New Formulation and Valid Inequalities for the Optimal Power Flow Problem

Burak Kocuk, Santanu S. Dey, X. Andy Sun
Georgia Institute of Technology

1 The Optimal Power Flow Problem: Introduction

The Optimal Power Flow Problem: High level

Optimal Power Flow (OPF) is a fundamental problem in power systems operations.

The Optimal Power Flow Problem: High level

Optimal Power Flow (OPF) is a fundamental problem in power systems operations.

The Set-up:

1. We have an infrastructure of generators and demand nodes and interconnecting lines.
2. The production cost is different for different generators.
3. Goal: Meet demands for power at minimum cost of production.

The Optimal Power Flow Problem: High level

Optimal Power Flow (OPF) is a fundamental problem in power systems operations.

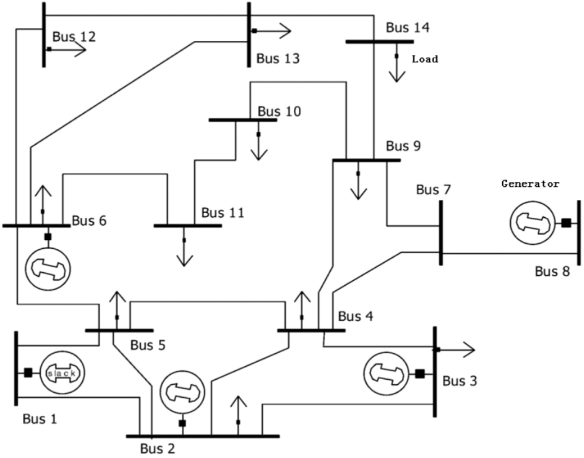
The Set-up:

1. We have an infrastructure of generators and demand nodes and interconnecting lines.
2. The production cost is different for different generators.
3. Goal: Meet demands for power at minimum cost of production.

Variables and constraints:

1. This is achieved by determining values of voltages and phase angles at each node, which in effect control the generation and flow of power.
2. The power flow satisfies physical constraints like Ohm's law and Kirchoff's Current law.

One-Line Diagram



Graph: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

Demand: (p_i^d, q_i^d)

Active: $[p_i^{\min}, p_i^{\max}]$

Reactive: $[q_i^{\min}, q_i^{\max}]$

Voltage : $[V_i^{\min}, V_i^{\max}]$

Admittance: (G_{ij}, B_{ij})

Thermal line limits

Polar formulation

Variables:

1. Active power at generator i: p_i^g.
2. Reactive power at generator i: q_i^g.
3. Voltage magnitude at node i: v_i.
4. Phase angle at node i: θ_i.

Polar formulation

Using Ohm's Law and Kirchoff's current law (at each node):

$$p_i^g - p_i^d = G_{ii} v_i^2 + \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

$$q_i^g - q_i^d = -B_{ii} v_i^2 - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

Polar formulation

Using Ohm's Law and Kirchoff's current law (at each node):

$$p_i^g - p_i^d = G_{ii} v_i^2 + \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

$$q_i^g - q_i^d = -B_{ii} v_i^2 - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

Bounds (at each node):

$$\begin{aligned} V_i^{\text{min}} \le & \quad v_i \quad \le V_i^{\text{max}} \\ p_i^{\text{min}} \le & \quad p_i^g \quad \le p_i^{\text{max}} \\ q_i^{\text{min}} \le & \quad q_i^g \quad \le q_i^{\text{max}}. \end{aligned}$$

Polar formulation

Using Ohm's Law and Kirchoff's current law (at each node):

$$p_i^g - p_i^d = G_{ii} v_i^2 + \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

$$q_i^g - q_i^d = -B_{ii} v_i^2 - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

Bounds (at each node):

$$\begin{aligned} V_i^{\min} \leq \ & v_i \leq V_i^{\max} \\ p_i^{\min} \leq \ & p_i^g \leq p_i^{\max} \\ q_i^{\min} \leq \ & q_i^g \leq q_i^{\max}. \end{aligned}$$

Objective function:

$$\min \sum_{i \in \mathcal{N}} f_i(p_i^g)$$

Polar formulation

Using Ohm's Law and Kirchoff's current law (at each node):

$$p_i^g - p_i^d = G_{ii} v_i^2 + \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

$$q_i^g - q_i^d = -B_{ii} v_i^2 - \sum_{j \in \delta(i)} B_{ij} \left(v_i v_j cos(\theta_i - \theta_j) \right) - \sum_{j \in \delta(i)} G_{ij} \left(v_i v_j sin(\theta_i - \theta_j) \right)$$

Bounds (at each node):

$$\begin{aligned} V_i^{\min} \leq\ & v_i \leq V_i^{\max} \\ p_i^{\min} \leq\ & p_i^g \leq p_i^{\max} \\ q_i^{\min} \leq\ & q_i^g \leq q_i^{\max}. \end{aligned}$$

Objective function:

$$\min \sum_{i \in \mathcal{N}} f_i(p_i^g)$$

Many global solver do not deal with trigonometric functions.

Rectangular formulation

Variables:

1. Active power at generator i: p_i^g.
2. Reactive power at generator i: q_i^g.

Rectangular formulation

Variables:

1. Active power at generator i: p_i^g.
2. Reactive power at generator i: q_i^g.
3. Voltage magnitude at node i $\times$ cosine (phase angle at i): e_i.
4. Voltage magnitude at node i $\times$ sine (phase angle at i): f_i.

Rectangular formulation

Variables:

1. Active power at generator i: p_i^g.
2. Reactive power at generator i: q_i^g.
3. Voltage magnitude at node $i \times$ cosine (phase angle at i): e_i.
4. Voltage magnitude at node $i \times$ sine (phase angle at i): f_i.

$$
\begin{aligned}
cos(\theta_i - \theta_j) &= cos(\theta_i)cos(\theta_j) + sin(\theta_i)sin(\theta_j) \\
sin(\theta_i - \theta_j) &= sin(\theta_i)cos(\theta_j) - sin(\theta_j)cos(\theta_i)
\end{aligned}
$$

Rectangular formulation

Using Ohm's Law and Kirchoff's current law (at each node):

$$
\begin{aligned}
p_i^g - p_i^d &= G_{ii}(e_i^2 + f_i^2) + \sum_{j \in \delta(i)} G_{ij}\,(e_i e_j + f_i f_j) - \sum_{j \in \delta(i)} B_{ij}\,(e_i f_j - f_i e_j) \\
q_i^g - q_i^d &= -B_{ii}(e_i^2 + f_i^2) - \sum_{j \in \delta(i)} B_{ij}\,(e_i e_j + f_i f_j) - \sum_{j \in \delta(i)} G_{ij}\,(e_i f_j - f_i e_j)
\end{aligned}
$$

Bounds (at each node):

$$
\begin{aligned}
(V_i^{\min})^2 \leq \; & e_i^2 + f_i^2 \leq (V_i^{\max})^2 \\
p_i^{\min} \leq \; & p_i^g \leq p_i^{\max} \\
q_i^{\min} \leq \; & q_i^g \leq q_i^{\max}.
\end{aligned}
$$

Objective function:

$$
\min \sum_{i \in \mathcal{N}} f_i(p_i^g)
$$

Literature

We can categorize the previous work into three classes:

- **Local optimal solutions** based on interior point solvers like MATPOWER.

