Power System State Estimation with a Limited Number of Measurements

Ramtin Madani and Javad Lavaei

Industrial Engineering and Operations Research
University of California, Berkeley

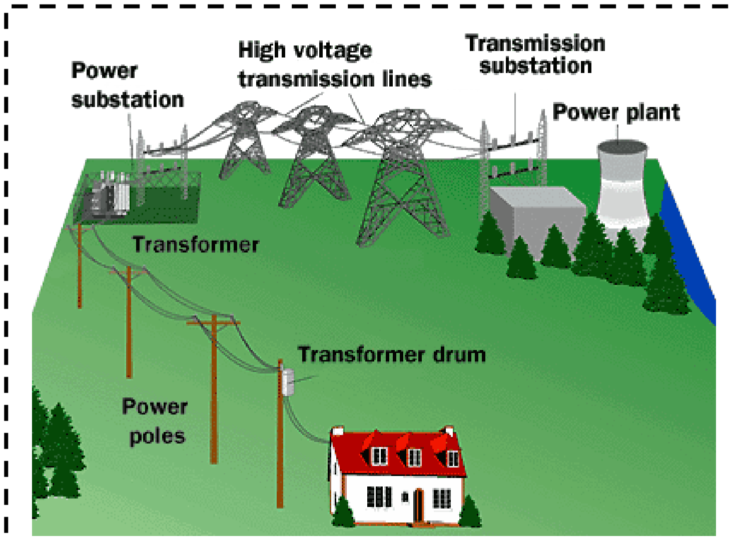
Ross Baldick

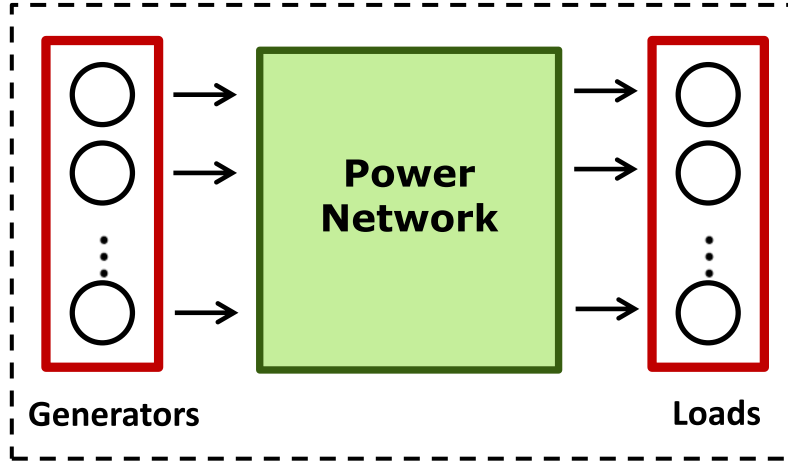
Electrical and Computer Engineering
University of Texas at Austin

Power System State Estimation (PSSE)

Network Parameters:

1. Voltages
2. Currents
3. Phase angles
4. Power injections
5. Power flows



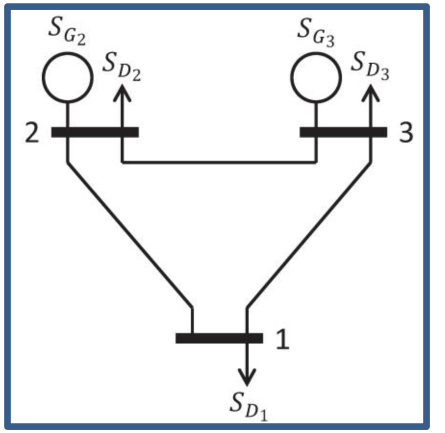


PSSE: Estimating the unknown parameters of the network, based on a limited number of measurements corrupted with noise and bad data.

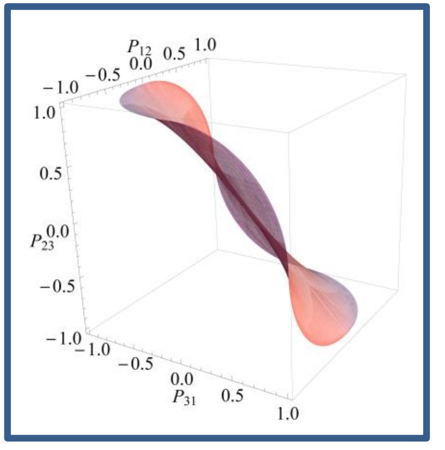
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

Simple network:



Feasible set:

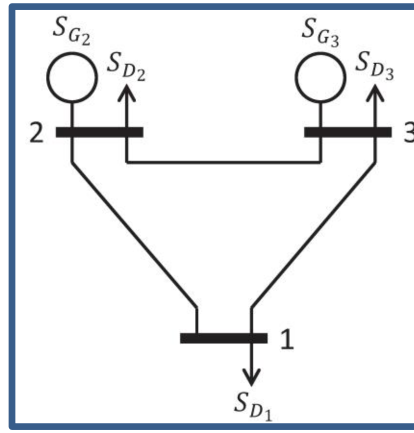


Power System State Estimation (PSSE)

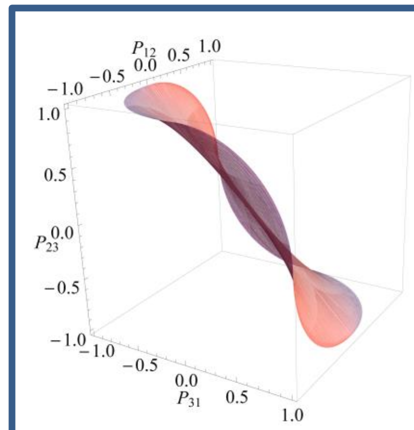
Noiseless Scenario: Power flow feasibility

Nonlinear
Network Equations

Simple network:



Feasible set:



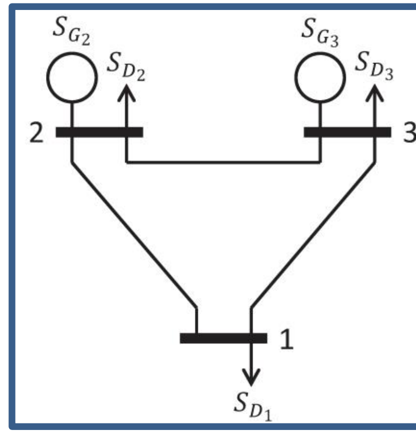
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

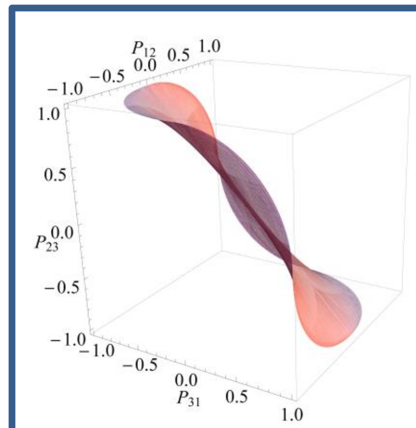
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].

Simple network:



Feasible set:



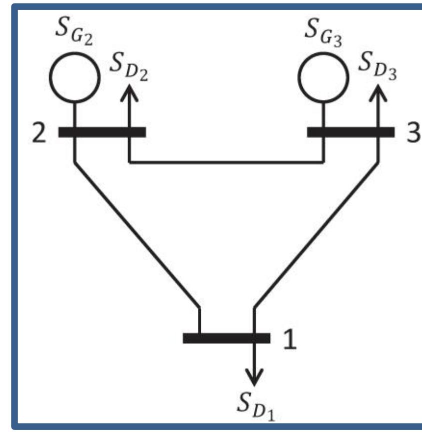
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

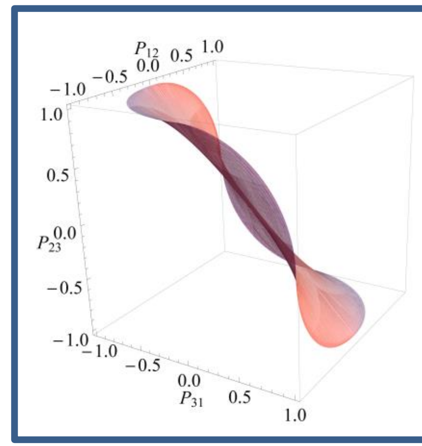
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.

Simple network:



Feasible set:



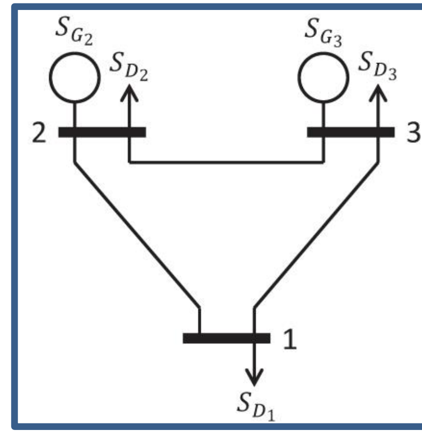
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

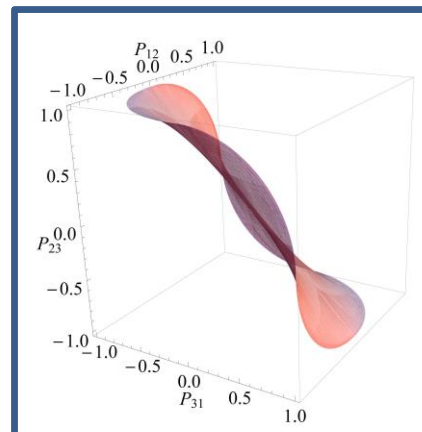
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

Simple network:



Feasible set:



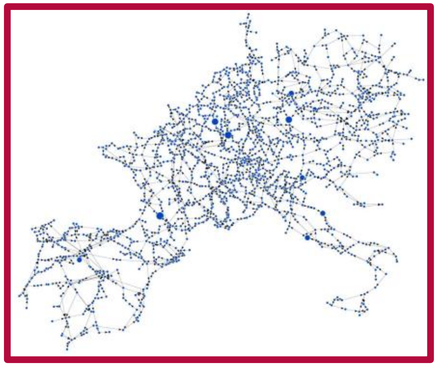
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

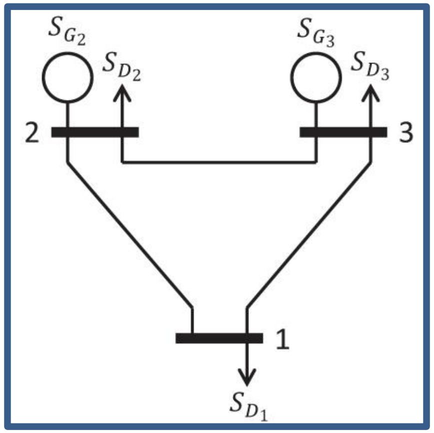
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

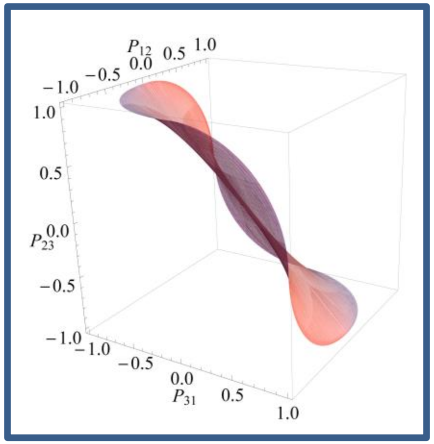
Noisy Case:



Simple network:



Feasible set:



