Approximating Feasible Power Injection Regions of Radial AC Networks via Dual SOCP

Yue Chen¹, Changhong Zhao²

1. Mechanical and Automation Engineering
2. Information Engineering
The Chinese University of Hong Kong (CUHK)

ArXiv: 2109.02294. Funded by Hong Kong Research Grants Council (ECS No. 24210220)
Feasible power injection regions in AC networks

Nonlinear, nonconvex, implicit (intertwined with voltages, currents, line power flows, etc.)

A simple closed-form approximation is important for resilient grid applications, e.g.,
- Solve OPF quickly for fast-timescale control in grid restoration
- Decide hosting capacities of renewable energy sources

“Feasible”: the power injections and their associated voltages, currents, etc. satisfy:
- Physical laws of circuit “solvability”
- Operational limits “safety”
Prior efforts

DC approximation (convex polyhedral regions)
• Simple computation
• DC is coarse for distribution networks

AC solvability proved by fixed point theorems
- S. Bolognani, S. Zampieri, 2015. (Banach)
- C. Wang, A. Bernstein, J.-Y. Le Boudec, M. Paolone, 2016. (Banach)
- K. Dvijotham, H. Nguyen, K. Turitsyn, 2017. (Brouwer)
- J. W. Simpson-Porco, 2017. (Brouwer)
• Accurate AC models, reliable results
• How to incorporate safety limits?

Convex optimization for inner approximations
- M. Nick, R. Cherkaoui, J.-Y. Le Boudec, M. Paolone, 2017. (Tightened-relaxed SOC)
- N. Nazir, M. Almassalkhi, 2019. (Constant estimates for nonlinear terms)
• Both solvability and safety are addressed
• Explore feasible region in a specific shape/direction of the power-injection vector

A review:
Molzahn, Hiskens, 2017
[Chapter 4.5]
This work

A **closed-form polyhedral** approximation of feasible power injection regions in radial AC networks

- Simple form and moderate computation
- Built through *dual* second-order cone program (SOCP), a *convex* program that preserves nonlinearity of AC power flow
- Fulfills both *solvability and safety*
- No need to specify a shape/direction of power-injection vector
Problem statement

AC dist-flow equations for a radial network
(Solvability): [Baran, Wu, 1989]

\[ \forall i \rightarrow j : \quad P_{ij} - r_{ij} \ell_{ij} - \sum_{k:j \rightarrow k} P_{jk} + p_j = 0 \quad (1a) \]

\[ Q_{ij} - x_{ij} \ell_{ij} - \sum_{k:j \rightarrow k} Q_{jk} + q_j = 0 \quad (1b) \]

\[ v_i - v_j - 2(r_{ij} P_{ij} + x_{ij} Q_{ij}) + (r_{ij}^2 + x_{ij}^2) \ell_{ij} = 0 \quad (1c) \]

\[ P_{ij}^2 + Q_{ij}^2 - v_i \ell_{ij} = 0. \quad (1d) \]

Active/reactive power injections: \((p, q) \in \mathbb{R}^{2N}\)

State \(x := (v, \ell, P, Q) \in \mathbb{R}^{4N}\)

\(N = \) number of lines
\(= \) number of nodes excluding root/slack node

Safety limits:

\[ v_i \leq \overline{v}_i, \quad \forall i = 1, \ldots, N \quad (2a) \]

\[ 0 \leq \ell_{ij} \leq \overline{\ell}_{ij}, \quad \forall i \rightarrow j \quad (2b) \]

\(\cdots\) nodal voltage magnitudes
\(\cdots\) on-line current magnitudes
Problem statement

AC dist-flow equations for a radial network
(Solvability): [Baran, Wu, 1989]

\[ \forall i \to j : \quad P_{ij} - r_{ij} \ell_{ij} - \sum_{k:j \to k} P_{jk} + p_j = 0 \]  \hspace{1cm} (1a)

\[ Q_{ij} - x_{ij} \ell_{ij} - \sum_{k:j \to k} Q_{jk} + q_j = 0 \]  \hspace{1cm} (1b)

\[ v_i - v_j - 2(r_{ij} P_{ij} + x_{ij} Q_{ij}) + (r_{ij}^2 + x_{ij}^2) \ell_{ij} = 0 \]  \hspace{1cm} (1c)

\[ P_{ij}^2 + Q_{ij}^2 - v_i \ell_{ij} = 0. \]  \hspace{1cm} (1d)