Literature

We can categorize the previous work into three classes:

- **Local optimal solutions** based on interior point solvers like MATPOWER.
- **Convex relaxations using semidefinite programming (SDP) and Lasserre relaxations**. (Lavaei and Low, 2012; Madani et. al., 2013; Zhang and Tse, 2012; Lavaei et. al., 2014, Molzahn et al. 2013, Molzahn and Hiskens, 2014).
- **Second order cone program (SOCP) relaxation of polar formulation** (Jabr 2006, Hijazi et al., 2014)
- **Approximation algorithms with guaranteed bounds for the AC-OPF problem on graphs with bounded tree-width** (Bienstock, Munoz, 2015).

Literature

We can categorize the previous work into three classes:

- **Local optimal solutions** based on interior point solvers like MATPOWER.
- **Convex relaxations using semidefinite programming (SDP) and Lasserre relaxations**. (Lavaei and Low, 2012; Madani et. al., 2013; Zhang and Tse, 2012; Lavaei et. al., 2014, Molzahn et al. 2013, Molzahn and Hiskens, 2014).
- **Second order cone program (SOCP) relaxation of polar formulation** (Jabr 2006, Hijazi et al., 2014)
- **Approximation algorithms with guaranteed bounds for the AC-OPF problem on graphs with bounded tree-width** (Bienstock, Munoz, 2015).
- **Global optimal solutions** based on branch-and-bound (Phan, 2012)

Our Goal

1. Solve LPs/SOCPs of reasonable size to produce comparable bounds to the SDPs.
2. Generate valid linear inequalities that can be added to a solver such as BARON to improve the speed of finding globally optimal solutions.

1.1 The Optimal Power Flow Problem: An Alternative formulation

Rectangular formulation

$$
\begin{aligned}
p_i^g - p_i^d &= G_{ii}(e_i^2 + f_i^2) + \sum_{j \in \delta(i)} G_{ij}\,(e_i e_j + f_i f_j) - \sum_{j \in \delta(i)} B_{ij}\,(e_i f_j - f_i e_j) \\
q_i^g - q_i^d &= -B_{ii}(e_i^2 + f_i^2) - \sum_{j \in \delta(i)} B_{ij}\,(e_i e_j + f_i f_j) - \sum_{j \in \delta(i)} G_{ij}\,(e_i f_j - f_i e_j) \\
(V_i^{\min})^2 \leq & \quad e_i^2 + f_i^2 \quad \leq (V_i^{\max})^2 \\
p_i^{\min} \leq & \quad p_i^g \quad \leq p_i^{\max} \\
q_i^{\min} \leq & \quad q_i^g \quad \leq q_i^{\max}.
\end{aligned}
$$

Rectangular formulation: Identifying non-convexities

$$
\begin{aligned}
p_i^g - p_i^d &= G_{ii}(\mathbf{e_i^2} + \mathbf{f_i^2}) + \sum_{j \in \delta(i)} G_{ij}\left(\mathbf{e_i e_j} + \mathbf{f_i f_j}\right) - \sum_{j \in \delta(i)} B_{ij}\left(\mathbf{e_i f_j} - \mathbf{f_i e_j}\right) \\
q_i^g - q_i^d &= -B_{ii}(\mathbf{e_i^2} + \mathbf{f_i^2}) - \sum_{j \in \delta(i)} B_{ij}\left(\mathbf{e_i e_j} + \mathbf{f_i f_j}\right) - \sum_{j \in \delta(i)} G_{ij}\left(\mathbf{e_i f_j} - \mathbf{f_i e_j}\right) \\
(V_i^{\min})^2 \leq \ & \mathbf{e_i^2} + \mathbf{f_i^2} \leq (V_i^{\max})^2 \\
p_i^{\min} \leq \ & p_i^g \leq p_i^{\max} \\
q_i^{\min} \leq \ & q_i^g \leq q_i^{\max}.
\end{aligned}
$$

Rectangular formulation: Identifying non-convexities

$$p_i^g - p_i^d = G_{ii}(\mathbf{e_i^2} + \mathbf{f_i^2}) + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{e_i e_j} + \mathbf{f_i f_j}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{e_i f_j} - \mathbf{f_i e_j}\right)$$

$$q_i^g - q_i^d = -B_{ii}(\mathbf{e_i^2} + \mathbf{f_i^2}) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{e_i e_j} + \mathbf{f_i f_j}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{e_i f_j} - \mathbf{f_i e_j}\right)$$

$$(V_i^{\min})^2 \le \mathbf{e_i^2} + \mathbf{f_i^2} \le (V_i^{\max})^2$$

$$p_i^{\min} \le p_i^g \le p_i^{\max}$$

$$q_i^{\min} \le q_i^g \le q_i^{\max}.$$

$$\mathbf{e_i^2} + \mathbf{f_i^2} =: \mathbf{c_{ii}}$$

$$\mathbf{e_i e_j} + \mathbf{f_i f_j} =: \mathbf{c_{ij}}$$

$$\mathbf{e_i f_j} - \mathbf{f_i e_j} =: \mathbf{s_{ij}}.$$

Rectangular formulation: Identifying non-convexities

$$\left.\begin{array}{rcll} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\left.\begin{array}{rcl} \mathbf{e_i^2} + \mathbf{f_i^2} & = & \mathbf{c_{ii}} \\ \mathbf{e_i}\mathbf{e_j} + \mathbf{f_i}\mathbf{f_j} & = & \mathbf{c_{ij}} \\ \mathbf{e_i}\mathbf{f_j} - \mathbf{f_i}\mathbf{e_j} & = & \mathbf{s_{ij}}. \end{array}\right\} \text{non-convex quadratic.}$$

Implied equations in the *c*, *s* space

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\left.\begin{array}{rcl} \mathbf{e_i^2} + \mathbf{f_i^2} & = & \mathbf{c_{ii}} \\ \mathbf{e_i e_j} + \mathbf{f_i f_j} & = & \mathbf{c_{ij}} \\ \mathbf{e_i f_j} - \mathbf{f_i e_j} & = & \mathbf{s_{ij}}. \end{array}\right\} \text{non-convex quadratic.}$$

New implied inequalities:

$$\boxed{\begin{array}{rcl} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} & = & \mathbf{c_{ii} c_{jj}} \\ \mathbf{c_{ij}} & = & \mathbf{c_{ji}} \\ \mathbf{s_{ij}} & = & -\mathbf{s_{ji}}. \end{array}}$$

Getting rid of the “e, f" variables

$$
\left.\begin{array}{rcl}
p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\
q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\
(V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\
p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\
q_i^{\min} \leq & q_i^g & \leq q_i^{\max}.
\end{array}\right\} \text{linear}
$$

$$
\begin{aligned}
\mathbf{c_{ij}}^2 + \mathbf{s_{ij}}^2 &= \mathbf{c_{ii}}\mathbf{c_{jj}} \\
\mathbf{c_{ij}} &= \mathbf{c_{ji}} \\
\mathbf{s_{ij}} &= -\mathbf{s_{ji}}.
\end{aligned}
$$

Getting rid of the “e, f" variables

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{aligned} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} &= \mathbf{c_{ii}}\mathbf{c_{jj}} \\ \mathbf{c_{ij}} &= \mathbf{c_{ji}} \\ \mathbf{s_{ij}} &= -\mathbf{s_{ji}}. \end{aligned}$$

Is this a correct formulation?

