 \rightarrow (x^U - x) \cdot (y - y^L) = x^U y - xy - x^U y^L + xy^L \geq 0$$

$$w \leq x^U y - x^U y^L + xy^L$$

$$a = (x - x^L), b = (y^U - y)$$

$$a \cdot b \geq 0 \rightarrow (x - x^L) \cdot (y^U - y) = xy^U - xy - x^L y^U + x^L y \geq 0$$

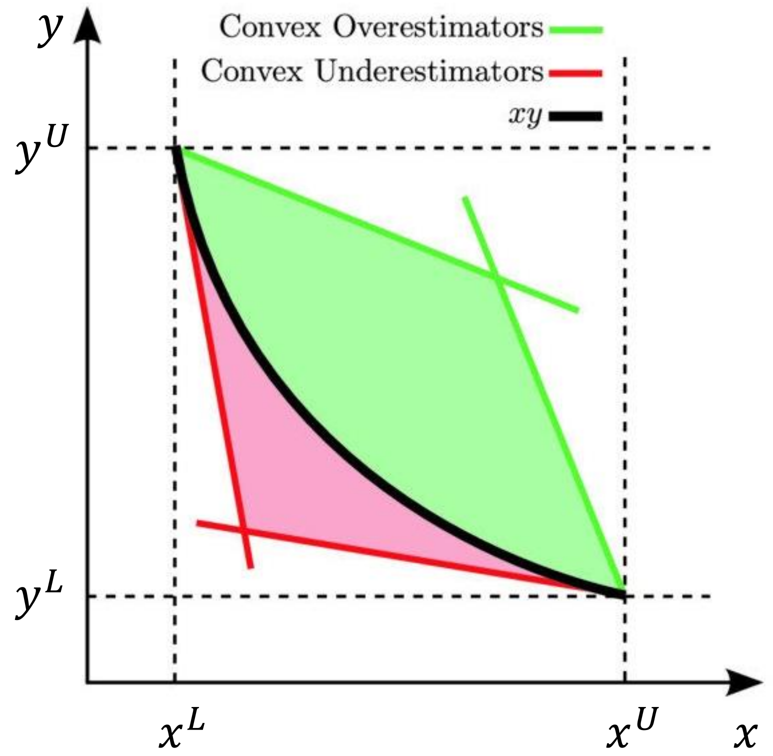
$$w \leq xy^U - x^L y^U + x^L y$$

The relaxation process

Example #2 cont'd

The relaxed and convex problem is

$$\min_{x,y} f(x,y)$$
$$s.t.$$
$$x^L \leq x \leq x^U$$
$$y^L \leq y \leq y^U$$
$$w \geq x^L y + x y^L - x^L y^L$$
$$w \geq x y^U + y x^U - x^U y^U$$
$$w \leq x^U y - x^U y^L + x y^L$$
$$w \leq x y^U - x^L y^U + x^L y$$



Clearly, in this example **the domain of the relaxed problem include the one of the non-relaxed one**: $\mathcal{D}^r \supseteq \mathcal{D}$ **and the solution of the relaxed problem may be unfeasible for the non-relaxed one.**

Outline

The relaxation process

Second order cone programming

SOCP and the branch flow model

Second order cone programming

Let's start with a simple example: the **second-order cone**, or **quadratic cone**, it is given by the elements of $(x, y, z) \in \mathbb{R}^3$ satisfying this condition:

$$\sqrt{x^2 + y^2} < z$$

Now, **let us consider** z **to be a parameter, say** t, we can generalize the above inequality to a second-order cone of dimension $n + 1$:

$$\mathcal{C}_{n+1}: \left\{ \begin{bmatrix} x \\ t \end{bmatrix} \middle| \, \|x\|_2 \leq t \right\}$$

$$x \in \mathbb{R}^n, t \in \mathbb{R}$$

This constraint for x is convex and we can further generalise it as:

$$\|Ax + b\|_2 \leq c^T x + d$$

$$x, c \in \mathbb{R}^n$$

$$A \in \mathbb{R}^{n+1 \times n}$$

$$b \in \mathbb{R}^{n+1}$$

$$d \in \mathbb{R}$$

Second order cone programming

Therefore, an optimisation problem with the following form, is called **second-order cone programming (SOCP)**:

$$\min_x h^T x$$
$$s.t.$$
$$\|A_i x + b_i\|_2 \le c_i^T x + d_i, \qquad i = 1, \dots, m$$

where

$$x, f, c \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{n_i \times n}$$
$$b \in \mathbb{R}^{n_i}$$
$$d \in \mathbb{R}$$

The above problem is convex. In what follows, we show how we can **relax the OPF in the form of a SOCP problem**.