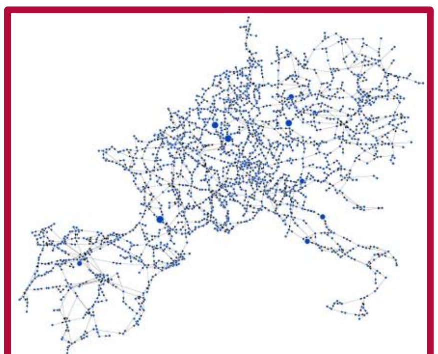
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

Nonlinear Network Equations → NP-hard

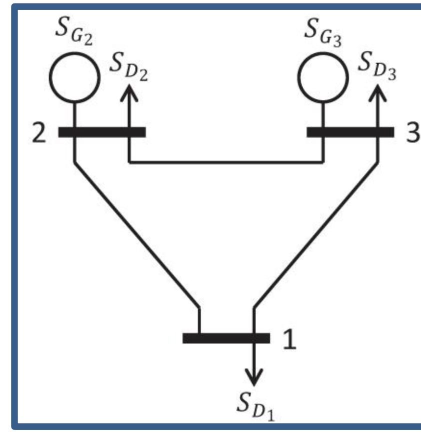
- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

Noisy Case:

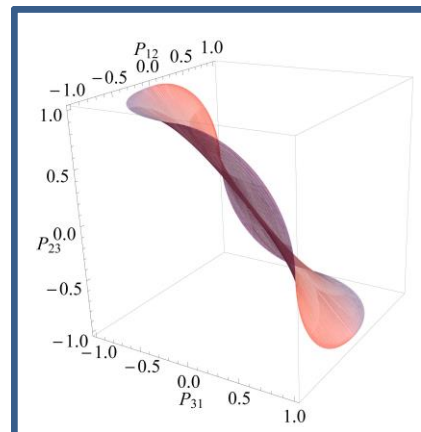


- European grid with 9241 nodes.

Simple network:

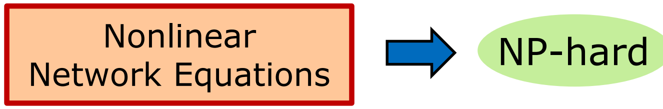


Feasible set:



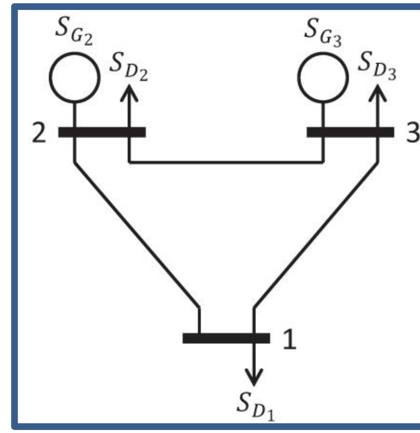
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

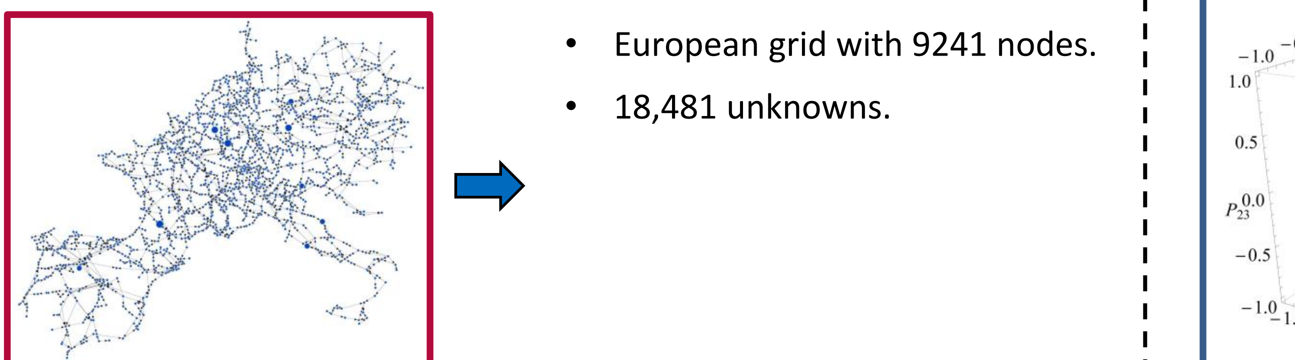


- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

Simple network:

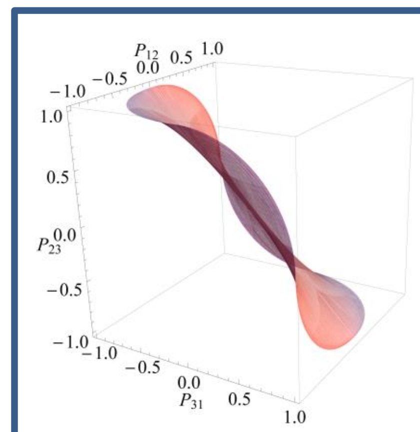


Noisy Case:



- European grid with 9241 nodes.
- 18,481 unknowns.

Feasible set:



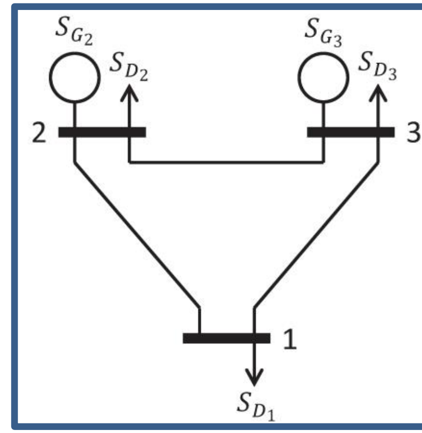
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

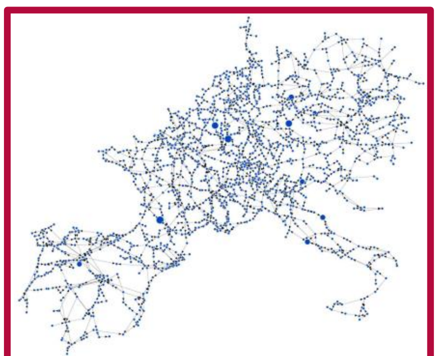
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

Simple network:

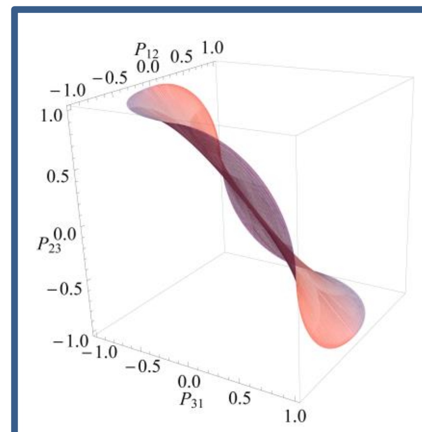


Noisy Case:



- European grid with 9241 nodes.
- 18,481 unknowns.
- 31,320 noisy measurements with 1% error.

Feasible set:



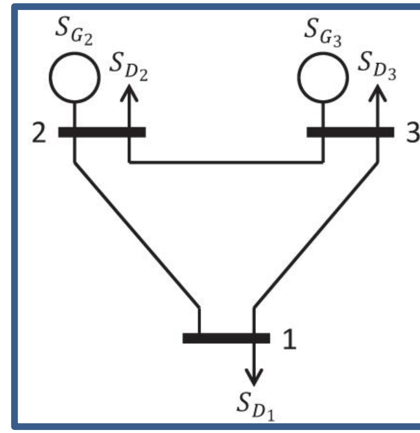
Power System State Estimation (PSSE)

Noiseless Scenario: Power flow feasibility

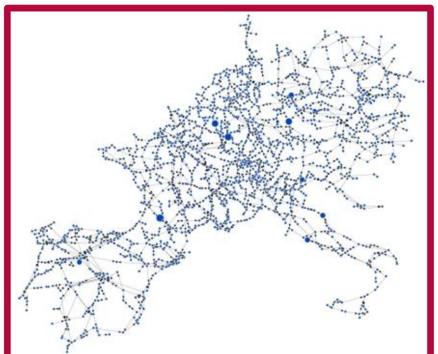
Nonlinear Network Equations → NP-hard

- **Complexity:** Strongly NP-hard [Bienstock et al., 09].
- **Common practice:** Linearization, local search algorithms.
- **Proposed Approach:** Convex optimization.

Simple network:



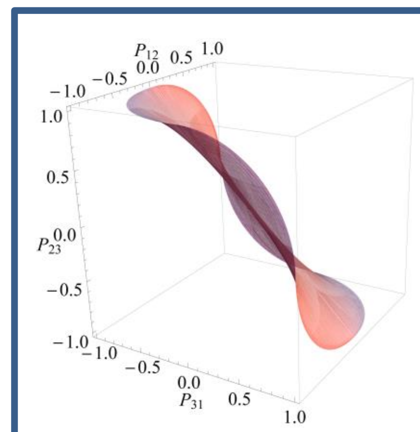
Noisy Case:



- European grid with 9241 nodes.
- 18,481 unknowns.
- 31,320 noisy measurements with 1% error.

Solution recovered with 0.5% accuracy

Feasible set:



Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\boxed{\left\{\begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} & = & y_m \end{array}\right.}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Vector of complex voltages

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} = y_m \end{cases}$$

Measured data

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\begin{cases} \mathbf{v}^{\mathrm{H}} \mathbf{M}_1 \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}} \mathbf{M}_2 \mathbf{v} = y_2 \\ \vdots \\ \mathbf{v}^{\mathrm{H}} \mathbf{M}_m \mathbf{v} = y_m \end{cases}$$

Network model

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \qquad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Noisy Case:

$$y_k = z_k + \omega_k + \eta_k$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = \underbrace{z_k}_{\text{True values}} + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\left\{ \begin{aligned} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &= y_2 \\ &\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &= y_m \end{aligned} \right.$$

Noisy Case:

$$y_k = z_k + \omega_k + \eta_k$$

Modest noise

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Sparse bad data

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.

Noiseless Scenario:

$$\boxed{\left\{ \begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} & = & y_m \end{array} \right.}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} = y_2 \\ \qquad \vdots \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:
  - Matrix completion, phase retrieval: [Recht et al., 2010], [Candes et al., 2013]

$$m = O(n \log(n))$$

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:
  - Matrix completion, phase retrieval: [Recht et al., 2010], [Candes et al., 2013]

    $$m = O(n \log(n))$$

  - Power System state estimation: [Giannakis et al., 2011], [Ilic et al., 2012].

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} = y_2 \\ \vdots \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:
  - Matrix completion, phase retrieval: [Recht et al., 2010], [Candes et al., 2013]

$$m = O(n \log(n))$$

  - Power System state estimation: [Giannakis et al., 2011], [Ilic et al., 2012].
- **Our focus:** When the number of equations is close to unknowns.