Safety limits:

\[ v_i \leq v_i \leq \bar{v}_i, \quad \forall i = 1, ..., N \]  \hspace{1cm} (2a)

\[ 0 \leq \ell_{ij} \leq \bar{\ell}_{ij}, \quad \forall i \to j \]  \hspace{1cm} (2b)

Let \((p, q) = (d, u) \in \mathbb{R}^{2N}, \quad 2N = D + U\)

- Known constant injections: \(d \in \mathbb{R}^D\)
- Unknown variable injections: \(u \in \mathbb{R}^U\)

A power-injection vector \(u\) is \textit{feasible} if there exists \(x = (v, \ell, P, Q)\) such that \((x; d, u) = (x; p, q)\) satisfies (1)(2).

The \textit{feasible power injection region} is

\[ \mathcal{U} := \{ u \in \mathbb{R}^U \mid u \text{ is feasible.} \} \]

Our goal: find a closed-form approximation of \(\mathcal{U}\)
Problem statement

AC dist-flow equations for a radial network
(Solvability): [Baran, Wu, 1989]

\[ \forall i \to j : \quad P_{ij} - r_{ij} E_{ij} - \sum_{k : j \to k} P_{jk} + p_j = 0 \quad (1a) \]
\[ Q_{ij} - x_{ij} E_{ij} - \sum_{k : j \to k} Q_{jk} + q_j = 0 \quad (1b) \]
\[ v_i - v_j - 2(r_{ij} P_{ij} + x_{ij} Q_{ij}) + (r_{ij}^2 + x_{ij}^2) E_{ij} = 0 \quad (1c) \]
\[ P_{ij}^2 + Q_{ij}^2 - v_i E_{ij} = 0. \quad (1d) \]

Safety limits:

\[ v_i \leq v_i \leq \bar{v}_i, \quad \forall i = 1, \ldots, N \quad (2a) \]
\[ 0 \leq E_{ij} \leq \bar{E}_{ij}, \quad \forall i \to j \quad (2b) \]

Feasibility problem for \( u \):

\[ \text{FP}(u) : \min \quad 1^T \bar{z} \]
\[ \text{over} \quad x = (v, E, P, Q), \quad \bar{z} = (z_s, z_q, \bar{z}_q) \geq 0 \]
\[ \text{s. t.} \quad A_f x + B_f u + \gamma_f = 0 \]
\[ A_s x + \gamma_s \leq z_s \]
\[ P_{ij}^2 + Q_{ij}^2 - v_i E_{ij} \leq z_{q,i}, \quad \forall i \to j \]
\[ v_i E_{ij} - (P_{ij}^2 + Q_{ij}^2) \leq \bar{z}_{q,i}, \quad \forall i \to j \]

An equivalent definition of the feasible power injection region:

\[ u = \{ u \in \mathbb{R}^U \mid \text{fp}(u) = 0 \} \]

where \( \text{fp}(u) \) is the min. obj. val. of \( \text{FP}(u) \)
Step 1: Convex relaxation of feasibility problem

Feasibility problem for $u$:

$$\text{FP}(u) : \min 1^T \tilde{z}$$

over $x = (v, \ell, P, Q), \quad \tilde{z} = (z_s, z_q, \bar{z}_q) \geq 0$

s. t. $A_f x + B_f u + \gamma_f = 0$

$A_s x + \gamma_s \leq z_s$

$P_{ij}^2 + Q_{ij}^2 - v_i \ell_{ij} \leq z_{q,ij}, \; \forall i \rightarrow j$

$$v_i \ell_{ij} - (P_{ij}^2 + Q_{ij}^2) \leq \bar{z}_{q,ij}, \; \forall i \rightarrow j$$