Outline

The relaxation process

Second order cone programming

SOCP and the branch flow model

SOCP and the branch flow model

Let us consider a power system whose **topology is identified by a connected graph** $\mathcal{G} = (\mathcal{S}, \mathcal{L})$ where $\mathcal{S}$ **represents the set of nodes**, and $\mathcal{L}$ **the set of branches**:

$\mathcal{S} = \{1, \dots, s\}$ where s is the **total number of buses**;

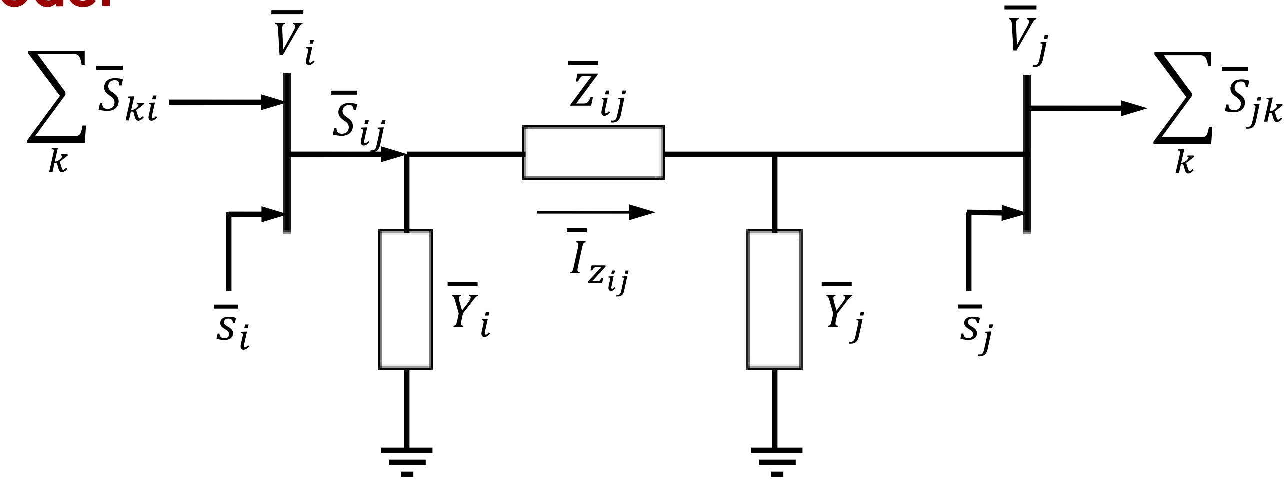
$\mathcal{L} = \{1, \dots, l\}$ where l is the **total number of branches**. Naturally, a branch between buses i and j is an element in $\mathcal{L}$: $(i, j) \in \mathcal{L}$.

Trivially, the considered power system is called **radial if its graph** $\mathcal{G}$ **is a tree**. For distribution networks, which are typically radial, the root of the tree (node 1) represents the substation bus and, usually, the slack.

For a generally meshed transmission network, node 1 represents the slack bus being the largest power plant (or equivalent) in the system.

SOCP and the branch flow model

Let us recall the **branch flow model**: consider a **generic branch between nodes i and j of the network modelled by a generic Π-equivalent model**



$\overline{V}_i, \overline{V}_j$: complex nodal voltages respectively at nodes i and j

$\overline{s}_i, \overline{s}_j$: complex apparent power injections respectively at nodes i and j

$\overline{S}_{ij}$: complex power flow from node i to node j

$\overline{I}_{z_{ij}}$: complex current flow through the branch impedance $\overline{Z}_{ij}$

$\overline{Y}_i, \overline{Y}_j$: complex shunt admittances respectively at nodes i and j

$\sum_k \overline{S}_{ki}$: complex apparent power entering into node i from the rest of the grid

$\sum_k \overline{S}_{jk}$: complex apparent power leaving node j towards the rest of the grid