Noiseless Scenario:

$$\boxed{\left\{\begin{aligned} \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_1\ \mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_2\ \mathbf{v} &= y_2 \\ &\ \vdots \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_m\ \mathbf{v} &= y_m \end{aligned}\right.}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:
  - Matrix completion, phase retrieval: [Recht et al., 2010], [Candes et al., 2013]

$$m = O(n \log(n))$$

  - Power System state estimation: [Giannakis et al., 2011], [Ilic et al., 2012].
- **Our focus:** When the number of equations is close to unknowns.
- **Challenges:**
  - Nonconvexity
  - Noise

Noiseless Scenario:

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} = y_m \end{cases}}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Quadratic Equations

- **State of the network:** $\mathbf{v} \in \mathbb{C}^n$
- **Communities:** Power, Signal processing and Operations research.
- Easy with a large number of equations:
  - Matrix completion, phase retrieval: [Recht et al., 2010], [Candes et al., 2013]

$$m = O(n \log(n))$$

  - Power System state estimation: [Giannakis et al., 2011], [Ilic et al., 2012].
- **Our focus:** When the number of equations is close to unknowns.
- **Challenges:**
  - Nonconvexity
  - Noise

→ **Penalized semidefinite Programing (SDP)**

Noiseless Scenario:

$$\boxed{\left\{ \begin{aligned} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &= y_2 \\ &\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &= y_m \end{aligned} \right.}$$

Noisy Case:

$$\boxed{y_k = z_k + \omega_k + \eta_k}$$

Nonconvexity: Semidefinite Relaxation

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_1\, \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_2\, \mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\, \mathbf{M}_m\, \mathbf{v} = y_m \end{cases}$$

High dimensional space

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \quad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$

$$\left\{\begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &=& y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &=& y_2 \\ &\vdots& \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &=& y_m \end{array}\right.$$

High dimensional space

$$\left\{\begin{array}{rcl} \langle \mathbf{M}_1, \mathbf{W} \rangle &=& y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle &=& y_2 \\ &\vdots& \\ \langle \mathbf{M}_m, \mathbf{W} \rangle &=& y_m \\ \mathbf{W} \succeq 0. && \end{array}\right.$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &= y_2 \\ &\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &= y_m \end{cases}$$

High dimensional space

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle &= y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle &= y_2 \\ &\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle &= y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \to W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\mathrm{rank}\{\mathbf{W}\} = 1}$$

$$\left\{ \begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} & = & y_m \end{array} \right.$$

High dimensional space

$$\left\{ \begin{array}{rcl} \langle \mathbf{M}_1, \mathbf{W} \rangle & = & y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle & = & y_2 \\ & \vdots & \\ \langle \mathbf{M}_m, \mathbf{W} \rangle & = & y_m \\ \mathbf{W} \succeq 0. & & \end{array} \right.$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \to W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\mathrm{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].

$$\boxed{\left\{ \begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_1\ \mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_2\ \mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_m\ \mathbf{v} & = & y_m \end{array} \right.}$$

High dimensional space

$$\boxed{\left\{ \begin{array}{rcl} \langle \mathbf{M}_1, \mathbf{W} \rangle & = & y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle & = & y_2 \\ & \vdots & \\ \langle \mathbf{M}_m, \mathbf{W} \rangle & = & y_m \\ \mathbf{W} \succeq 0. & & \end{array} \right.}$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\operatorname{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].
- Finding a rank-1 solution, solves the original problem.

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}}$$

High dimensional space

$$\boxed{\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}}$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\mathrm{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].
- Finding a rank-1 solution, solves the original problem.
- How to find a rank one solution?

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

High dimensional space

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \quad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\mathrm{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].
- Finding a rank-1 solution, solves the original problem.
- How to find a rank one solution?
- How to reject bad data?

$$\left\{ \begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} & = & y_m \end{array} \right.$$

High dimensional space

$$\left\{ \begin{array}{rcl} \langle \mathbf{M}_1, \mathbf{W} \rangle & = & y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle & = & y_2 \\ & \vdots & \\ \langle \mathbf{M}_m, \mathbf{W} \rangle & = & y_m \\ \mathbf{W} \succeq 0. & & \end{array} \right.$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \rightarrow W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\operatorname{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].
- Finding a rank-1 solution, solves the original problem.
- How to find a rank one solution?
- How to reject bad data?
- How to bound the error for noisy scenarios?

$$\left\{ \begin{array}{rcl} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} & = & y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} & = & y_2 \\ & \vdots & \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} & = & y_m \end{array} \right.$$

High dimensional space

$$\left\{ \begin{array}{rcl} \langle \mathbf{M}_1, \mathbf{W} \rangle & = & y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle & = & y_2 \\ & \vdots & \\ \langle \mathbf{M}_m, \mathbf{W} \rangle & = & y_m \\ \mathbf{W} \succeq 0. & & \end{array} \right.$$

Nonconvexity: Semidefinite Relaxation

- Lifting : $\boxed{v_i v_j^* \to W_{ij}}$
- The problem can be cast linearly w.r.t. to the matrix $\mathbf{W} \triangleq \mathbf{v}\mathbf{v}^{\mathrm{H}}$:

$$\boxed{\mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = \langle \mathbf{M}_k, \mathbf{v}\mathbf{v}^{\mathrm{H}} \rangle = \langle \mathbf{M}_k, \mathbf{W} \rangle}$$

- The matrix $\mathbf{W}$ is structured:

$$\boxed{\mathbf{W} \succeq 0,} \qquad \boxed{\cancel{\mathrm{rank}\{\mathbf{W}\} = 1}}$$

- Semidefinite programing (SDP) relaxation [Alizadeh 91] and [Nesterov et al., 94].
- Finding a rank-1 solution, solves the original problem.
- How to find a rank one solution?
- How to reject bad data?
- How to bound the error for noisy scenarios?

$$\boxed{\begin{cases} \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_1\ \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_2\ \mathbf{v} = y_2 \\ \qquad \vdots \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_m\ \mathbf{v} = y_m \end{cases}}$$

High dimensional space

$$\boxed{\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \qquad \vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}}$$

Nonconvexity: Penalized SDP

Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Nonconvexity: Penalized SDP

Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \quad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

Nonconvexity: Penalized SDP

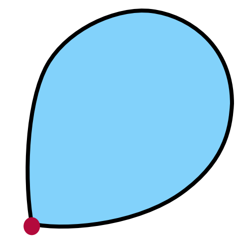
Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \qquad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \qquad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

High dimensional set:



Nonconvexity: Penalized SDP

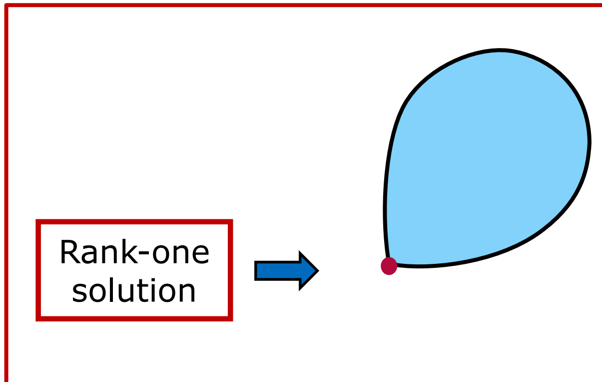
Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} = y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \quad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

High dimensional set:



Nonconvexity: Penalized SDP

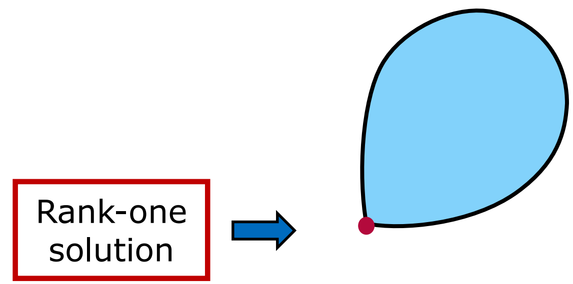
Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_1\ \mathbf{v} = y_1 \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_2\ \mathbf{v} = y_2 \\ \quad\vdots \\ \mathbf{v}^{\mathrm{H}}\ \mathbf{M}_m\ \mathbf{v} = y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle = y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle = y_2 \\ \quad\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle = y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

High dimensional set:



- Extra equations turn the feasible set into a single point.

Nonconvexity: Penalized SDP

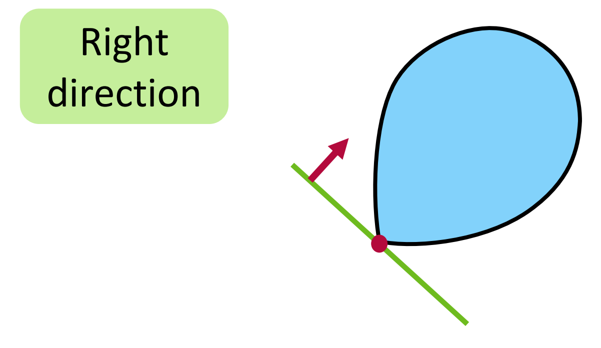
Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &= y_2 \\ &\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &= y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle &= y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle &= y_2 \\ &\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle &= y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

High dimensional set:



- Extra equations turn the feasible set into a single point.
- **Our approach:** Design a linear cost $g(\mathbf{W}) \triangleq \langle \mathbf{M}, \mathbf{W} \rangle$ that imposes a right direction.

Nonconvexity: Penalized SDP

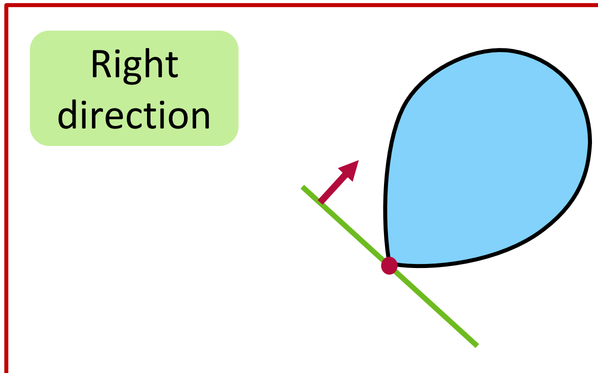
Original space:

$$\begin{cases} \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_1\,\mathbf{v} &= y_1 \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_2\,\mathbf{v} &= y_2 \\ &\vdots \\ \mathbf{v}^{\mathrm{H}}\,\mathbf{M}_m\,\mathbf{v} &= y_m \end{cases}$$

Lifted space:

$$\begin{cases} \langle \mathbf{M}_1, \mathbf{W} \rangle &= y_1 \\ \langle \mathbf{M}_2, \mathbf{W} \rangle &= y_2 \\ &\vdots \\ \langle \mathbf{M}_m, \mathbf{W} \rangle &= y_m \\ \mathbf{W} \succeq 0. \end{cases}$$

High dimensional set:



- Extra equations turn the feasible set into a single point.
- **Our approach:** Design a linear cost $g(\mathbf{W}) \triangleq \langle \mathbf{M}, \mathbf{W} \rangle$ that imposes a right direction.
- Use $g(\mathbf{W})$ as the objective to build an optimization problem:

Penalized SDP:

$$\begin{aligned} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} \quad & g(\mathbf{W}) \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, \ldots, m \\ & \mathbf{W} \succeq 0 \end{aligned}$$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\Rightarrow$$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

- Reactive power: $\mathbf{M} = -\mathbf{B}$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

- Reactive power: $\mathbf{M} = -\mathbf{B}$ $\Rightarrow$ Susceptance matrix

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

- Reactive power: $\mathbf{M} = -\mathbf{B}$ $\Rightarrow$ Susceptance matrix
- Current injections: $\mathbf{M} = \mathbf{Y}^{\mathrm{H}} \mathbf{Y}$

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

- Reactive power: $\mathbf{M} = -\mathbf{B}$ $\Rightarrow$ Susceptance matrix
- Current injections: $\mathbf{M} = \mathbf{Y}^{\mathrm{H}}\mathbf{Y}$ $\Rightarrow$ Admittance matrix

Nonconvexity: Penalty Design

How to design penalty functions for power systems applications?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & g(\mathbf{W}) \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Common practice:

- Nuclear norm: $g(\mathbf{W}) = \mathrm{trace}\{\mathbf{W}\}$ $\Rightarrow$ Convex envelope of the rank function

Proposed penalty: $g(\mathbf{W}) = \langle \mathbf{M}, \mathbf{W} \rangle$

- Reactive power: $\mathbf{M} = -\mathbf{B}$ $\Rightarrow$ Susceptance matrix
- Current injections: $\mathbf{M} = \mathbf{Y}^{\mathrm{H}}\mathbf{Y}$ $\Rightarrow$ Admittance matrix

Theorem: Minimization of "reactive power" or "current magnitudes" both recover every solution in a regime where the phase angles are small.

Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \ k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Guess for solution: $\hat{\mathbf{v}}$

1. $\mathbf{M} \succeq 0$
2. $\mathbf{M}\,\hat{\mathbf{v}} = 0$
3. 0 is a simple eigenvalue for $\mathbf{M}$

Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \ k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Guess for solution: $\hat{\mathbf{v}}$

1. $\mathbf{M} \succeq 0$
2. $\mathbf{M}\,\hat{\mathbf{v}} = 0$
3. 0 is a simple eigenvalue for $\mathbf{M}$

Theorem: If there exists a solution in a vicinity of $\hat{\mathbf{v}}$ then SDP is exact.

Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

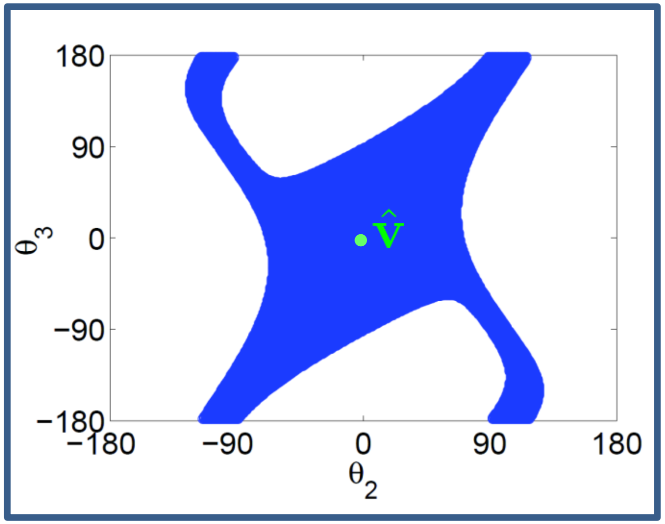
$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Guess for solution: $\hat{\mathbf{v}}$

1. $\mathbf{M} \succeq 0$
2. $\mathbf{M}\,\hat{\mathbf{v}} = 0$
3. 0 is a simple eigenvalue for $\mathbf{M}$

Theorem: If there exists a solution in a vicinity of $\hat{\mathbf{v}}$ then SDP is exact.

Recovery region $\mathcal{R}_{\mathbf{M}} \in \mathbb{C}^n$:



Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

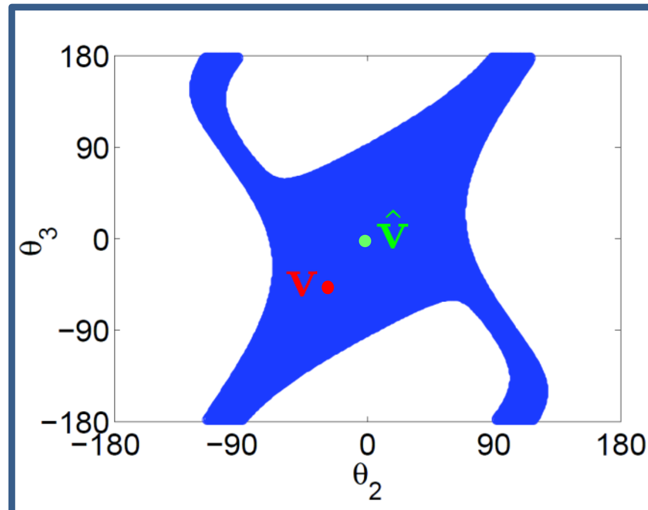
$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Guess for solution: $\hat{\mathbf{v}}$

1. $\mathbf{M} \succeq 0$
2. $\mathbf{M}\,\hat{\mathbf{v}} = 0$
3. 0 is a simple eigenvalue for $\mathbf{M}$

Theorem: If there exists a solution in a vicinity of $\hat{\mathbf{v}}$ then SDP is exact.

Recovery region $\mathcal{R}_{\mathbf{M}} \in \mathbb{C}^n$:



Nonconvexity: Penalty Design

How to design penalty functions for an arbitrary problem?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} = y_k, \ k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{\mathbf{W} \in \mathbb{H}_n}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle = y_k, \quad k = 1, 2, \ldots, m \\ & \mathbf{W} \succeq 0 \end{array}$$

Guess for solution: $\hat{\mathbf{v}}$

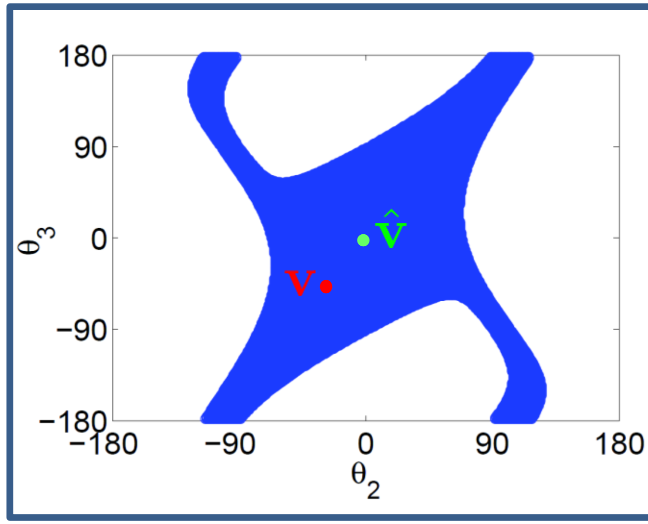
1. $\mathbf{M} \succeq 0$
2. $\mathbf{M}\,\hat{\mathbf{v}} = 0$
3. 0 is a simple eigenvalue for $\mathbf{M}$

Theorem: If there exists a solution in a vicinity of $\hat{\mathbf{v}}$ then SDP is exact.

Theorem: There are finite number of $\mathbf{M}$'s that cover the entire space.

$\Rightarrow$

Recovery region $\mathcal{R}_{\mathbf{M}} \in \mathbb{C}^n$:



Comparisons with Newton's Method

Randomly chosen phase angles and voltage magnitude:

$$(\measuredangle v_1, \measuredangle v_2, \ldots, \measuredangle v_n) \in [-\theta, \theta]^n$$

$$(|v_1|, |v_2|, \ldots, |v_n|) \in [0.9, 1.1]^n$$

Comparisons with Newton's Method

Randomly chosen phase angles and voltage magnitude:

$$(\measuredangle v_1, \measuredangle v_2, \ldots, \measuredangle v_n) \in [-\theta, \theta]^n$$
$$(|v_1|, |v_2|, \ldots, |v_n|) \in [0.9, 1.1]^n$$

➡

Derive the corresponding set of specifications

Recover an operating point using SDP

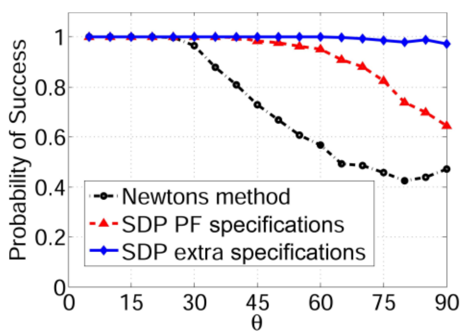
Comparisons with Newton's Method

Randomly chosen phase angles and voltage magnitude:

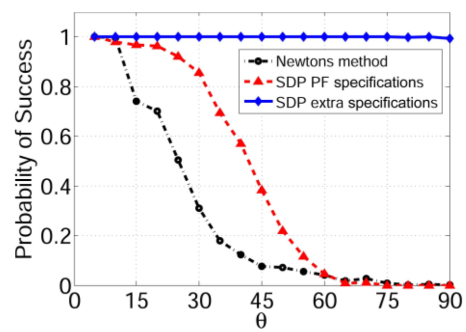
$$(\measuredangle v_1, \measuredangle v_2, \ldots, \measuredangle v_n) \in [-\theta, \theta]^n$$
$$(|v_1|, |v_2|, \ldots, |v_n|) \in [0.9, 1.1]^n$$

Derive the corresponding set of specifications

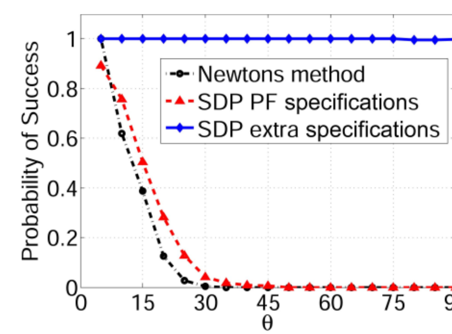
Recover an operating point using SDP



IEEE 9-bus system



New England 39-bus



IEEE 57-bus system

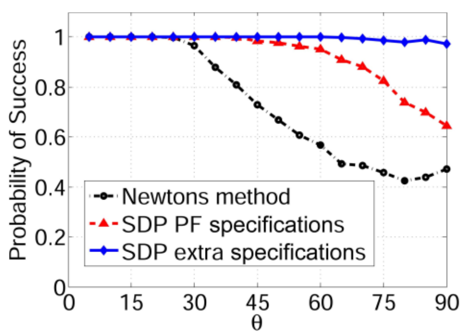
Comparisons with Newton's Method

Randomly chosen phase angles and voltage magnitude:

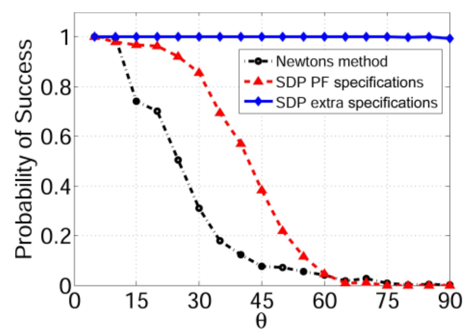
$$(\measuredangle v_1, \measuredangle v_2, \ldots, \measuredangle v_n) \in [-\theta, \theta]^n$$
$$(|v_1|, |v_2|, \ldots, |v_n|) \in [0.9, 1.1]^n$$

Derive the corresponding set of specifications

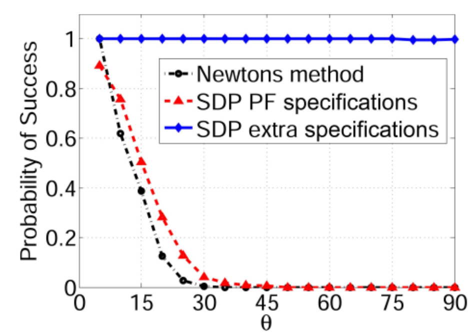
Recover an operating point using SDP



IEEE 9-bus system



New England 39-bus



IEEE 57-bus system

- The proposed approach outperforms Newton’s method.