An equivalent definition of the feasible power injection region:

$$U = \{ u \in \mathbb{R}^U \mid \text{fp}(u) = 0\}$$

where $\text{fp}(u)$ is the min. obj. val. of $\text{FP}(u)$

SOCP relaxation:

$$\text{FP}'(u) : \min 1^T z$$

over $x, \; y, \quad z = (z_s, z_q) \geq 0$

s. t. $A_f x + B_f u + \gamma_f = 0$

$A_s x + \gamma_s \leq z_s$

$y = A_y x + b_y$

$$\|y_{ij}\|_2 \leq c_{q,ij} x + \gamma_{q,ij} + z_{q,ij}, \; \forall i \rightarrow j$$

SOCP-relaxed feasible region:

$$U' := \{ u \in \mathbb{R}^U \mid \text{fp}'(u) = 0\}$$

where $\text{fp}'(u)$ is the min. obj. val. of $\text{FP}'(u)$
Step 2: Dual SOCP

SOCP relaxation:

\[ \text{FP}'(u) : \min 1^Tz \quad \text{Slack variables} \]
\[ \text{over } x, y, z = (z_s, z_q) \geq 0 \]
\[ \text{s. t. } A_f x + B_f u + \gamma_f = 0 \]
\[ A_s x + \gamma_s \leq z_s \]
\[ y = A_y x + b_y \]
\[ \|y_{ij}\|_2 \leq c_{q,ij} x + \gamma_{q,ij} + z_{q,ij}, \forall i \rightarrow j \]

**SOCP-relaxed** feasible region:

\[ \mathcal{U}' := \{ u \in \mathbb{R}^U \mid \text{fp}'(u) = 0 \} \]

where \(\text{fp}'(u)\) is the min. obj. val. of \(\text{FP}'(u)\)

Dual SOCP:

\[ \text{DP}'(u) : \max_{\mu, \lambda} \mu_f^T (B_f u + \gamma_f) + \lambda_s^T \gamma_s - \mu_y^T b_y - \lambda_q^T \gamma_q \]
\[ \text{s. t. } 0 \leq \lambda \leq 1 \]
\[ A_f^T \mu_f + A_s^T \lambda_s = A_y^T \mu_y + c_q^T \lambda_q \]
\[ \|\mu_{y,ij}\|_2 \leq \lambda_{q,ij}, \forall i \rightarrow j \]

Slater’s condition, i.e., (strict) feasibility holds for \(\text{FP}'(u)\) \quad \textbf{Strong duality}

An equivalent definition of the **SOCP-relaxed** feasible region:

\[ \mathcal{U}' = \{ u \in \mathbb{R}^U \mid dp'(u) = 0 \} \]
\[ = \{ u \in \mathbb{R}^U \mid D_u(\mu, \lambda) \leq 0, \forall (\mu, \lambda) \text{ satisfying (6)} \} \]

where \(dp'(u)\) is the max. obj. val. of \(\text{DP}'(u)\).

\(\mathcal{U}'\) is convex.
Step 3: Closed-form approximation of $\mathcal{U}'$

**Algorithm 1: Approximate relaxed feasible region $\mathcal{U}'$**

1. **Initialization:** $\mathcal{U}'_{poly} = \{ u \in \mathbb{R}^U | u \leq \overline{u} \}$ for sufficiently low $\underline{u}$ and high $\overline{u}$; $\mathcal{V}_{safe} = \emptyset$; $c = 0$;
2. update vertices $\mathcal{V}(\mathcal{U}'_{poly})$. Let $dp'_{max} = 0$;
3. for $u \in \mathcal{V}(\mathcal{U}'_{poly})$ and $u \notin \mathcal{V}_{safe}$ do
   solve $DP'(u)$ to obtain an optimal solution $(\mu^*, \lambda^*)$ and maximum objective value $dp'(u)$;
   if $dp'(u) > dp'_{max}$ then
     $dp'_{max} \leftarrow dp'(u)$;
     $(\mu_{max}, \lambda_{max}) \leftarrow (\mu^*, \lambda^*)$;
   else if $dp'(u) \leq 0$ then $\mathcal{V}_{safe} = \mathcal{V}_{safe} \cup \{u\}$;
4. end
5. if $dp'_{max} = 0$ or $c = C_{max}$ then
   return $\mathcal{U}'_{poly}$;
6. else
   add to $\mathcal{U}'_{poly}$ a cutting plane:
   $\mu_{f,\max}^T (B_f u + \gamma_f) + \lambda_{\max}^T \gamma_u \leq 0$;
   $\mu_{y,\max}^T b_y + \lambda_{\max}^T \gamma_q$;
   $c \leftarrow c + 1$;
   go back to Line 2;
7. end