SOCP and the branch flow model

By **introducing two variables composed by the squares of magnitudes of nodal voltages and currents**:

$$v_i = \left|\overline{V}_i\right|^2, v_j = \left|\overline{V}_j\right|^2$$

$$i_{z_{ij}} = \left|\overline{I}_{z_{ij}}\right|^2$$

we got the **branch flow model**

$$\overline{S}_{ij} - \overline{Z}_{ij} i_{z_{ij}} - \overline{Y}_i v_i - \overline{Y}_j v_j + \overline{s}_j = \sum_k \overline{S}_{jk}$$

$$\sum_k \overline{S}_{ki} + \overline{s}_i = \overline{S}_{ij}$$

$$v_j = v_i + \left|\overline{Z}_{ij}\right|^2 i_{z_{ij}} - 2\Re\left[\underline{Z}_{ij}\left(\overline{S}_{ij} - \overline{Y}_i v_i\right)\right]$$

$$i_{z_{ij}} = \frac{\left|\overline{S}_{ij} - \overline{Y}_i v_i\right|^2}{v_i}$$

Let us use the first constraint and **separate the nodal injected active and reactive powers in node j since they are the OPF decision variables.**

SOCP and the branch flow model

We have for a generic branch $(i,j) \in \mathcal{L}$

$$P_j = \sum_k P_{jk} - P_{ij} + R_{ij} i_{z_{ij}} + G_i v_i + G_j v_j$$

$$Q_j = \sum_k Q_{jk} - Q_{ij} + X_{ij} i_{z_{ij}} + B_i v_i + B_j v_j$$

along with the two other constraints of the branch flow model

$$v_j = v_i + \left|\overline{Z}_{ij}\right|^2 i_{z_{ij}} - 2\Re\left[\underline{Z}_{ij}\left(\overline{S}_{ij} - \overline{Y}_i v_i\right)\right]$$

$$i_{z_{ij}} = \frac{\left|\overline{S}_{ij} - \overline{Y}_i v_i\right|^2}{v_i}$$

The first three constraints are linear and **the non-convexity of the branch flow model is only due to the last one**. Let's see how we can **relax this equality by means of the SOCP**.

SOCP and the branch flow model

It is easy to show that the constraint

$$i_{z_{ij}} = \frac{\left|\overline{S}_{ij} - \overline{Y}_i v_i\right|^2}{v_i} = \frac{\left(P_{ij} - G_i v_i\right)^2 + \left(Q_{ij} - B_i v_i\right)^2}{v_i}$$

can be rewritten as follows:

$$\left\| \begin{matrix} 2\left(P_{ij} - G_i v_i\right) \\ 2\left(Q_{ij} - B_i v_i\right) \\ i_{z_{ij}} - v_i \end{matrix} \right\|_2 = i_{z_{ij}} + v_i$$

Indeed, we have the following:

$$\sqrt{4\left(P_{ij} - G_i v_i\right)^2 + 4\left(Q_{ij} - B_i v_i\right)^2 + \left(i_{z_{ij}} - v_i\right)^2} = \left(i_{z_{ij}} + v_i\right)$$

$$4\left(P_{ij} - G_i v_i\right)^2 + 4\left(Q_{ij} - B_i v_i\right)^2 + i_{z_{ij}}^2 + v_i^2 - 2 i_{z_{ij}} v_i = i_{z_{ij}}^2 + v_i^2 + 2 i_{z_{ij}} v_i$$

$$\left(P_{ij} - G_i v_i\right)^2 + \left(Q_{ij} - B_i v_i\right)^2 = i_{z_{ij}} v_i$$

SOCP and the branch flow model

It is interesting to note that the constraint

$$\left\| \begin{matrix} 2(P_{ij} - G_i v_i) \\ 2(Q_{ij} - B_i v_i) \\ i_{z_{ij}} - v_i \end{matrix} \right\|_2 = i_{z_{ij}} + v_i$$