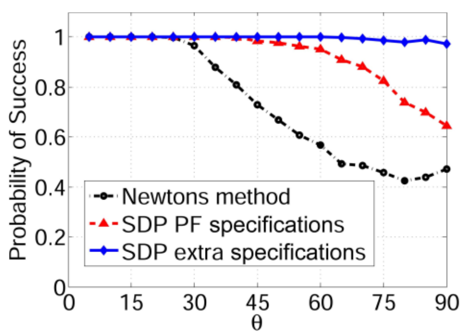
Comparisons with Newton's Method

Randomly chosen phase angles and voltage magnitude:

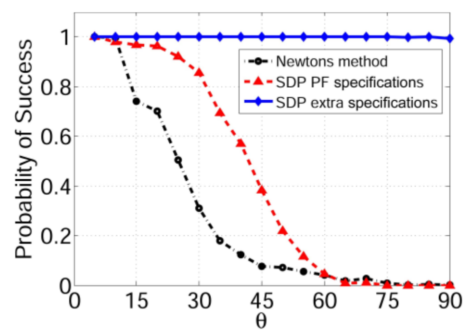
$$(\measuredangle v_1, \measuredangle v_2, \ldots, \measuredangle v_n) \in [-\theta, \theta]^n$$
$$(|v_1|, |v_2|, \ldots, |v_n|) \in [0.9, 1.1]^n$$

Derive the corresponding set of specifications

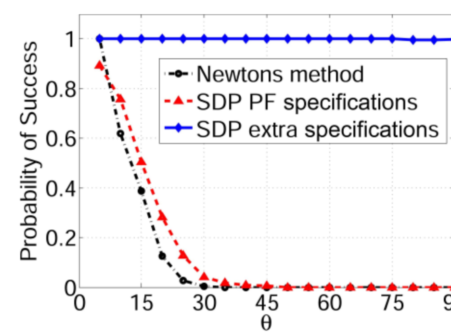
Recover an operating point using SDP



IEEE 9-bus system



New England 39-bus



IEEE 57-bus system

- The proposed approach outperforms Newton's method.
- Given additional equations, SDP is almost always exact.

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \omega_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\begin{array}{ll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{array}$$

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \omega_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\Rightarrow$$

$$\begin{array}{ll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{array}$$

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \omega_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\Rightarrow$$

$$\begin{array}{lll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Theorem (Noiseless Case): If Jacobian is nonsingular, the initial guess is sufficiently close and μ is large, then SDP+L1 is exact.

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \omega_k = y_k, \ \ k = 1, \ldots, m \end{array}$$

$$\begin{array}{ll} \underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{array}$$

Theorem (Noiseless Case): If Jacobian is nonsingular, the initial guess is sufficiently close and μ is large, then SDP+L1 is exact.

Theorem (Bad Data): A limited number of wrong equations have no effect on the solution.

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \omega_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{lll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Theorem (Noiseless Case): If Jacobian is nonsingular, the initial guess is sufficiently close and μ is large, then SDP+L1 is exact.

Theorem (Bad Data): A limited number of wrong equations have no effect on the solution.

Theorem (Small Noise): If $\mathbf{x}$ is recoverable with noiseless measurements, then every solution $(\mathbf{W}^{\text{opt}}, \boldsymbol{\nu}^{\text{opt}})$ of penalized SDP + L1 satisfies:

$$\|\mathbf{W}^{\text{opt}} - \alpha \mathbf{v}\mathbf{v}^*\|_F \leq c \times \sqrt{\mu \times \|\boldsymbol{\omega}\|_1 \times \text{trace}\{\mathbf{W}^{\text{opt}}\}}$$

Noisy Scenarios

How to handle noise?

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

Elevates the likelihood of estimated noise values

1- Y. Zhang, R. Madani and J. Lavaei, "Power System State Estimation with Line Measurements," Preprint, 2016.

Noisy Scenarios

How to handle noise?

$$
\begin{aligned}
\underset{\substack{\mathbf{W}\in\mathbb{H}_n\\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\
\text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\
& \mathbf{W} \succeq 0
\end{aligned}
$$

Elevates the likelihood of estimated noise values

- What is the statistical behavior of the bound?

1- Y. Zhang, R. Madani and J. Lavaei, "Power System State Estimation with Line Measurements," Preprint, 2016.

Noisy Scenarios

How to handle noise?

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

Elevates the likelihood of estimated noise values

- What is the statistical behavior of the bound?
- How does the number of measurements effect the quality of estimation?

1- Y. Zhang, R. Madani and J. Lavaei, "Power System State Estimation with Line Measurements," Preprint, 2016.

Noisy Scenarios

How to handle noise?

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

Elevates the likelihood of estimated noise values

- What is the statistical behavior of the bound?
- How does the number of measurements effect the quality of estimation?
- Define the root-mean-square error:

$$\zeta := \frac{\|\mathbf{W}^{\text{opt}} - \alpha \mathbf{v}\mathbf{v}^*\|_F}{\sqrt{n \times \text{trace}\{\mathbf{W}^{\text{opt}}\}}}$$

1- Y. Zhang, R. Madani and J. Lavaei, "Power System State Estimation with Line Measurements," Preprint, 2016.

Noisy Scenarios

How to handle noise?

$$\begin{array}{ll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{array}$$

$\Rightarrow$ Elevates the likelihood of estimated noise values

- What is the statistical behavior of the bound?
- How does the number of measurements effect the quality of estimation?
- Define the root-mean-square error:

$$\zeta := \frac{\|\mathbf{W}^{\text{opt}} - \alpha \mathbf{v}\mathbf{v}^*\|_F}{\sqrt{n \times \text{trace}\{\mathbf{W}^{\text{opt}}\}}}$$

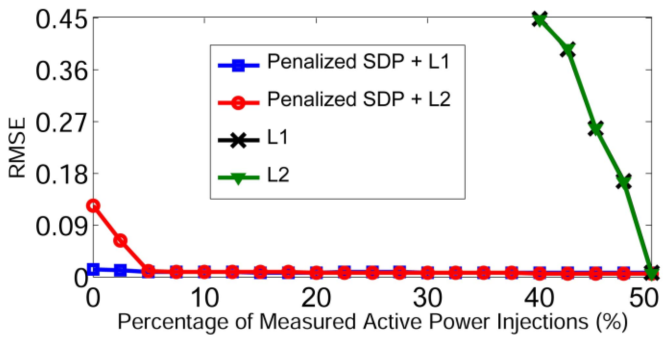
Theorem: Under recoverability assumption, the tail probability of the root-mean-square estimation error is upper bounded as:

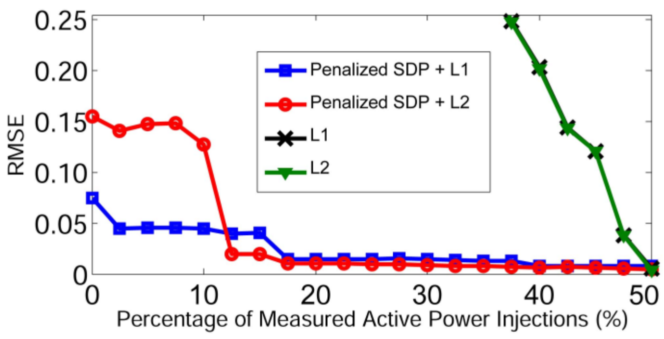
$$\mathbb{P}(\zeta > t) \leq \mathrm{e}^{-\gamma(t)m} \quad \text{where} \quad \gamma(t) := \frac{t^4 \lambda^2}{32\kappa^2\rho^2} - \ln 2.$$

1- Y. Zhang, R. Madani and J. Lavaei, “Power System State Estimation with Line Measurements,” Preprint, 2016.

Case Studies

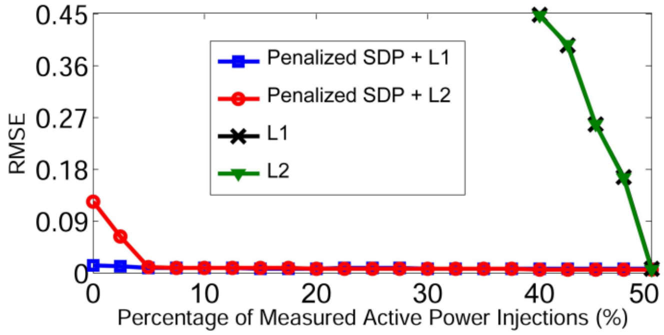
- A 1354 node European grid with 1% and 2% measurement noise values:

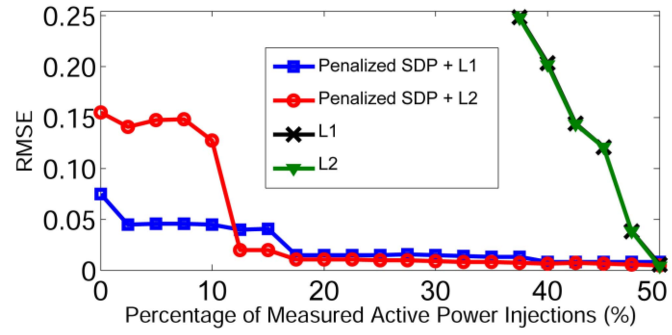




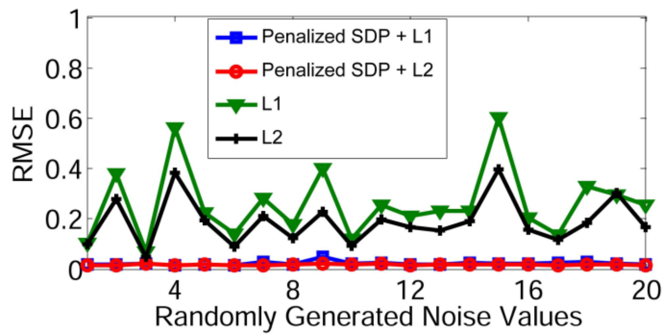
Case Studies

- A 1354 node European grid with 1% and 2% measurement noise values:



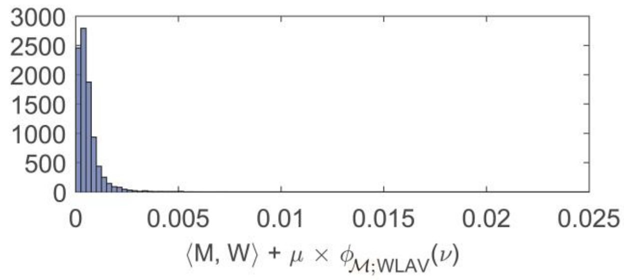


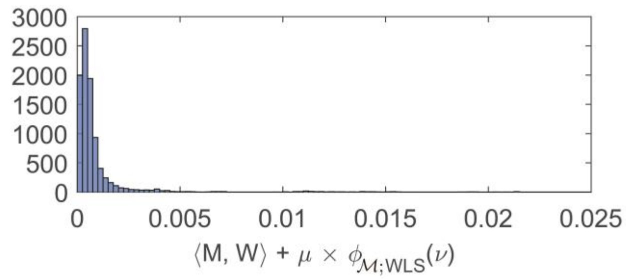
- Random realizations with 5% measurement noise and 30% extra equations:



Case Studies

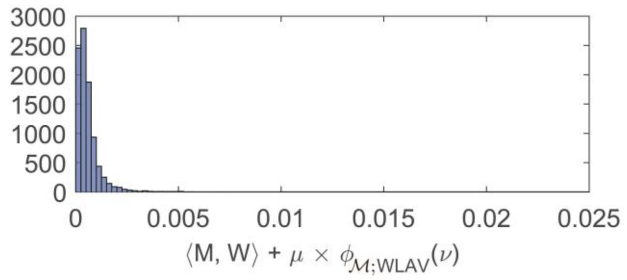
- Histograms of absolute differences between the actual and estimated complex voltages for the PEGASE 9241-bus system, using the penalized convex problem equipped with the WLAV and WLS estimators respectively:

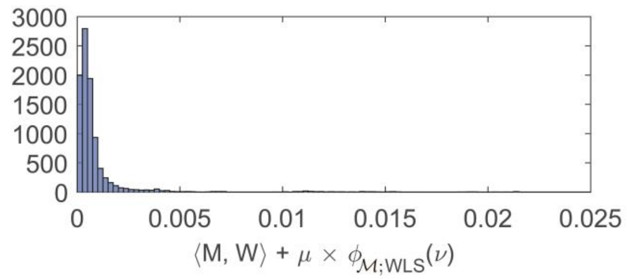




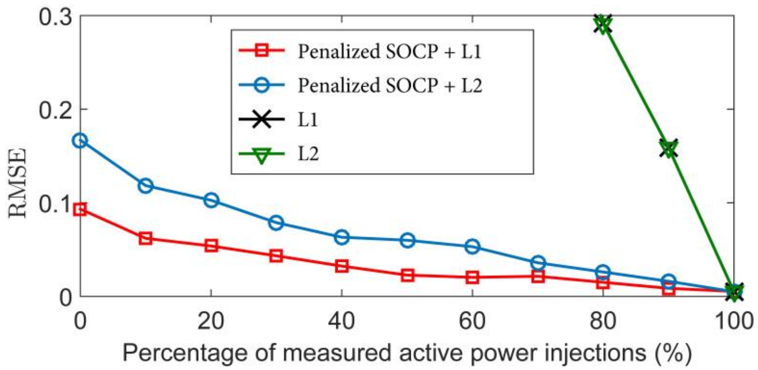
Case Studies

- Histograms of absolute differences between the actual and estimated complex voltages for the PEGASE 9241-bus system, using the penalized convex problem equipped with the WLAV and WLS estimators respectively:



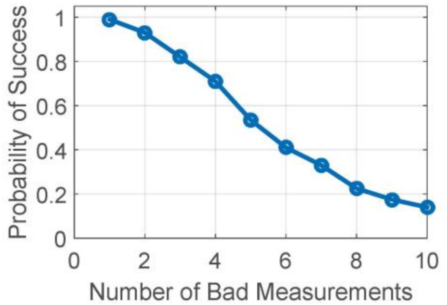


- Average RMSE of the estimated voltages obtained by SOCP over 10 Monte-Carlo simulations for different noise realizations:

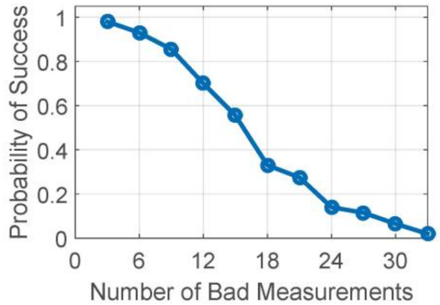


Case Studies

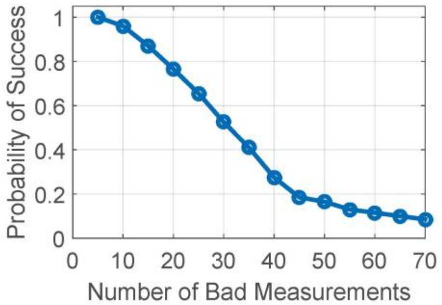
- The probability of success for the penalized SDP problem when different numbers of measurements are corrupted for New England 39-bus, IEEE 57-bus and IEEE 118-bus systems, respectively:



New England 39-bus

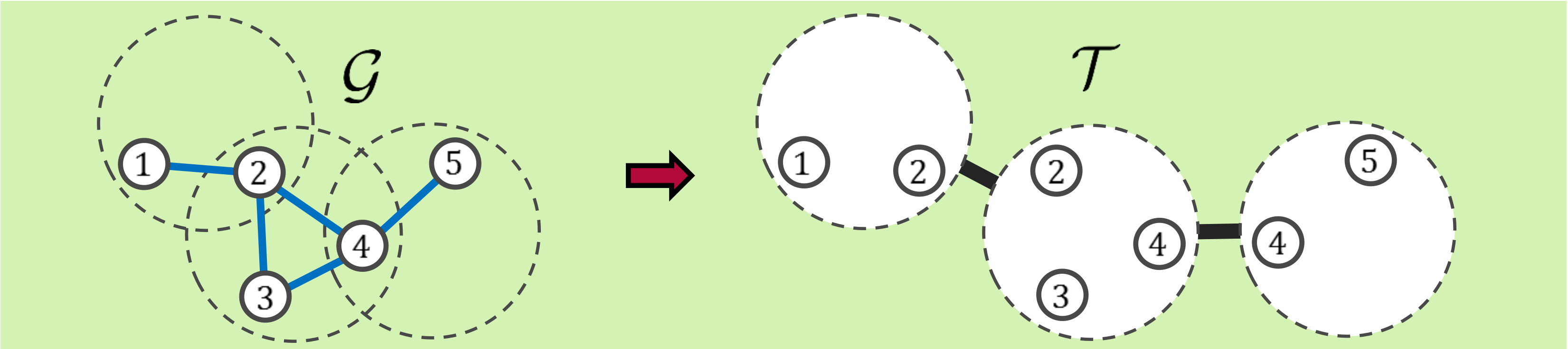


IEEE 57-bus

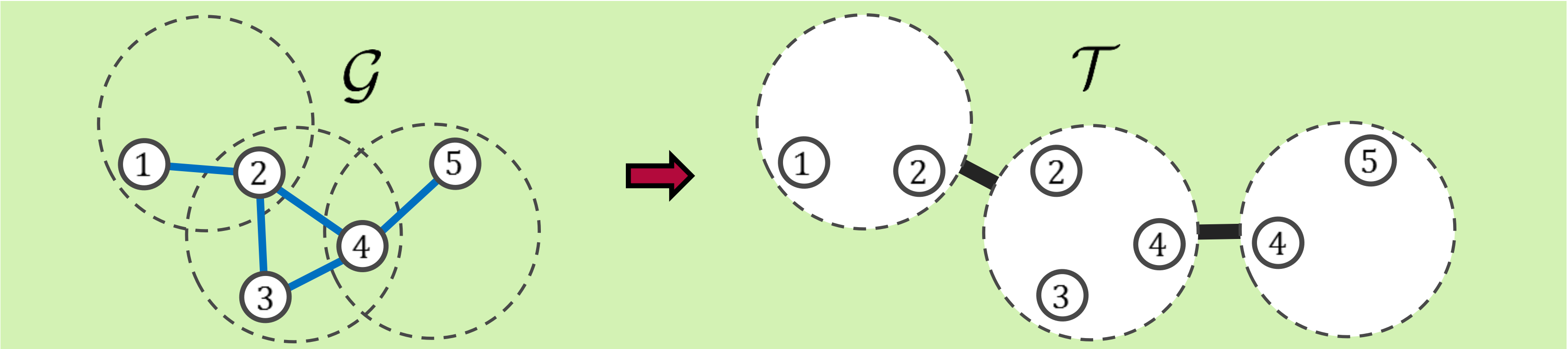


IEEE 118-bus

Tree Decomposition

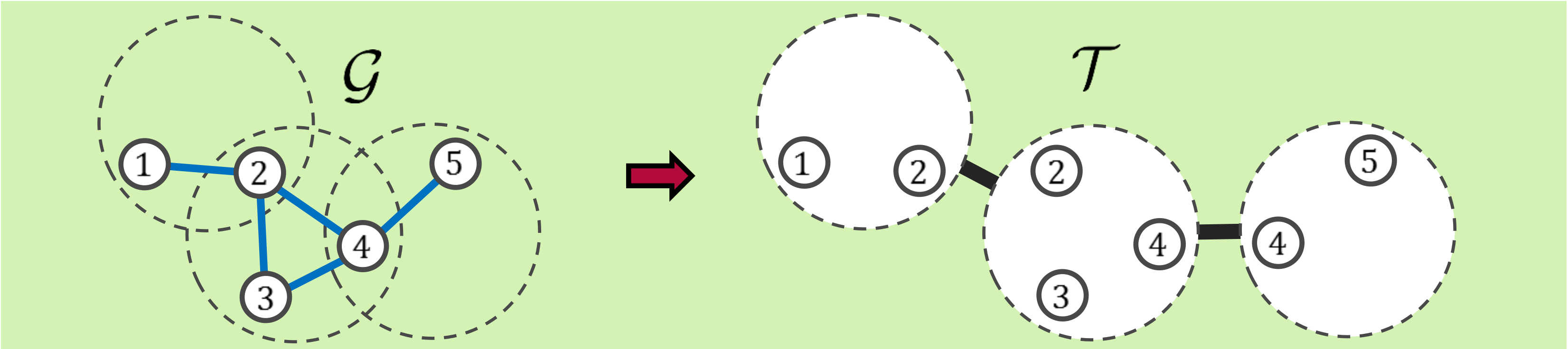


Tree Decomposition



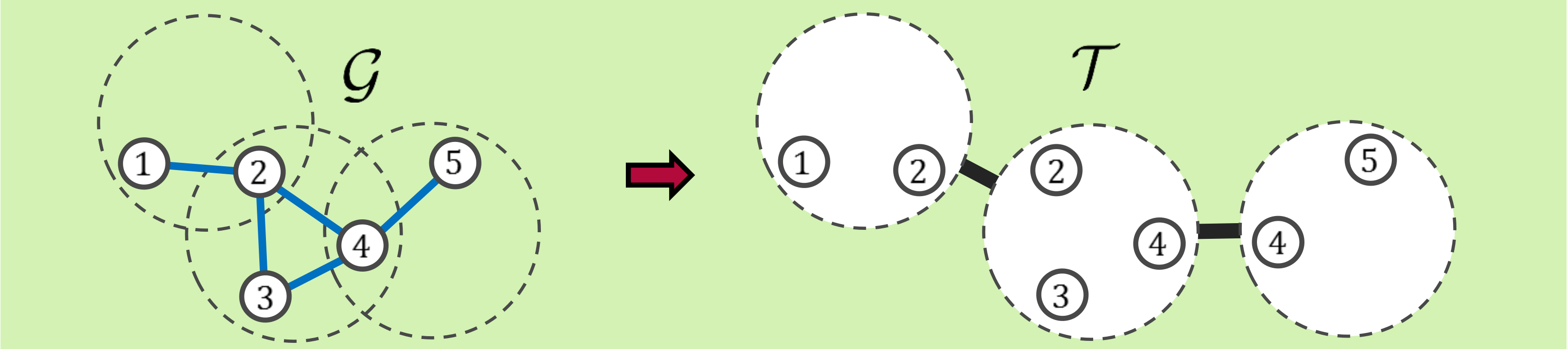
- **Tree decomposition:** A way of grouping the vertices in order to turn the graph into a tree [Robertson and Seymour, 84].