**Main idea of Alg. 1:**

- Start with an over-estimate convex polyhedron;
- Traverse its vertices; for each vertex $u$, solve $DP'(u)$; if $dp'(u) \leq 0$, then $u \in \mathcal{U}'$ and is not checked again.
- Record the vertex $u$ with highest $dp'(u) > 0$, i.e., violating $\mathcal{U}'$ the most; add a cutting plane to remove this $u$; update polyhedron and vertices;
- Terminate Alg. 1 if $dp'(u) \leq 0$ for all vertices $u$ (or if maximum number of iterations is reached); otherwise Repeat.

$$U' = \{ u \in \mathbb{R}^U \mid dp'(u) = 0 \} = \{ u \in \mathbb{R}^U \mid \mathcal{D}_u(\mu, \lambda) \leq 0, \forall (\mu, \lambda) \text{ satisfying (6)} \}$$
Step 3: Closed-form approximation of $\mathcal{U}'$  

**Proposition 3.** The output $\mathcal{U}'_{\text{poly}}$ in an arbitrary iteration of Algorithm 1 is an outer approximation of $\mathcal{U}'$.  

$dp'(u) = 0$, i.e., $u \in \mathcal{U}'$, for all vertices $u$  

**Proposition 4.** If Algorithm 1 terminates with $dp'_{\text{max}} = 0$ in a finite number of iterations, then it returns a convex polyhedron $\mathcal{U}'_{\text{poly}} = \mathcal{U}'$.  

**Corollary:** If $\mathcal{U}'$ is not a polyhedron, then Algorithm 1 cannot terminate in a finite number of iterations with $dp'_{\text{max}} = 0$.  

Step 4: Removing **SOCP-inexact** injections from $\mathcal{U}'$

**Proposition 5.** For every $u' \in \mathcal{U}'$, there must be $u \leq u'$ (element-wise), such that $u \in \mathcal{U}$.

*Consistent with the “load over-satisfaction” condition for exact SOCP relaxation of OPF*

**Definition 2.** A power injection $u \in \mathcal{U}'$ is **SOCP-inexact**, if every optimal solution of $\mathcal{F}^{'}(u)$ satisfies:

$$\|y_{ij}\|_2 < c_{q,ij}x + \gamma_{q,ij} \text{ for some } i \to j.$$  

*The SOCP-inexact power injection region is:*

$$\tilde{\mathcal{U}} = \{u \in \mathcal{U}' \mid u \text{ is SOCP-inexact}\}.$$
Step 4: Removing **SOCP-inexact** injections from $U'$

**Definition 2.** A power injection $u \in U'$ is **SOCP-inexact**, if every optimal solution of $FP'(u)$ satisfies:

$$\|y_{ij}\|_2 < c_{q,ij}x + \gamma_{q,ij} \quad \text{for some } i \rightarrow j.$$  

The **SOCP-inexact power injection region** is:

$$\tilde{U} = \{ u \in U' \mid u \text{ is SOCP-inexact} \}.$$

An approximate definition from Dual SOCP:

$$\tilde{U}_d := \{ u \in U' \mid \text{Every optimal solution of } DP'(u) \text{ satisfies } \lambda_{q,ij} = 0 \text{ for some } i \rightarrow j \}.$$  

Due to complementary slackness, $\tilde{U} \subseteq \tilde{U}_d$.  