Getting rid of the “e, f" variables

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{array}{rcl} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} & = & \mathbf{c_{ii}}\mathbf{c_{jj}} \\ \mathbf{c_{ij}} & = & \mathbf{c_{ji}} \\ \mathbf{s_{ij}} & = & -\mathbf{s_{ji}}. \end{array}$$

Above is valid formulation, if $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is tree. In general we also need:

Getting rid of the “e, f" variables

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{aligned} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} &= \mathbf{c_{ii}}\mathbf{c_{jj}} \\ \mathbf{c_{ij}} &= \mathbf{c_{ji}} \\ \mathbf{s_{ij}} &= -\mathbf{s_{ji}}. \end{aligned}$$

Above is valid formulation, if $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is tree. In general we also need:

$$\left.\sum_{(i,j)\in \text{cycle}} \text{atan2}\left(\frac{\mathbf{s_{ij}}}{\sqrt{\mathbf{c_{ii}}\mathbf{c_{jj}}}}, \frac{\mathbf{c_{ij}}}{\sqrt{\mathbf{c_{ii}}\mathbf{c_{jj}}}}\right) = 0 \ \right\} \text{For all cycles.}$$

Getting rid of the “e, f" variables

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \le & \mathbf{c_{ii}} & \le (V_i^{\max})^2 \\ p_i^{\min} \le & p_i^g & \le p_i^{\max} \\ q_i^{\min} \le & q_i^g & \le q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{array}{rcl} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} & = & \mathbf{c_{ii}}\mathbf{c_{jj}} \\ \mathbf{c_{ij}} & = & \mathbf{c_{ji}} \\ \mathbf{s_{ij}} & = & -\mathbf{s_{ij}}. \end{array}$$

$$\sum_{(i,j)\in\text{cycle}} \text{atan2}\left(\frac{\mathbf{s_{ij}}}{\sqrt{\mathbf{c_{ii}}\mathbf{c_{jj}}}}, \frac{\mathbf{c_{ij}}}{\sqrt{\mathbf{c_{ii}}\mathbf{c_{jj}}}}\right) = 0 \;\Big\}\; \text{For all cycles in a cycle basis.}$$

This formulation is described in Jabr(2006).

2 Which formulation to use: comparing quality of relaxations

The different formulations

1. Polar formulation [Variables: p, v, θ]
2. Rectangular formulation [Variables: p, e, f]
3. Alternative formulation [Variables: p, c, s]

The different formulations

2. Rectangular formulation [Variables: p, e, f]
3. Alternative formulation [Variables: p, c, s]

If we relax the "atan2" constraints in Alternative formulation, both the above are non-convex quadratic formulation

Standard reformulation

$$\begin{aligned} \min \quad & x^T C x + c^T x \\ \text{s.t.} \quad & x^T A_k x + a_k^T x \leq b_k \ \forall k \in \{1, \ldots, m\} \\ & l \leq x \leq u. \end{aligned}$$

Standard reformulation

$$\begin{aligned} \min \quad & x^T C x + c^T x \\ \text{s.t.} \quad & x^T A_k x + a_k^T x \leq b_k \ \forall k \in \{1, \ldots, m\} \\ & l \leq x \leq u. \end{aligned}$$

$$\begin{aligned} \min \quad & \langle C, X \rangle + c^T x \\ \text{s.t.} \quad & \langle A_k, X \rangle + a_k^T x \leq b_k \ \forall k \in \{1, \ldots, m\} \\ & X = x x^T \\ & l \leq x \leq u. \end{aligned}$$

McCormick relaxation: A Linear programming relaxation

$$\begin{array}{ll} \min & \langle C, X\rangle + c^T x \\ \text{s.t.} & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1,\ldots,m\} \\ & X = xx^T \\ & l \leq x \leq u. \end{array}$$

$$\begin{array}{ll} \min & \langle C, X\rangle + c^T x \\ \text{s.t.} & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1,\ldots,m\} \\ & X_{ij} - l_i x_j - l_j x_i + l_i l_j \geq 0 \\ & u_j x_i - X_{ij} - l_i u_j + l_i x_j \geq 0 \\ & u_i x_j - u_i l_j - X_{ij} + l_j x_i \geq 0 \\ & u_i x_j - u_i x_j - u_j x_i + X_{ij} \geq 0 \\ & l \leq x \leq u. \end{array}$$

Standard semi-definite programming relaxation

$$\begin{aligned} \min \quad & \langle C, X\rangle + c^T x \\ \text{s.t.} \quad & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1, \ldots, m\} \\ & X = xx^T \\ & l \leq x \leq u. \end{aligned}$$

$$\begin{aligned} \min \quad & \langle C, X\rangle + c^T x \\ \text{s.t.} \quad & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1, \ldots, m\} \\ & \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0 \\ & l \leq x \leq u. \end{aligned}$$

Standard second order conic relaxation

$$\begin{aligned} \min \quad & \langle C, X\rangle + c^T x \\ \text{s.t.} \quad & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1, \dots, m\} \\ & X = xx^T \\ & l \leq x \leq u. \end{aligned}$$

$$\begin{aligned} \min \quad & \langle C, X\rangle + c^T x \\ \text{s.t.} \quad & \langle A_k, X\rangle + a_k^T x \leq b_k \ \forall k \in \{1, \dots, m\} \\ & \text{All } 2 \times 2 \text{ principal submatrix of } \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0 \\ & l \leq x \leq u. \end{aligned}$$

McCormick Relaxation of Rectangular and Alternative formulation

Theorem

Let $\mathcal{R}^M$ and $\mathcal{A}^M$ be the projection on the space of (p, q) variables of the McCormick relaxation of the Rectangular and Alternative formulation (where we drop the atan2 constraints). Then

$$\mathcal{R}^M \supseteq \mathcal{A}^M.$$

Moreover, there exist instances where the above inclusion is proper.

A different SOCP relaxation of Alternative formulation

$$\left.\begin{array}{rcll} p_i^g - p_i^d & = & G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = & -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 & \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} & \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} & \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} = \mathbf{c_{ii}}\mathbf{c_{jj}} \qquad <----\text{relax this constraint}$$

$$\mathbf{c_{ij}} = \mathbf{c_{ji}}$$

$$\mathbf{s_{ij}} = -\mathbf{s_{ji}}.$$

A different SOCP relaxation of Alternative formulation

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(\mathbf{c_{ii}}) & + \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{s_{ij}}\right) \\ q_i^g - q_i^d & = -B_{ii}(\mathbf{c_{ii}}) & - \sum_{j\in\delta(i)} B_{ij}\left(\mathbf{c_{ij}}\right) - \sum_{j\in\delta(i)} G_{ij}\left(\mathbf{s_{ij}}\right) \\ (V_i^{\min})^2 \leq & \mathbf{c_{ii}} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{array}{rcl} \mathbf{c_{ij}^2} + \mathbf{s_{ij}^2} & \leq & \mathbf{c_{ii}}\mathbf{c_{jj}} \\ \mathbf{c_{ij}} & = & \mathbf{c_{ji}} \\ \mathbf{s_{ij}} & = & -\mathbf{s_{ij}}. \end{array}$$

Comparison of SOCPs for trees

Theorem

Let $\mathcal{A}^{M}$, $\mathcal{A}^{SOCP*}$ be the projection on the space of (p, q) of the McCormick and SOCP relaxation of the Alternative formulation for a OPF instance on a tree. Let $\mathcal{R}^{SDP}$ be the projection on the space of (p, q) of the SDP relaxation of the rectangular formulation of the same OPF instance. Then

$$\mathcal{A}^{M} \supseteq \mathcal{A}^{SOCP*} = \mathcal{R}^{SDP}.$$

Moreover, there exist instances where the first inclusion is proper.