Tree Decomposition



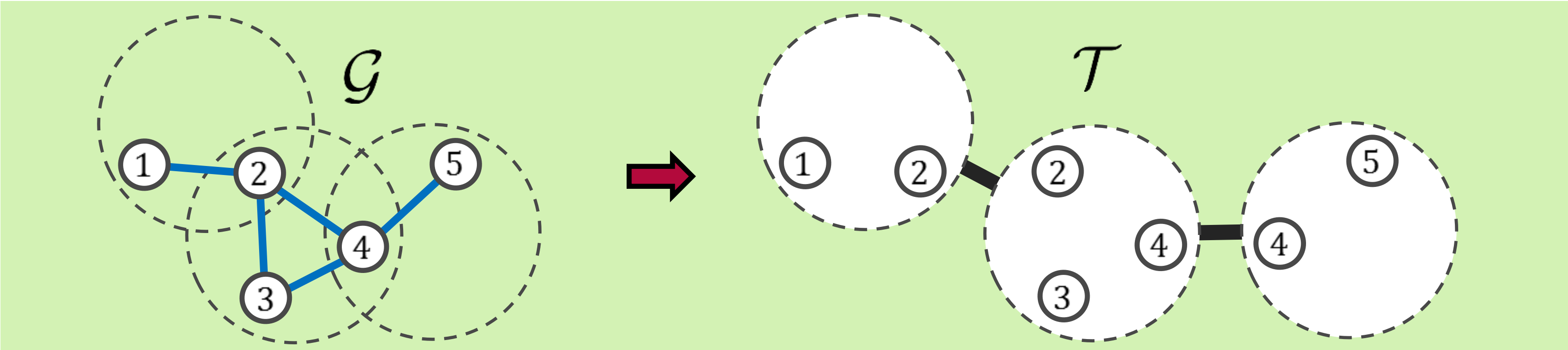
- **Tree decomposition:** A way of grouping the vertices in order to turn the graph into a tree [Robertson and Seymour, 84].
- **Tree width:** The most efficient way of grouping in terms of the size for largest group.

Tree Decomposition



- **Tree decomposition:** A way of grouping the vertices in order to turn the graph into a tree [Robertson and Seymour, 84].
- **Tree width:** The most efficient way of grouping in terms of the size for largest group.
- Very efficient methods to calculating upper bounds [Bodlaender, 11].

Tree Decomposition



- **Tree decomposition:** A way of grouping the vertices in order to turn the graph into a tree [Robertson and Seymour, 84].
- **Tree width:** The most efficient way of grouping in terms of the size for largest group.
- Very efficient methods to calculating upper bounds [Bodlaender, 11].

TW of the New York power grid with 8500 nodes ≤ 40.

TW of the European power grid 9000 nodes ≤ 31.

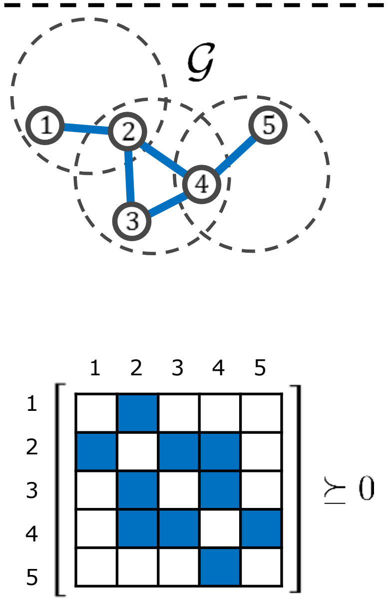
Decomposed SDP

Quadratic Problem:

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \nu_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\begin{array}{lll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Full-scale SDP:



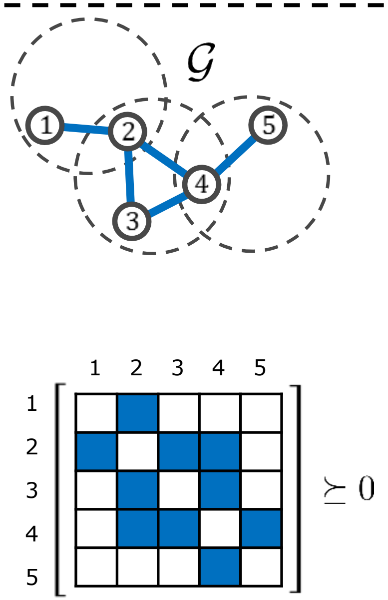
Decomposed SDP

Quadratic Problem:

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \nu_k = y_k, \;\; k = 1, \ldots, m \end{array}$$

$$\begin{array}{lll} \underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Full-scale SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].

Decomposed SDP

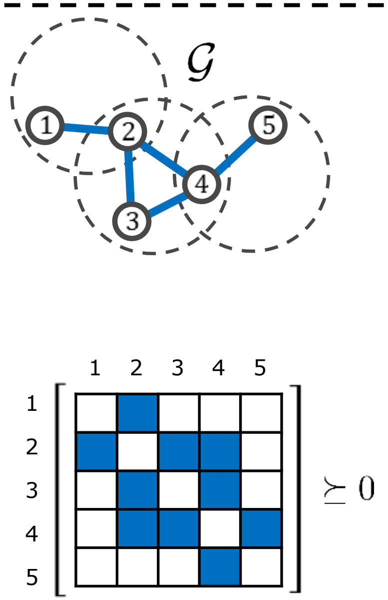
Quadratic Problem:

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{array}$$

$\Rightarrow$

$$\begin{array}{lll} \underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Full-scale SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?

Decomposed SDP

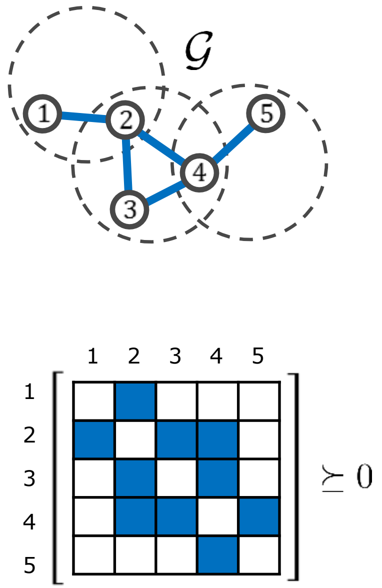
Quadratic Problem:

$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{array}$$

$$\Rightarrow$$

$$\begin{array}{lll} \underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Full-scale SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?
- The high-dim conic constraint can be decomposed to smaller order cones.

Decomposed SDP

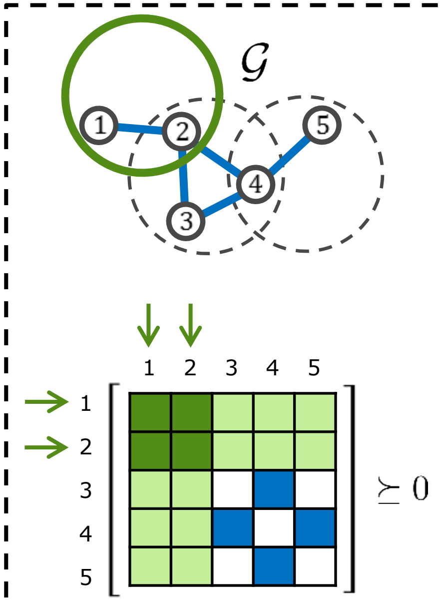
Quadratic Problem:

$$\begin{aligned} &\text{find} && \mathbf{v} \in \mathbb{C}^n \\ &\text{subject to} && \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{aligned}$$

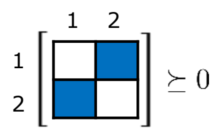
$$\Rightarrow$$

$$\begin{aligned} &\underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} && \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ &\text{subject to} && \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & && \mathbf{W} \succeq 0 \end{aligned}$$

Full-scale SDP:



Decomposed SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?
- The high-dim conic constraint can be decomposed to smaller order cones.

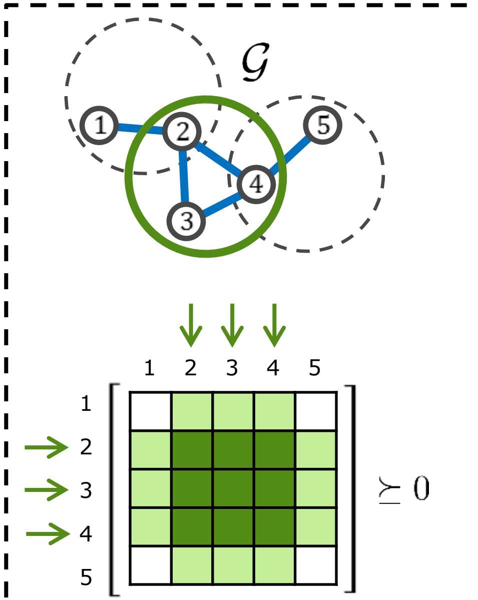
Decomposed SDP

Quadratic Problem:

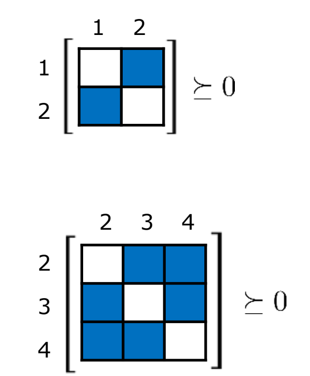
$$\begin{aligned} &\text{find} && \mathbf{v} \in \mathbb{C}^n \\ &\text{subject to} && \mathbf{v}^{\mathrm{H}} \mathbf{M}_k \mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{aligned}$$

$$\begin{aligned} &\underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} && \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ &\text{subject to} && \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & && \mathbf{W} \succeq 0 \end{aligned}$$

Full-scale SDP:



Decomposed SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?
- The high-dim conic constraint can be decomposed to smaller order cones.

Decomposed SDP

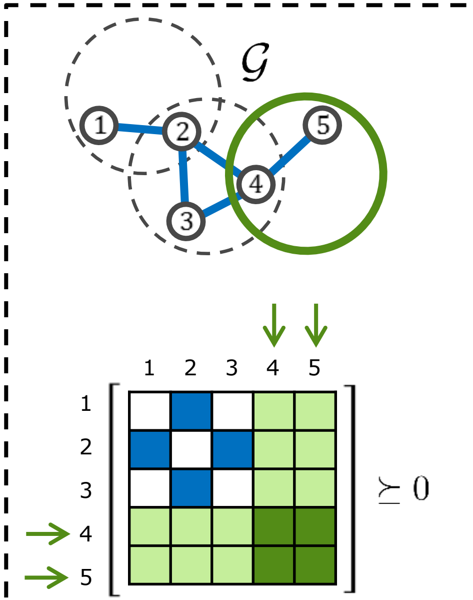
Quadratic Problem:

$$\begin{aligned} &\text{find} && \mathbf{v} \in \mathbb{C}^n \\ &\text{subject to} && \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{aligned}$$

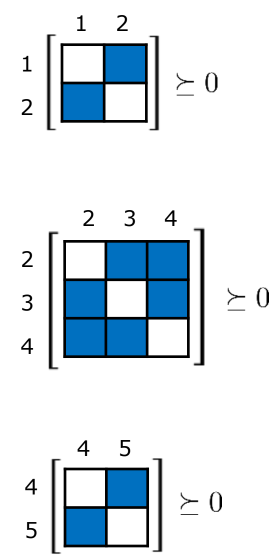
$$\Rightarrow$$

$$\begin{aligned} &\underset{\substack{\mathbf{W} \in \mathbb{H}_n \\ \boldsymbol{\nu} \in \mathbb{R}^m}}{\text{minimize}} && \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ &\text{subject to} && \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & && \mathbf{W} \succeq 0 \end{aligned}$$

Full-scale SDP:



Decomposed SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?
- The high-dim conic constraint can be decomposed to smaller order cones.

