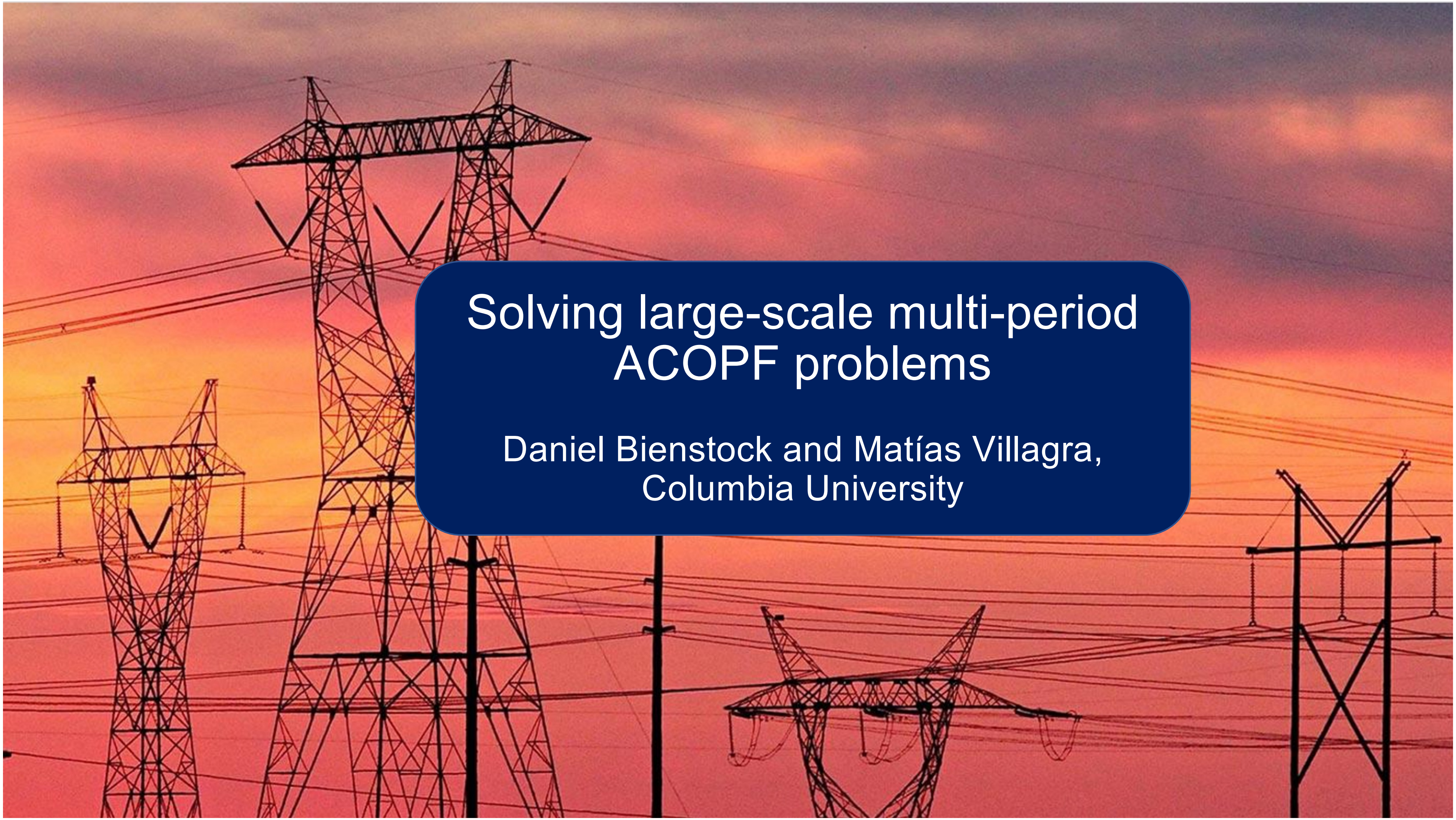


Solving large-scale multi-period ACOPF problems

Daniel Bienstock and Matías Villagra,
Columbia University

Preliminaries

All **code**, **model files**, **AMPL files**, **solution files** available from:
http://www.github.com/matias-vm