Some empirical evidence

number of nodes	$\mathcal{R}^M$	$\mathcal{A}^M$	$\mathcal{A}^{SOCP}$
9	89.46	69.13	52.69
9	34.13	22.56	15.63
9	60.14	45.58	35.42
14	86.32	38.86	0.59
14	83.92	44.29	0.60
30	100.00	100.00	2.11
30	100.00	100.00	15.84
30	9.13	9.13	1.82
30	5.79	5.79	2.30
30	16.56	16.56	1.91
39	44.90	17.57	0.37
39	23.78	6.35	0.16
Average:	57.3	42.7	11.8

Table: Percent optimality gap for McCormick vs. SOCP with respect to global optimal solution found by BARON for tree instances.

All comparisons for general graphs

$$\begin{array}{ccccccccc} \mathcal{R}^{SDP} & \subseteq & \mathcal{R}^{SOCP} & = & \mathcal{A}^{SOCP*} & \subseteq & \mathcal{A}^{SDP} & \subseteq & \mathcal{A}^{SOCP} \\ & & & & \cap\!\mid & & & & \\ & & \mathcal{R}^{M} & \supseteq & \mathcal{A}^{M} & & & & \end{array}$$

3

The Optimal Power Flow Problem on Trees

3.1
Improving the McCormick Formulation (and still keeping the relaxation an LP)

The main idea is:

The main idea is:

Improve Bounds!

Bound tightening

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$ and $(V_i^{\min})^2 \leq c_{ii} \leq (V_i^{\max})^2$.

- **Implied bounds** on c_{ij} and s_{ij} are

$$-V_i^{\max}V_j^{\max} \leq c_{ij}, s_{ij} \leq V_i^{\max}V_j^{\max} \quad (i,j) \in \mathcal{L}$$

Bound tightening

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$ and $(V_i^{\max})^2 \leq c_{ii} \leq (V_i^{\max})^2$.

- **Implied bounds** on c_{ij} and s_{ij} are

$$-V_i^{\max} V_j^{\max} \leq c_{ij}, s_{ij} \leq V_i^{\max} V_j^{\max} \quad (i,j) \in \mathcal{L}$$

- However, these bounds are very **loose**

Bound tightening

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$ and $(V_i^{\min})^2 \leq c_{ii} \leq (V_i^{\max})^2$.

- **Implied bounds** on c_{ij} and s_{ij} are

$$-V_i^{\max}V_j^{\max} \leq c_{ij}, s_{ij} \leq V_i^{\max}V_j^{\max} \quad (i,j) \in \mathcal{L}$$

- However, these bounds are very **loose**
- One way to obtain **better variable bounds** is to optimize c_{ij} and s_{ij} over the feasible region of **SOCP relaxation**.

Valid inequalities

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$ and $(V_i^{\min})^2 \leq c_{ii} \leq (V_i^{\max})^2$ again.

Valid inequalities

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$ and $(V_i^{\min})^2 \leq c_{ii} \leq (V_i^{\max})^2$ again. Define

$$\mathcal{B}_{ij} = [\underline{c}_{ij}, \overline{c}_{ij}] \times [\underline{s}_{ij}, \overline{s}_{ij}]$$

$$\mathcal{R}_{ij} = \{(c_{ij}, s_{ij}) : (V_i^{\min} V_j^{\min})^2 \leq c_{ij}^2 + s_{ij}^2 \leq (V_i^{\max} V_j^{\max})^2\}$$

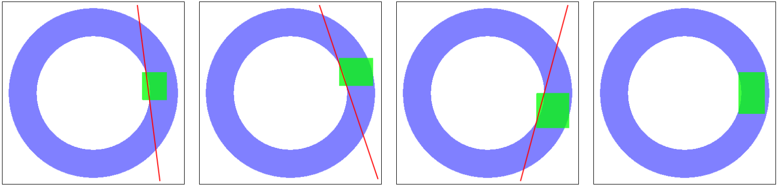
Valid inequalities

Consider $c_{ij}^2 + s_{ij}^2 = c_{ii} c_{jj}$ and $(V_i^{\text{min}})^2 \leq c_{ii} \leq (V_i^{\text{max}})^2$ again. Define

$$\mathcal{B}_{ij} = [\underline{c}_{ij}, \overline{c}_{ij}] \times [\underline{s}_{ij}, \overline{s}_{ij}]$$

$$\mathcal{R}_{ij} = \{(c_{ij}, s_{ij}) : (V_i^{\text{min}} V_j^{\text{min}})^2 \leq c_{ij}^2 + s_{ij}^2 \leq (V_i^{\text{max}} V_j^{\text{max}})^2\}$$

Figure: Positioning of $\mathcal{B}_{ij}$ and $\mathcal{R}_{ij}$.



(a) Case 1 (b) Case 2 (c) Case 3 (d) Case 4

Results

PT: preprocessing time (s)
BT: BARON solution time (s)
TT: total time (s)

	BARON	with bounds		bounds and cuts	
number of nodes	BT	PT	TT	TT	#cuts
9	1.17				
9	1.11				
9	1.36				
14	35.32				
14	0.79				
30	8347.79				
30	2494.31				
30	2.52				
30	8.50				
30	5.16				
39	110.59				
39	1566.88				

Results

PT: preprocessing time (s)
BT: BARON solution time (s)
TT: total time (s)

	BARON	with bounds		bounds and cuts	
number of nodes	BT	PT	TT	TT	#cuts
9	1.17	4.34	5.42		
9	1.11	4.12	4.98		
9	1.36	4.36	5.58		
14	35.32	7.11	37.56		
14	0.79	6.95	7.79		
30	8347.79	16.91	917.41		
30	2494.31	16.89	266.37		
30	2.52	17.21	19.12		
30	8.50	17.60	19.99		
30	5.16	16.53	18.93		
39	110.59	28.07	54.10		
39	1566.88	26.94	99.74		

Results

PT: preprocessing time (s)
BT: BARON solution time (s)
TT: total time (s)

	BARON	with bounds		bounds and cuts	
number of nodes	BT	PT	TT	TT	#cuts
9	1.17	4.34	5.42	5.42	6
9	1.11	4.12	4.98	5.42	6
9	1.36	4.36	5.58	5.46	6
14	35.32	7.11	37.56	48.88	7
14	0.79	6.95	7.79	7.80	7
30	8347.79	16.91	917.41	17.63	14
30	2494.31	16.89	266.37	17.30	14
30	2.52	17.21	19.12	21.35	13
30	8.50	17.60	19.99	18.14	13
30	5.16	16.53	18.93	18.64	13
39	110.59	28.07	54.10	60.98	12
39	1566.88	26.94	99.74	70.79	12