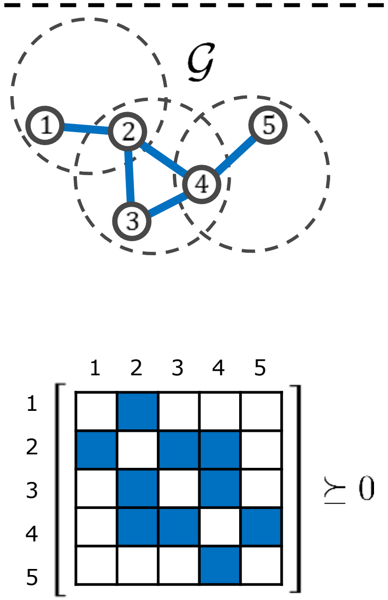
Decomposed SDP

Quadratic Problem:

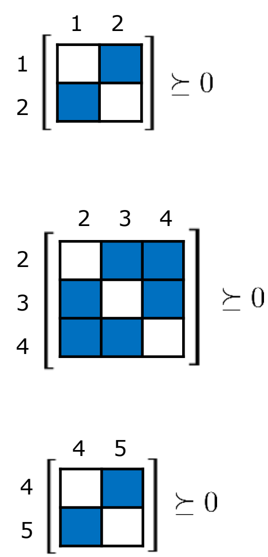
$$\begin{array}{ll} \text{find} & \mathbf{v} \in \mathbb{C}^n \\ \text{subject to} & \mathbf{v}^{\mathrm{H}}\mathbf{M}_k\mathbf{v} + \nu_k = y_k, \quad k = 1, \ldots, m \end{array}$$

$$\begin{array}{lll} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 & \\ \text{subject to} & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, & k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 & \end{array}$$

Full-scale SDP:



Decomposed SDP:



- Tree decomposition can be used for solving large scale SDPs [Fukuda et al., 01] and [Nakata et al., 03].
- How to solve a high-dim SDP?
- The high-dim conic constraint can be decomposed to smaller order cones.

Rank One Approximation

SDP relaxation:

$$\begin{aligned}
\underset{\substack{\mathbf{W}\in\mathbb{H}_n\\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\
\text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\
& \mathbf{W} \succeq 0
\end{aligned}$$

Rank One Approximation

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- When all measurements are noisy, the solution is never rank one.

Rank One Approximation

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- When all measurements are noisy, the solution is never rank one.
- How to find a rank one of approximate?

Rank One Approximation

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \quad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- When all measurements are noisy, the solution is never rank one.
- How to find a rank one of approximate?

Rank-1 approximation algorithm: Given an optimal solution $\mathbf{W}^{\text{opt}}$ of the SDP problem, we recover an approximate solution $\tilde{\mathbf{v}} \in \mathbb{C}^n$ as follows:

1. Set the voltage magnitude $|\tilde{v}_k| := \sqrt{W_{kk}^{\text{opt}}}$ for $k = 1, \ldots, n$.
2. Find the phases of the entries of $\tilde{\mathbf{v}}$ by solving the convex program:

$$\begin{aligned} \underset{\boldsymbol{\theta}\in[-\pi,\pi]^n}{\text{minimize}} \quad & \sum_{(i,j)\in\mathcal{L}} \left| \measuredangle W_{ij}^{\text{opt}} - \theta_i + \theta_j \right| \\ \text{subject to} \quad & \theta_o = 0, \end{aligned}$$

where $o \in \mathcal{N}$ is the slack bus.

Rank One Approximation

SDP relaxation:

$$\begin{aligned}\underset{\substack{\mathbf{W}\in\mathbb{H}_n\\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1,\ldots,m, \\ & \mathbf{W} \succeq 0\end{aligned}$$

- When all measurements are noisy, the solution is never rank one.
- How to find a rank one of approximate?

Rank-1 approximation algorithm: Given an optimal solution $\mathbf{W}^{\text{opt}}$ of the SDP problem, we recover an approximate solution $\tilde{\mathbf{v}} \in \mathbb{C}^n$ as follows:

1. Set the voltage magnitude $|\tilde{v}_k| := \sqrt{W_{kk}^{\text{opt}}}$ for $k = 1,\ldots,n$.
2. Find the phases of the entries of $\tilde{\mathbf{v}}$ by solving the convex program:

$$\begin{aligned}\underset{\boldsymbol{\theta}\in[-\pi,\pi]^n}{\text{minimize}} \quad & \sum_{(i,j)\in\mathcal{L}} \left|\measuredangle W_{ij}^{\text{opt}} - \theta_i + \theta_j\right| \\ \text{subject to} \quad & \theta_o = 0,\end{aligned}$$

where $o \in \mathcal{N}$ is the slack bus.

Rank One Approximation

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1,\ldots,m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- When all measurements are noisy, the solution is never rank one.
- How to find a rank one of approximate?

Rank-1 approximation algorithm: Given an optimal solution $\mathbf{W}^{\text{opt}}$ of the SDP problem, we recover an approximate solution $\tilde{\mathbf{v}} \in \mathbb{C}^n$ as follows:

1. Set the voltage magnitude $|\tilde{v}_k| := \sqrt{W_{kk}^{\text{opt}}}$ for $k = 1, \ldots, n$.
2. Find the phases of the entries of $\tilde{\mathbf{v}}$ by solving the convex program:

$$\begin{aligned} \underset{\boldsymbol{\theta}\in[-\pi,\pi]^n}{\text{minimize}} \quad & \sum_{(i,j)\in\mathcal{L}} \left|\measuredangle W_{ij}^{\text{opt}} - \theta_i + \theta_j\right| \\ \text{subject to} \quad & \theta_o = 0, \end{aligned}$$

where $o \in \mathcal{N}$ is the slack bus.

Zero Injection Buses

SDP relaxation:

$$
\begin{aligned}
\underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\
\text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\
& \mathbf{W} \succeq 0
\end{aligned}
$$

Zero Injection Buses

SDP relaxation:

$$
\begin{aligned}
\underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W}\rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\
\text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W}\rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\
& \mathbf{W} \succeq 0
\end{aligned}
$$

- In real world grids, there are many buses without any loads or generators.

Zero Injection Buses

SDP relaxation:

$$
\begin{aligned}
\underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\
\text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\
& \mathbf{W} \succeq 0
\end{aligned}
$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Zero Injection Buses

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Number of zero injection buses			
IEEE 30-bus	6	Polish 2746wop	711
New England 39	10	Polish 2746wp	705
IEEE 57-bus	15	Polish 3012wp	726
IEEE 118-bus	10	Polish 3120sp	798
IEEE 300-bus	65	Polish 3375wp	890
Polish 2383wp	552	PEGASE 1354	421
Polish 2736sp	645	PEGASE 2869	868
Polish 2737sop	689	PEGASE 9241	4346

Zero Injection Buses

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Kron reduction:

- We lose line measurements that are incident to the removed buses.
- We lose sparsity.

Number of zero injection buses			
IEEE 30-bus	6	Polish 2746wop	711
New England 39	10	Polish 2746wp	705
IEEE 57-bus	15	Polish 3012wp	726
IEEE 118-bus	10	Polish 3120sp	798
IEEE 300-bus	65	Polish 3375wp	890
Polish 2383wp	552	PEGASE 1354	421
Polish 2736sp	645	PEGASE 2869	868
Polish 2737sop	689	PEGASE 9241	4346

Zero Injection Buses

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Kron reduction:

- We lose line measurements that are incident to the removed buses.
- We lose sparsity.

Number of zero injection buses			
IEEE 30-bus	6	Polish 2746wop	711
New England 39	10	Polish 2746wp	705
IEEE 57-bus	15	Polish 3012wp	726
IEEE 118-bus	10	Polish 3120sp	798
IEEE 300-bus	65	Polish 3375wp	890
Polish 2383wp	552	PEGASE 1354	421
Polish 2736sp	645	PEGASE 2869	868
Polish 2737sop	689	PEGASE 9241	4346

$$\sum_{j\in\mathcal{N}} v_k^*(v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} (v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} v_l^*(v_k - v_j)y_{kj} = 0 \quad \forall l \in \mathcal{L}$$

Zero Injection Buses

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \dots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Kron reduction:

- We lose line measurements that are incident to the removed buses.
- We lose sparsity.

Number of zero injection buses			
IEEE 30-bus	6	Polish 2746wop	711
New England 39	10	Polish 2746wp	705
IEEE 57-bus	15	Polish 3012wp	726
IEEE 118-bus	10	Polish 3120sp	798
IEEE 300-bus	65	Polish 3375wp	890
Polish 2383wp	552	PEGASE 1354	421
Polish 2736sp	645	PEGASE 2869	868
Polish 2737sop	689	PEGASE 9241	4346

$$\sum_{j\in\mathcal{N}} v_k^*(v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} (v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} v_l^*(v_k - v_j)y_{kj} = 0 \quad \forall l \in \mathcal{L}$$

Each zero injection bus introduces $2n$ valid equalities.

Zero Injection Buses

SDP relaxation:

$$\begin{aligned} \underset{\substack{\mathbf{W}\in\mathbb{H}_n \\ \boldsymbol{\nu}\in\mathbb{R}^m}}{\text{minimize}} \quad & \langle \mathbf{M}, \mathbf{W} \rangle + \mu \times \|\boldsymbol{\nu}\|_1 \\ \text{subject to} \quad & \langle \mathbf{M}_k, \mathbf{W} \rangle + \nu_k = y_k, \qquad k = 1, \ldots, m, \\ & \mathbf{W} \succeq 0 \end{aligned}$$

- In real world grids, there are many buses without any loads or generators.
- These buses can be used to strengthen SDP.

Kron reduction:

- We lose line measurements that are incident to the removed buses.
- We lose sparsity.

Number of zero injection buses			
IEEE 30-bus	6	Polish 2746wop	711
New England 39	10	Polish 2746wp	705
IEEE 57-bus	15	Polish 3012wp	726
IEEE 118-bus	10	Polish 3120sp	798
IEEE 300-bus	65	Polish 3375wp	890
Polish 2383wp	552	PEGASE 1354	421
Polish 2736sp	645	PEGASE 2869	868
Polish 2737sop	689	PEGASE 9241	4346

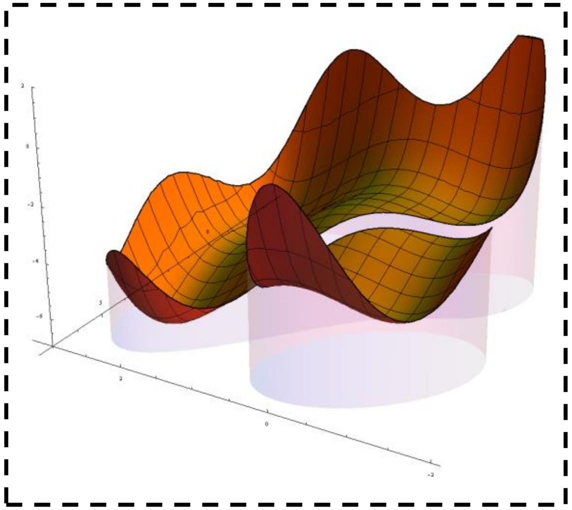
$$\sum_{j\in\mathcal{N}} v_k^*(v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} (v_k - v_j)y_{kj} = 0 \;\Rightarrow\; \sum_{j\in\mathcal{N}} v_l^*(v_k - v_j)y_{kj} = 0 \quad \forall l \in \mathcal{L}$$

Each zero injection bus introduces $2n$ valid equalities.

$$\sum_{j\in\mathcal{N}} (W_{lk} - W_{lj})y_{kj} = 0 \quad \forall l \in \mathcal{L}$$

Conclusions

- Penalized SDP relaxation:
  - Polynomial Optimization
  - Polynomial Feasibility
- Advantages:
  - Guaranteed rank one solution
  - Rejection of bad data
  - Robustness to noise
- Applications:
  - Optimal Power Flow
  - Power System State Estimation
  - Tensor completion



Thank you