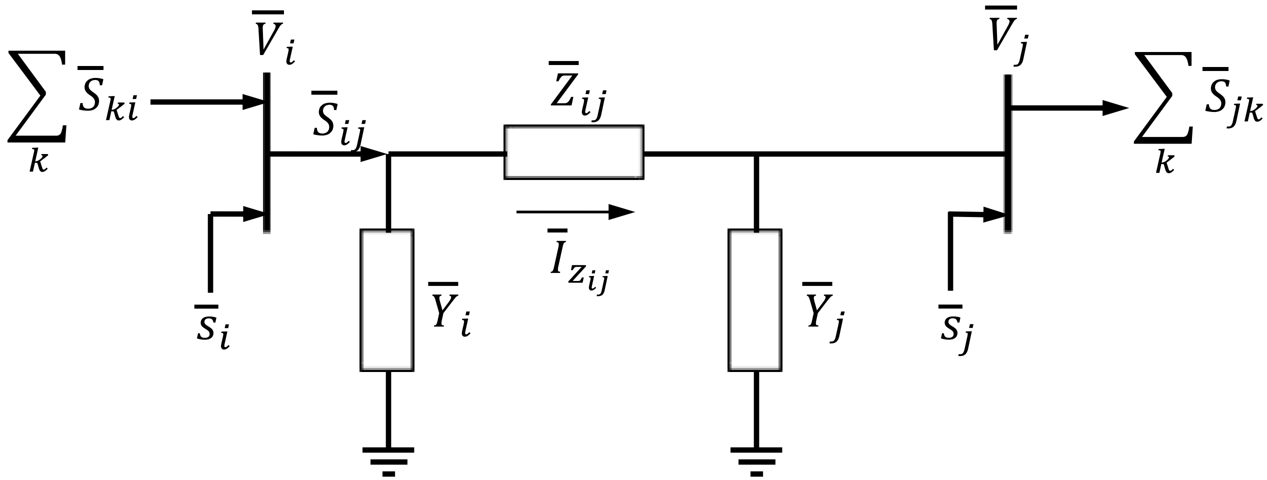
can be written in the general form of a second-order cone inequality $\|A_i x + b_i\|_2 \leq c_i^T x + d_i$

$$\left\| \begin{matrix} 2(P_{ij} - G_i v_i) \\ 2(Q_{ij} - B_i v_i) \\ i_{z_{ij}} - v_i \end{matrix} \right\|_2 \leq i_{z_{ij}} + v_i$$

or, in other terms, can be written **by relaxing $i_{z_{ij}}$ (i.e, we relax the branch losses $\overline{Z}_{ij} i_{z_{ij}}$ being $\overline{Z}_{ij}$ a parameter)**:

$$i_{z_{ij}} \geq \frac{\left|\overline{S}_{ij} - \overline{Y}_i v_i\right|^2}{v_i}$$

SOCP and the branch flow model



There is now the need to **satisfy the branch power/ampacity limit**.

In the case we apply a **constraint to the branch power**, we need to have:

$$P_{ij}^2 + Q_{ij}^2 \leq \left(S_{ij}^{max}\right)^2$$

It is interesting to note that the above constraint is also a second-order cone, so it fits the SOCP.

In case we need to **constraint the branch current module**, we may **neglect the current flowing through the shunt** $\overline{Y}_i$ and constraint the $\overline{I}_{z_{ij}}$:

$$\left|\overline{I}_{ij}\right|^2 \approx i_{z_{ij}}(t) \leq \left(I_{ij}^{max}\right)^2$$

SOCP and the branch flow model

The **SOCP-relaxed OPF** is the following:

$$\min_{\substack{P_{g_2}(t),\dots,P_{g_g}(t),Q_{g_2}(t),\dots,Q_{g_g}(t)\\ P_{s_1}(t),\dots,P_{s_m}(t),Q_{s_1}(t),\dots,Q_{s_m}(t)}} \sum_{t=1}^{T_{max}} \sum_{i=1}^{g} C_i\left(P_{g_i}(t), Q_{g_i}(t)\right) + \sum_{i=1}^{m} C_i\left(P_{s_i}(t)\right)$$

$s.t.$

$P_j(t) = \sum_k P_{jk}(t) - P_{ij}(t) + R_{ij} i_{z_{ij}}(t) + G_i v_i(t) + G_j v_j(t), (i,j) \in \mathcal{L}$

$Q_j(t) = \sum_k Q_{jk}(t) - Q_{ij}(t) + X_{ij} i_{z_{ij}}(t) + B_i v_i(t) + B_j v_j(t), (i,j) \in \mathcal{L}$