Results

BARON Time for rectangular formulation >2hrs

	BARON	with bounds		bounds and cuts	
number of nodes	BT	PT	TT	TT	#cuts
9	1.17	4.34	5.42	5.42	6
9	1.11	4.12	4.98	5.42	6
9	1.36	4.36	5.58	5.46	6
14	35.32	7.11	37.56	48.88	7
14	0.79	6.95	7.79	7.80	7
30	8347.79	16.91	917.41	17.63	14
30	2494.31	16.89	266.37	17.30	14
30	2.52	17.21	19.12	21.35	13
30	8.50	17.60	19.99	18.14	13
30	5.16	16.53	18.93	18.64	13
39	110.59	28.07	54.10	60.98	12
39	1566.88	26.94	99.74	70.79	12

4 Solving OPF over general graphs

4.1
A new bilinear formulation in an “extended" cs-space

Revisiting the Alternative formulation

$$\left.\begin{array}{rcl} p_i^g - p_i^d &=& G_{ii}(c_{ii}) + \sum_{j\in\delta(i)} G_{ij}\left(c_{ij}\right) - \sum_{j\in\delta(i)} B_{ij}\left(s_{ij}\right) \\ q_i^g - q_i^d &=& -B_{ii}(c_{ii}) - \sum_{j\in\delta(i)} B_{ij}\left(c_{ij}\right) - \sum_{j\in\delta(i)} G_{ij}\left(s_{ij}\right) \\ (V_i^{\min})^2 \leq & c_{ii} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$\begin{array}{rcl} c_{ij}^2 + s_{ij}^2 &\leq& c_{ii}c_{jj} \\ c_{ij} &=& c_{ji} \\ s_{ij} &=& -s_{ij}. \end{array}$$

$$\sum_{(i,j)\in \text{cycle}} \text{atan2}\left(\frac{s_{ij}}{\sqrt{c_{ii}c_{jj}}}, \frac{c_{ij}}{\sqrt{c_{ii}c_{jj}}}\right) = 0 \;\Big\}$$ For all cycles in a cycle basis.

The cycle constraints

Definition (cs-monomial and signs)

Let i_1, i_2, ..., $i_k (\equiv i_1)$ form a cycle. We construct a degree $k-1$ monomial of the form

$$\prod_{j=1}^{k-1} x_{i_j, i_{j+1}},$$

where for each $j \in \{1, \dots, k-1\}$ we either have $x = c$ or $x = s$.

The cycle constraints

Definition (cs-monomial and signs)

Let i_1, i_2, ..., $i_k(\equiv i_1)$ form a cycle. We construct a degree $k-1$ monomial of the form

$$\prod_{j=1}^{k-1} x_{i_j,i_{j+1}},$$

where for each $j \in \{1,\ldots,k-1\}$ we either have $x = c$ or $x = s$.

1. Odd cs-monomial: The number of s's in the cs-monomial is odd. The sign of an odd cs-monomial is defined as $-1^{\frac{\text{number of } s - 1}{2}}$.
2. Even cs-monomial: The number of s's in the cs-monomial is even. The sign of an even cs-monomial is defined as $-1^{\frac{\text{number of } s}{2}}$.

The cycle constraints, contd.

$$\left.\sum_{j=1}^{k} \text{atan2}\left(\frac{s_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}, \frac{c_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}\right) = 0 \quad \right\} \equiv \text{ sum of angle difference zero}$$

The cycle constraints, contd.

$$\left.\sum_{j=1}^{k} \text{atan2}\left(\frac{s_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}, \frac{c_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}\right) = 0 \right\} \equiv \text{ sum of angle difference zero}$$

is equivalent to:

$$\begin{aligned} \theta_{i_j} - \theta_{i_{j+1}} &= \text{atan2}\left(\frac{s_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}, \frac{c_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}\right) \\ sin\left(\sum_{j=1}^{k-2} (\theta_{i_j} - \theta_{i_{j+1}})\right) &= sin(\theta_1 - \theta_{k-1}) \\ cos\left(\sum_{j=1}^{k-2} (\theta_{i_j} - \theta_{i_{j+1}})\right) &= cos(\theta_1 - \theta_{k-1}). \end{aligned}$$

The cycle constraints, contd.

$$\left.\sum_{j=1}^{k} \text{atan2}\left(\frac{s_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}, \frac{c_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}\right) = 0 \;\right\} \equiv \text{ sum of angle difference zero}$$

is equivalent to:

$$\begin{aligned} \theta_{i_j} - \theta_{i_{j+1}} &= \text{atan2}\left(\frac{s_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}, \frac{c_{i_j i_{j+1}}}{\sqrt{c_{i_j i_j} c_{i_{j+1} i_{j+1}}}}\right) \\ sin\left(\sum_{j=1}^{k-2} (\theta_{i_j} - \theta_{i_{j+1}})\right) &= sin(\theta_1 - \theta_{k-1}) \\ cos\left(\sum_{j=1}^{k-2} (\theta_{i_j} - \theta_{i_{j+1}})\right) &= cos(\theta_1 - \theta_{k-1}). \end{aligned}$$

which is equivalent to:

$$\sum_{\alpha \text{ is an odd cs-monimial}} \text{sign}(\alpha)\alpha = s_{i_1, i_{k-1}} \prod_{j=2}^{k-2} c_{jj}$$

$$\sum_{\alpha \text{ is an even cs-monimial}} \text{sign}(\alpha)\alpha = c_{i_1, i_{k-1}} \prod_{j=2}^{k-2} c_{jj}$$

A new polynomial formulation for OPF

$$\left.\begin{array}{rcl} p_i^g - p_i^d & = G_{ii}(c_{ii}) & + \sum_{j\in\delta(i)} G_{ij}\left(c_{ij}\right) - \sum_{j\in\delta(i)} B_{ij}\left(s_{ij}\right) \\ q_i^g - q_i^d & = -B_{ii}(c_{ii}) & - \sum_{j\in\delta(i)} B_{ij}\left(c_{ij}\right) - \sum_{j\in\delta(i)} G_{ij}\left(s_{ij}\right) \\ (V_i^{\min})^2 \leq & c_{ii} & \leq (V_i^{\max})^2 \\ p_i^{\min} \leq & p_i^g & \leq p_i^{\max} \\ q_i^{\min} \leq & q_i^g & \leq q_i^{\max}. \end{array}\right\} \text{linear}$$

$$c_{ij}^2 + s_{ij}^2 = c_{ii}c_{jj}$$

$$c_{ij} = c_{ji}, \quad s_{ij} = -s_{ij}$$

For every cycle (in a cycle basis):

$$\sum_{\alpha \text{ is an odd cs-monimial}} \text{sign}(\alpha)\alpha = s_{i_1,i_{k-1}} \prod_{j=2}^{k-2} c_{jj}$$

$$\sum_{\alpha \text{ is an even cs-monimial}} \text{sign}(\alpha)\alpha = c_{i_1,i_{k-1}} \prod_{j=2}^{k-2} c_{jj}.$$

Observation

1. No inverse trigonometric functions, but

$$\text{degree of polynomial} = \text{size of cycle} - 1$$

Observation

1. No inverse trigonometric functions, but

$$\text{degree of polynomial} = \text{size of cycle} - 1$$

2. There are standard techniques for converting this into a bilinear program.

Observation

1. No inverse trigonometric functions, but

   $$\text{degree of polynomial} = \text{size of cycle} - 1$$

2. There are standard techniques for converting this into a bilinear program. This is how BARON solves this program.

Observation

1. No inverse trigonometric functions, but

$$\text{degree of polynomial} = \text{size of cycle} - 1$$

2. There are standard techniques for converting this into a bilinear program. This is how BARON solves this program.
3. We have tried a more “natural" way to convert this formulation into a bilinear program:

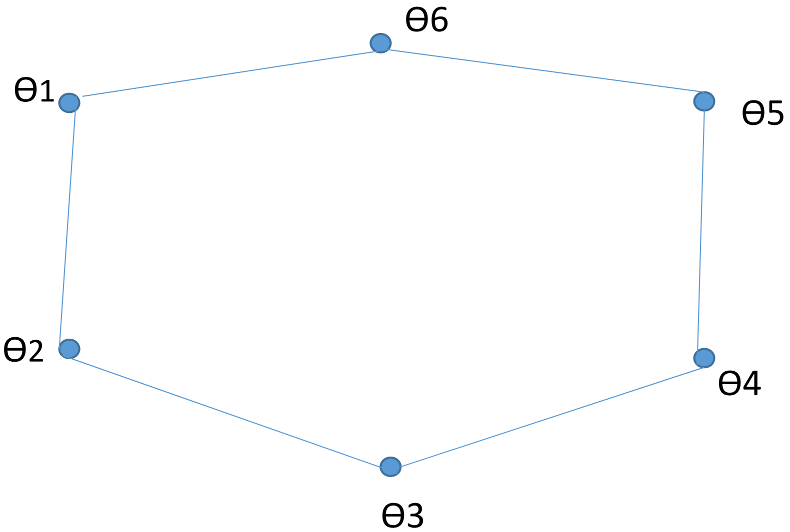
Observation

1. No inverse trigonometric functions, but

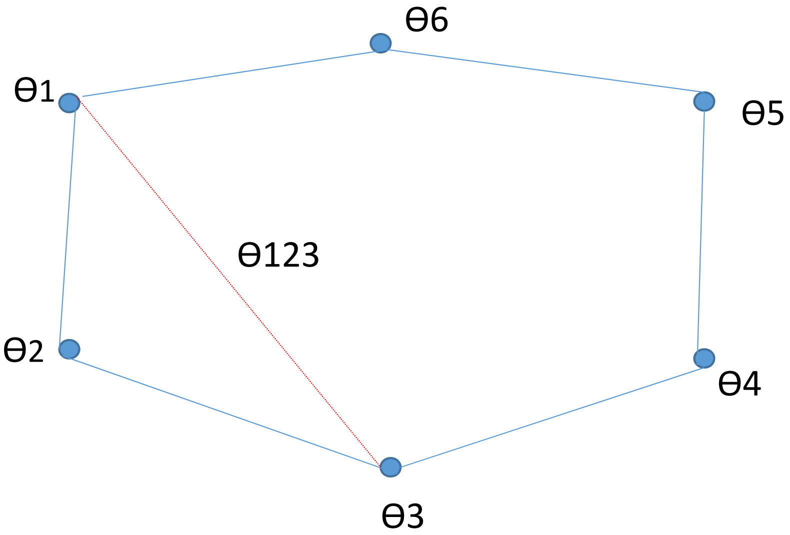
$$\text{degree of polynomial} = \text{size of cycle} - 1$$

2. There are standard techniques for converting this into a bilinear program. This is how BARON solves this program.
3. We have tried a more “natural" way to convert this formulation into a bilinear program: For 3 and 4 cycle also one can show that it is possible to write a bilinear formulation in the space of original variables.

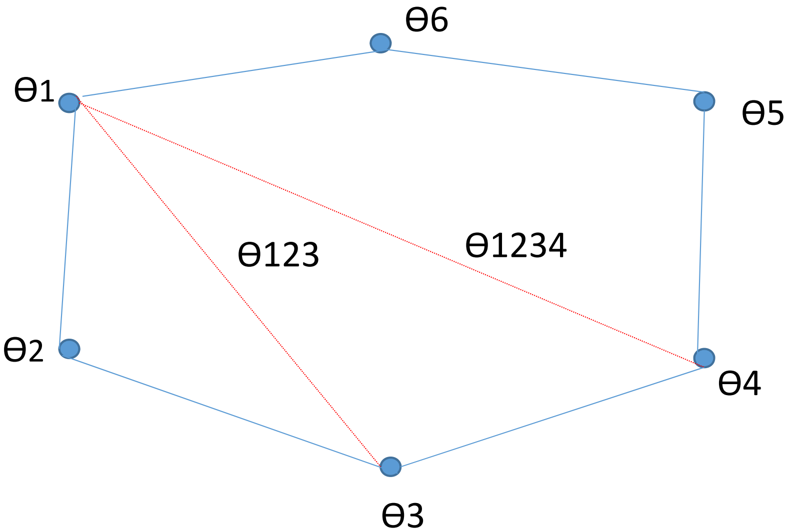
Artificial edges



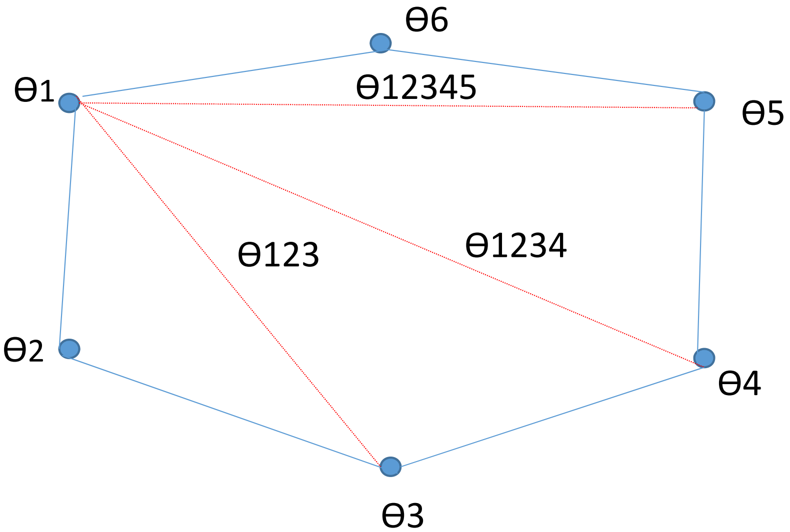
Artificial edges



Artificial edges



Artificial edges



3.2

Strengthening the $\mathcal{A}^{SOCP*}$ relaxation

Three main ideas

1. Use McCormick relaxation on the new bilinear equations.

Three main ideas

1. Use McCormick relaxation on the new bilinear equations.
2. Extended SOCP relaxation of the new bilinear equations.

Three main ideas

1. Use McCormick relaxation on the new bilinear equations.
2. Extended SOCP relaxation of the new bilinear equations.
3. Embracing the atan2 constrainsts: Re-introduce angle variables and devise linear outer-approximation of the constraints: (assuming phase differences is in the range $[-\frac{\pi}{2}, \ \frac{\pi}{2}]$)

$$\theta_i - \theta_j = \arctan\left(\frac{s_{ij}}{c_{ij}}\right)$$

Quality of McCormick envelops

$$
\begin{array}{ccccccccc}
\mathcal{R}^{SDP} & \subseteq & \mathcal{R}^{SOCP} & = & \mathcal{A}^{SOCP*} & \subseteq & \mathcal{A}^{SDP} & \subseteq & \mathcal{A}^{SOCP} \\
 & & & & \rotatebox{90}{$\subseteq$} & & & & \\
 & & \mathcal{R}^{M} & \supseteq & \mathcal{A}^{M} & & & &
\end{array}
$$

Quality of McCormick envelops

In particular, we have: $\mathcal{R}^{SDP} \subseteq \mathcal{A}^{SOCP*}$

Quality of McCormick envelops

In particular, we have: $\mathcal{R}^{SDP} \subseteq \mathcal{A}^{SOCP*}$

However: $\mathcal{R}^{SDP}$ incomparable with $\left(\mathcal{A}^{SOCP*} \cap \text{new McCormick constraints}\right)$

Some empirical evidence: a shooting experiment

- SOCP3: SOCP + McCormick inequalities from 3-decomposition.
- SOCP4: SOCP + McCormick inequalities from 4-decomposition.