Focus on $\tilde{U}_d$ as a good approximation of $\tilde{U}$.  

Step 4: Removing **SOCP-inexact** injections from $\mathcal{U}'$

---

**Algorithm 2:** Approximate $\tilde{U}_d$ (or SOCP-inexact $\hat{U}$)

1. **Initialization:** $\tilde{U}_p$ = $\mathcal{U}_p$ returned by Algorithm 1.
   
   Given $\delta \in \mathbb{R}_+^N$, $\eta, \eta' \in \mathbb{R}_+$; $\mathcal{V}_{safe} = \emptyset$; $c = 0$;

2. update vertices $\mathcal{V}(\tilde{U}_p)$. Let $d_{p_{max}}' = -\eta$;

3. for $u \in \mathcal{V}(\tilde{U}_p)$ and $u \notin \mathcal{V}_{safe}$ do

   4. solve DP''($u, \delta$) to obtain an optimal solution $(\mu^*, \lambda^*)$ and maximum objective value $d_{p_{max}}''(u, \delta)$;

   5. if $d_{p_{max}}''(u, \delta) > d_{p_{max}}''$ then

      6. $d_{p_{max}}'' \leftarrow d_{p_{max}}''(u, \delta)$;

      7. $(\mu_{max}, \lambda_{max}) \leftarrow (\mu^*, \lambda^*)$;

   8. else if $d_{p_{max}}''(u, \delta) \leq -\eta$ then

      9. $\mathcal{V}_{safe} = \mathcal{V}_{safe} \cup \{u\}$;

   10. if $d_{p_{max}}'' = -\eta$ or $c = C_{max}$ then

      11. return $\tilde{U}_p$;

   12. else

      13. add to $\tilde{U}_p$ a cutting plane:

         $\mu_{f_{max}} B_f u + \gamma_f + \lambda_{f_{max}} q_s \leq 0$

         $\mu_{y_{max}} b_y + \lambda_{q_{max}} q - \eta'$;

      14. $c \leftarrow c + 1$;

      15. go back to Line 2;

end

Tighten the dual feasible set to exclude $\lambda_q = 0$:

$$\lambda_q \geq \delta$$

For $u \in \tilde{U}_d$, the **tightened** Dual SOCP $DP''(u, \delta)$ should attain max. value **strictly** lower than 0.

In Alg. 2, this requirement is represented by

$$d_{p_{max}}''(u, \delta) \leq -\eta < 0$$
Step 4: Removing **SOCP-inexact** injections from $U'$

Alg. 2 returns a *convex polyhedral* approximation of $\tilde{U}_d$ (or $\tilde{U}$);

However, $\tilde{U}$ is generally nonconvex;

Moreover, output of Alg. 2 is sensitive to $(\delta, \eta)$.

**Proposed heuristic:**

- Run Alg. 2 multiple times with different $(\delta, \eta)$
- In each run, $\delta$ is a nonnegative perturbation to the dual optimal at one vertex of $U'_\text{poly}$ (from Alg. 1)
- The union of multiple Alg. 2 outcomes serves as an approximation of $\tilde{U}$
Preliminary numerical results

Alg. 1 converges in 26 iterations (263 seconds) to the SOCP-relaxed feasible region $\mathcal{U}'$

Max. dual obj. value over all the vertices
Preliminary numerical results

Left: Feasible region $U$ and its SOCP relaxation $U' = U \cup \tilde{U}$ found by checking sample points (close to actual cases)

Right: Alg. 1 output removing multiple Alg. 2 outputs.
Summary

A closed-form polyhedral approximation of feasible power injection regions in radial AC networks

- Model: nonlinear dist-flow
- Feasibility problem $\rightarrow$ SOCP relaxation $\rightarrow$ Dual SOCP $\rightarrow$ relaxed feasible region (Alg. 1)
- Heuristic to remove SOCP-inexact power injections (Alg. 2)
- Preliminary numerical results: simple form, moderate computation, relatively accurate

Limitations and future work: A better-justified design (rather than empirical heuristic) to remove SOCP-inexact power injections