$\sum_k P_{ki}(t) + P_i(t) = P_{ij}(t), (i,j) \in \mathcal{L}$

$\sum_k Q_{ki}(t) + Q_i(t) = Q_{ij}(t), (i,j) \in \mathcal{L}$

$P_j(t) = P_{g_j}(t) + P_{l_j}(t), +P_{s_j}(t), j \in \mathcal{S}$

$Q_j(t) = Q_{g_j}(t) + Q_{l_j}(t) + Q_{s_j}(t), j \in \mathcal{S}$

$v_j(t) = v_i(t) + \left|\overline{Z}_{ij}\right|^2 i_{z_{ij}}(t) - 2\Re\left[\underline{Z}_{ij}\left(\overline{S}_{ij}(t) - \overline{Y}_i v_i(t)\right)\right], (i,j) \in \mathcal{L}$

$P_{g_j}^{min} \le P_{g_j}(t) \le P_{g_j}^{max}, j = 1, \dots, g$

$Q_{g_j}^{min} \le Q_{g_j}(t) \le Q_{g_j}^{max}, j = 1, \dots, g$

$P_{s_j}^{min} \le P_{s_j}(t) \le P_{s_j}^{max}, j = 1, \dots, m$

$Q_{s_j}^{min} \le Q_{s_j}(t) \le Q_{s_j}^{max}, j = 1, \dots, m$

$\left|\overline{V}_1\right| = 1pu$

$V_{min}^2 \le v_j \le V_{max}^2, j \in \mathcal{S}$

$i_{z_{ij}}(t) \ge \dfrac{\left|\overline{S}_{ij}(t) - \overline{Y}_i v_i(t)\right|^2}{v_i(t)}, (i,j) \in \mathcal{L}$

$P_{ij}^2(t) + Q_{ij}^2(t) \le \left(S_{ij}^{max}\right)^2 or\ i_{z_{ij}}(t) \le \left(I_{ij}^{max}\right)^2, (i,j) \in \mathcal{L}$

$SoC_j(t+1) = SoC_j(t) + P_{s_j}(t+1)\Delta t,\ j = 1, \dots, m$

$SoC_j^{min} \le SoC_j(t+1) \le SoC_j^{max}, j = 1, \dots, m$

$\xi_{g_j}^{min} \le P_{g_j}(t+1) - P_{g_j}(t) \le \xi_{g_j}^{max},\ j = 1, \dots, g$

Where we have considered to control

- g generators;
- m energy storage devices.

Remember that in these constraints the voltage angles are not present and have to be retrieved a-posteriori (see lecture 2.4).

SOCP and the branch flow model

In view of the relaxation applied to the branch losses $\overline{Z}_{ij} i_{z_{ij}}$, **the SOCP-OPF does not satisfy the load flow equations (i.e., the relaxation is not tight).** In other terms, to satisfy the constraint

$$i_{z_{ij}}(t) \geq \frac{\left|\overline{S}_{ij}(t) - \overline{Y}_i v_i(t)\right|^2}{v_i(t)}, (i,j) \in \mathcal{L}$$

the SOCP OPF may "inflate" $i_{z_{ij}}(t)$ **up to the branch limits in power or current.**

A way to **force the branch losses relaxation to be tight, is to introduce a penalty term on the growth of** $i_{z_{ij}}(t)$. Fortunately, in the case of the OPF, this corresponds to **minimise the power systems losses (i.e., a desirable objective):**

$$\sum_{(i,j)\in\mathcal{L}} R_{ij} i_{z_{ij}}(t)$$

SOCP and the branch flow model

The **SOCP-relaxed OPF** with grid losses:

$$\min_{\substack{P_{g_2}(t),\dots,P_{g_g}(t),Q_{g_2}(t),\dots,Q_{g_g}(t)\\ P_{s_1}(t),\dots,P_{s_m}(t),Q_{s_1}(t),\dots,Q_{s_m}(t)}} \sum_{t=1}^{T_{max}} \sum_{i=1}^{g} C_i\left(P_{g_i}(t), Q_{g_i}(t)\right) + \sum_{i=1}^{m} C_i\left(P_{s_i}(t)\right) + \sum_{(i,j)\in\mathcal{L}} R_{ij} i_{z_{ij}}(t)$$