Some empirical evidence: a shooting experiment

- SOCP3: SOCP + McCormick inequalities from 3-decomposition.
- SOCP4: SOCP + McCormick inequalities from 4-decomposition.
- Graph of instance: A 8-cycle.
- 100 randomly generated linear constraints.

$\geq$	SOCP3	SOCP4	SDP-Rectangular
SOCP3	-	81	58
SOCP4	5	-	11
SDP-Rectangular	36	59	-

Idea 2: Extended SOCP relaxation of the bilinear constraints

Theorem

Let $a, \underline{x}, \bar{x}, \underline{y}, \bar{y} \in \mathbb{R}^n$ *such that* $\underline{x} < \bar{x}$ *and* $\underline{y} < \bar{y}$. *Let*

$$S := \left\{ (x, y) \in \mathbb{R}^n \times \mathbb{R}^n \,\middle|\, \sum_{i=1}^{n} a_i x_i y_i = 0,\ \underline{x} \leq x \leq \bar{x},\ \underline{y} \leq y \leq \bar{y} \right\}.$$

Then $\mathrm{conv}(S)$ *is SOCP-representable.*

Idea 2: Extended SOCP relaxation of the bilinear constraints

Theorem

Let $a, \underline{x}, \bar{x}, \underline{y}, \bar{y} \in \mathbb{R}^n$ *such that* $\underline{x} < \bar{x}$ *and* $\underline{y} < \bar{y}$. *Let*

$$S := \left\{ (x, y) \in \mathbb{R}^n \times \mathbb{R}^n \,\middle|\, \sum_{i=1}^{n} a_i x_i y_i = 0,\ \underline{x} \leq x \leq \bar{x},\ \underline{y} \leq y \leq \bar{y} \right\}.$$

Then conv(S) *is SOCP-representable.*

- Approximately 4^n new variables.

Idea 2: Extended SOCP relaxation of the bilinear constraints

Theorem

Let $a, \underline{x}, \bar{x}, \underline{y}, \bar{y} \in \mathbb{R}^n$ *such that* $\underline{x} < \bar{x}$ *and* $\underline{y} < \bar{y}$. *Let*

$$S := \left\{ (x, y) \in \mathbb{R}^n \times \mathbb{R}^n \;\middle|\; \sum_{i=1}^{n} a_i x_i y_i = 0,\ \underline{x} \leq x \leq \bar{x},\ \underline{y} \leq y \leq \bar{y} \right\}.$$

Then conv(S) *is SOCP-representable.*

- Approximately 4^n new variables.

1. The 4-cycle (resp. 3-cycle) has two equations with $n = 4$ (resp. $n = 3$).

Idea 2: Extended SOCP relaxation of the bilinear constraints

Theorem

Let $a, \underline{x}, \bar{x}, \underline{y}, \bar{y} \in \mathbb{R}^n$ *such that* $\underline{x} < \bar{x}$ *and* $\underline{y} < \bar{y}$*. Let*

$$S := \left\{ (x, y) \in \mathbb{R}^n \times \mathbb{R}^n \;\middle|\; \sum_{i=1}^{n} a_i x_i y_i = 0,\ \underline{x} \leq x \leq \bar{x},\ \underline{y} \leq y \leq \bar{y} \right\}.$$

Then $\operatorname{conv}(S)$ *is SOCP-representable.*

- Approximately 4^n new variables.

1. The 4-cycle (resp. 3-cycle) has two equations with $n = 4$ (resp. $n = 3$).
2. We apply the extended formulation of each equation.
3. Trivially this approach produces tighter bounds than McCormicks.

Idea 3: “Embrace" the arc-tan constraints

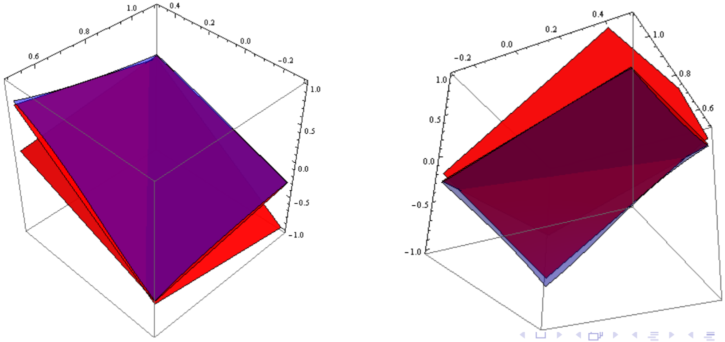
1. Introduce phase angle variables θ_i for each node i.

Idea 3: “Embrace" the arc-tan constraints

1. Introduce phase angle variables θ_i for each node i. Want to enforce $\theta_i - \theta_j = \arctan\left(\frac{s_{ij}}{c_{ij}}\right)$

Idea 3: “Embrace" the arc-tan constraints

1. Introduce phase angle variables θ_i for each node i. Want to enforce $\theta_i - \theta_j = \arctan\left(\frac{s_{ij}}{c_{ij}}\right)$
2. Outer approximation of the above set by 4 linear inequalities: Need to solve four simple global optimization problems to obtain these inequalities.



5
The business end: computational experiments over general graphs

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.
3. For each edge improve bounds on c_{ij}, s_{ij} variables by solving SOCPs. **[Parallelized]**

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.
3. For each edge improve bounds on c_{ij}, s_{ij} variables by solving SOCPs. **[Parallelized]**
4. For each edge add the arctan linearization to the $\mathcal{A}^{AOCP*}$ model. **[Parallelized]** [Optional]
5. Solve convex relaxation.

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.
3. For each edge improve bounds on c_{ij}, s_{ij} variables by solving SOCPs. **[Parallelized]**
4. For each edge add the arctan linearization to the $\mathcal{A}^{AOCP*}$ model. **[Parallelized]** [Optional]
5. Solve convex relaxation.
6. For i = 1 to num iterations

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.
3. For each edge improve bounds on c_{ij}, s_{ij} variables by solving SOCPs. **[Parallelized]**
4. For each edge add the arctan linearization to the $\mathcal{A}^{AOCP*}$ model. **[Parallelized]** [Optional]
5. Solve convex relaxation.
6. For i = 1 to num iterations
   6.1 Separate cutting planes from each cycle in cycle basis using any one (or both) of the following **[Parallelized]**:
      - McCormick of 3, 4-decomposition of the cycle.
      - Extended SOCP relaxation of cycle.

   6.2 Add cuts and resolve convex relaxation.

Algorithm

1. Input instance in the space of c, s and phase angle θ variables.
2. Compute a cycle basis.
3. For each edge improve bounds on c_{ij}, s_{ij} variables by solving SOCPs. **[Parallelized]**
4. For each edge add the arctan linearization to the $\mathcal{A}^{AOCP*}$ model. **[Parallelized]** [Optional]
5. Solve convex relaxation.
6. For i = 1 to num iterations
   6.1 Separate cutting planes from each cycle in cycle basis using any one (or both) of the following **[Parallelized]**:
      - McCormick of 3, 4-decomposition of the cycle.
      - Extended SOCP relaxation of cycle.

   6.2 Add cuts and resolve convex relaxation.
7. Use the final (infeasible) solution of convex relaxation as initial point of a interiror point solver to find a feasible point.

Solver, Hardware, Instances

1. Solver: MOSEK, IPOPT.
2. Hardware: 64-bit laptop with Intel Core i7 CPU with 2.00GHz processor and 8 GB RAM.
3. Instances: standard IEEE instances, standard instances where we randomly perturbed the demand $\pm 5\%$.

5.1
Quality of lower bound

instance	SOCP		SOCPA		SOCP34AM	
	% gap	time	% gap	time	% gap	time
case6ww	0.63	0.13	0.02	0.48	0.01	0.45
case9	0.00	0.04	0.00	0.19	0.00	0.20
case9Q	0.04	0.04	0.04	0.20	0.04	0.21
case14	0.08	0.05	0.08	0.44	0.06	0.48
caseieee30	0.04	0.07	0.04	0.83	0.04	0.88
case30	0.57	0.12	0.37	1.01	0.34	1.13
case30Q	2.48	0.11	2.35	1.07	2.32	1.16
case39	0.02	0.10	0.01	0.96	0.01	1.05
case57	0.06	0.11	0.06	1.50	0.06	1.55
case118	0.25	0.27	0.24	3.86	0.16	5.00
case300	0.15	0.62	0.12	8.04	0.11	10.83
case2383wp	1.05	21.39	0.89	104.71	0.88	145.29
case2736sp	0.51	17.90	0.39	127.61	0.38	184.09
case2737sop	0.36	14.27	0.31	116.89	0.31	143.58
case2746wop	0.44	16.02	0.35	129.14	0.35	223.26
case2746wp	0.45	20.86	0.33	130.59	0.23	201.06
case3012wp	0.79	19.65	0.70	154.72	0.70	195.56
case3120sp	0.54	16.14	0.47	137.70	0.44	206.20
case3375wp	0.26	18.66	0.24	158.21	0.23	431.87
	0.46 %	**7.71**	**0.37%**	**56.74**	**0.35%**	**92.31**

Robustness

	SOCP		SOCPA		SOCP34AM	
instance	% gap	time	% gap	time	% gap	time
case6ww	0.68	0.07	0.02	0.29	0.01	0.34
case9	0.00	0.04	0.00	0.18	0.00	0.17
case9Q	0.06	0.03	0.06	0.16	0.06	0.17
case14	0.06	0.04	0.07	0.32	0.05	0.35
caseieee30	0.04	0.05	0.04	0.66	0.04	0.68
case39	0.02	0.09	0.01	0.85	0.01	0.92
case57	0.05	0.09	0.05	1.28	0.05	1.33
case118	0.21	0.20	0.21	3.24	0.14	4.25
case300	0.74	0.53	0.71	6.92	0.69	9.92
case2736sp	0.42	16.49	0.32	107.33	0.32	159.58
case2737sop	0.30	12.60	0.26	101.61	0.26	128.52
case2746wop	0.37	15.32	0.30	115.87	0.30	204.13
case2746wp	0.38	19.57	0.28	126.39	0.19	192.56
case3012wp	0.67	14.04	0.59	118.24	0.59	160.21
case3120sp	0.45	13.93	0.37	122.88	0.36	175.57
	0.30 %	**6.21**	**0.22** %	**47.08**	**0.20** %	**69.25**

5.2 Comparison to SDPs

Comparison with SDP: quality of bounds

instance	$\frac{z^{SOCP}}{z^{SDP}}$	$\frac{z^{SOCPA}}{z^{SDP}}$	$\frac{z^{SOCP34A}}{z^{SDP}}$
case6ww	0.9937	0.9998	0.9999
case9	1.0000	1.0000	1.0000
case14	0.9992	0.9992	0.9994
caseieee30	0.9996	0.9996	0.9996
case30	0.9943	0.9963	0.9966
case39	0.9998	0.9999	0.9999
case57	0.9994	0.9994	0.9994
case118	0.9976	0.9976	0.9984
case300	0.9985	0.9988	0.9989
case2383wp	0.9932	0.9949	0.9950
case2736sp	0.9956	0.9968	0.9969
case2737sop	0.9967	0.9972	0.9972
case2746wop	0.9952	0.9960	0.9960
case2746wp	0.9953	0.9966	0.9975
case3012wp	0.9936	0.9946	0.9946
case3120sp	0.9955	0.9962	0.9965
case3375wp	NA	NA	NA
	0.9967	**0.9977**	**0.9979**

Comparison with SDP: time

instance	SDP time	SOCP Time	SOCP A time	SOCP34A time
case6ww	1.66	0.02	0.40	0.43
case9	0.84	0.02	0.17	0.18
case14	1.07	0.02	0.41	0.45
caseieee30	1.84	0.03	0.78	0.84
case30	2.19	0.06	0.95	1.07
case39	2.20	0.04	0.90	0.99
case57	2.60	0.04	1.43	1.47
case118	4.58	0.11	3.69	4.83
case300	9.81	0.21	7.62	10.40
case2383wp	682.86	7.11	92.83	130.03
case2736sp	853.92	11.13	120.77	177.77
case2737sop	792.25	9.73	111.73	138.60
case2746wop	1138.06	11.37	124.01	218.45
case2746wp	941.04	14.51	124.19	195.43
case3012wp	746.08	7.28	143.10	185.56
case3120sp	904.90	7.33	127.90	196.05
case3375wp	>3hr	8.25	149.03	422.35

Some fnal conclusions

1. SOCP relaxation is quite good!
2. Solving SOCP based relaxation is orders of magnitude faster than solving SDPs.

Some fnal conclusions

1. SOCP relaxation is quite good!
2. Solving SOCP based relaxation is orders of magnitude faster than solving SDPs.
3. The arctan linearization adds non-trivial strength to the SOCP relaxation.
4. The McCormick relaxation of the 3,4 decomposition is comparable to the effect of arctan linearzation.

Some fnal conclusions

1. SOCP relaxation is quite good!
2. Solving SOCP based relaxation is orders of magnitude faster than solving SDPs.
3. The arctan linearization adds non-trivial strength to the SOCP relaxation.
4. The McCormick relaxation of the 3,4 decomposition is comparable to the effect of arctan linearzation.
5. The solutions from relaxation are a good starting point for interior point solver.

Some fnal conclusions

1. SOCP relaxation is quite good!
2. Solving SOCP based relaxation is orders of magnitude faster than solving SDPs.
3. The arctan linearization adds non-trivial strength to the SOCP relaxation.
4. The McCormick relaxation of the 3,4 decomposition is comparable to the effect of arctan linearzation.
5. The solutions from relaxation are a good starting point for interior point solver.
6. There is a need to develop *global solvers!*

Thank you

Thank You!