$s.t.$

$$P_j(t) = \sum_k P_{jk}(t) - P_{ij}(t) + R_{ij} i_{z_{ij}}(t) + G_i v_i(t) + G_j v_j(t), (i,j) \in \mathcal{L}$$

$$Q_j(t) = \sum_k Q_{jk}(t) - Q_{ij}(t) + X_{ij} i_{z_{ij}}(t) + B_i v_i(t) + B_j v_j(t), (i,j) \in \mathcal{L}$$

$$\sum_k P_{ki}(t) + P_i(t) = P_{ij}(t), (i,j) \in \mathcal{L}$$

$$\sum_k Q_{ki}(t) + Q_i(t) = Q_{ij}(t), (i,j) \in \mathcal{L}$$

$$P_j(t) = P_{g_j}(t) + P_{l_j}(t), +P_{s_j}(t), j \in \mathcal{S}$$

$$Q_j(t) = Q_{g_j}(t) + Q_{l_j}(t) + Q_{s_j}(t), j \in \mathcal{S}$$

$$v_j(t) = v_i(t) + \left|\overline{Z}_{ij}\right|^2 i_{z_{ij}}(t) - 2\Re\left[\underline{Z}_{ij}\left(\overline{S}_{ij}(t) - \overline{Y}_i v_i(t)\right)\right], (i,j) \in \mathcal{L}$$

$$P_{g_j}^{min} \le P_{g_j}(t) \le P_{g_j}^{max}, j = 1, \dots, g$$

$$Q_{g_j}^{min} \le Q_{g_j}(t) \le Q_{g_j}^{max}, j = 1, \dots, g$$

$$P_{s_j}^{min} \le P_{s_j}(t) \le P_{s_j}^{max}, j = 1, \dots, m$$

$$Q_{s_j}^{min} \le Q_{s_j}(t) \le Q_{s_j}^{max}, j = 1, \dots, m$$

$$\left|\overline{V}_1\right| = 1pu$$

$$V_{min}^2 \le v_j \le V_{max}^2, j \in \mathcal{S}$$

$$i_{z_{ij}}(t) \ge \frac{\left|\overline{S}_{ij}(t) - \overline{Y}_i v_i(t)\right|^2}{v_i(t)}, (i,j) \in \mathcal{L}$$

$$P_{ij}^2(t) + Q_{ij}^2(t) \le \left(S_{ij}^{max}\right)^2 or \left|\overline{I}_{ij}\right|^2 \approx i_{z_{ij}}(t) \le \left(I_{ij}^{max}\right)^2, (i,j) \in \mathcal{L}$$

$$SoC_j(t+1) = SoC_j(t) + P_{s_j}(t+1)\Delta t,\ j = 1, \dots, m$$

$$SoC_j^{min} \le SoC_j(t+1) \le SoC_j^{max}, j = 1, \dots, m$$

$$\xi_{g_j}^{min} \le P_{g_j}(t+1) - P_{g_j}(t) \le \xi_{g_j}^{max},\ j = 1, \dots, g$$

Where we have considered to control

- g generators;
- m energy storage devices.

Remember that in these constraints the voltage angles are not present.

SOCP and the branch flow model

IMPORTANT: remember that the constraint the branch current module has neglected the current flowing through the shunt $\overline{Y}_i$

$$\left|\overline{I}_{ij}\right|^2 \approx i_{z_{ij}}(t) \leq \left(I_{ij}^{max}\right)^2$$

Therefore, the **solution of the SOCP-OPF may still not satisfy with exactness the branches ampacity limits.**

For those interested to the subject, there exist more sophisticated relaxations that have provided exact solutions of the SOCP-OPF for the case of radial power systems [1] and approximated ones for the case of meshed systems [2].

[1] M. Nick, R. Cherkaoui, J. -Y. L. Boudec and M. Paolone, "An Exact Convex Formulation of the Optimal Power Flow in Radial Distribution Networks Including Transverse Components," in IEEE Transactions on Automatic Control, vol. 63, no. 3, pp. 682-697, March 2018.

[2] Z. Yuan and M. Paolone, "Properties of convex optimal power flow model based on power loss relaxation,", Electric Power Systems Research, vol. 186, 2020, 106414.