Paper (long version): https://arxiv.org/abs/2312.04251

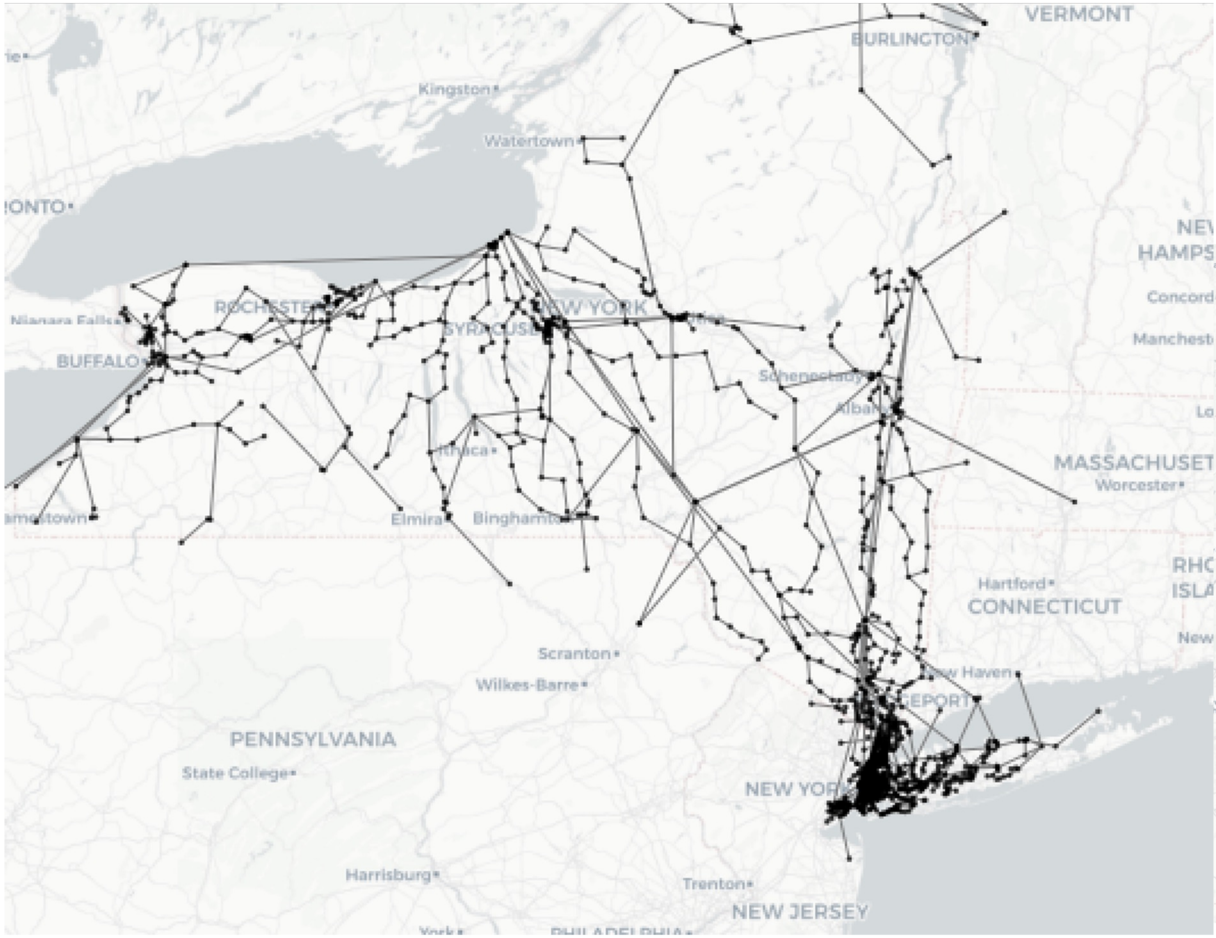
Realistic data?

Matpower: https://matpower.org/

PGLIB: arXiv:1908.02788v2

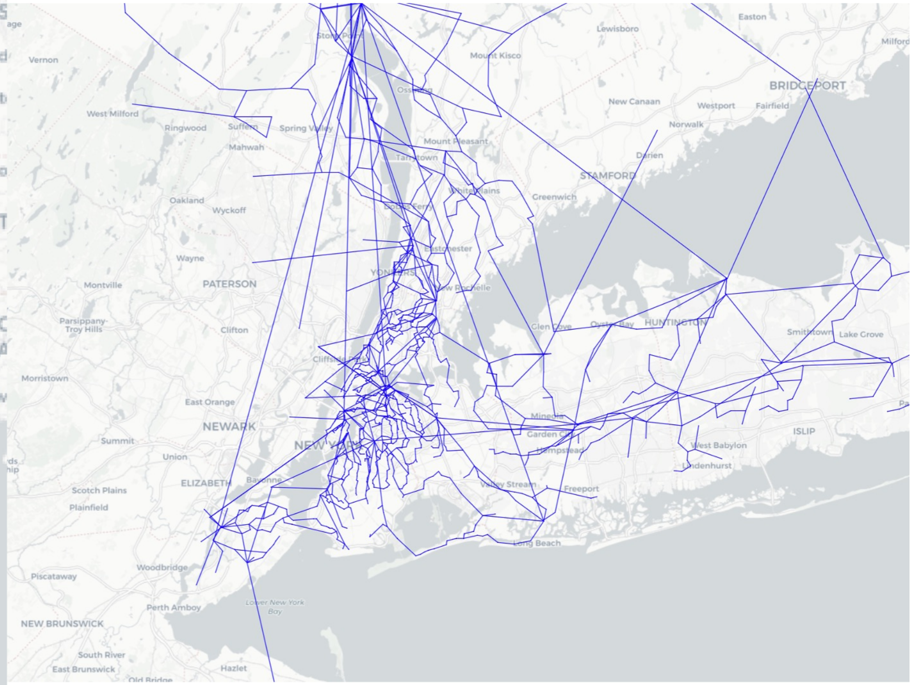
GO Competition: https://gocompetition.energy.gov/

A small example



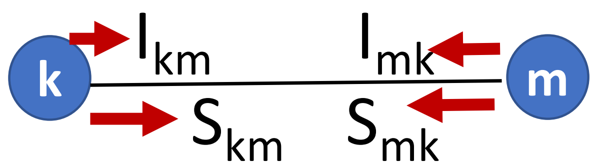
NY ISO system:
1814 buses
500+ generators
33 GW peak load

This is **not** a large system



AC Power basics:

$V_k = |V_k| e^{j\theta_k}$



$V_m = |V_m| e^{j\theta_m}$

$$Y = \begin{pmatrix} G_{kk} + jB_{kk} & G_{km} + jB_{km} \\ G_{mk} + jB_{mk} & G_{mm} + jB_{mm} \end{pmatrix}$$

Admittance matrix for line km

Ohm's Law

$$I_{km} = Y \begin{pmatrix} V_k \\ V_m \end{pmatrix}$$

Complex current

$$S_{km} = V_k I_{km}^*$$

complex power **injected** into **km** at **k**

$$S_{km} = V_k \left[Y \begin{pmatrix} V_k \\ V_m \end{pmatrix} \right]^*$$

S_{km} != $-S_{mk}$

$$S_{km} = (G_{kk} - jB_{kk})\, |V_k|^2 + (G_{km} - jB_{km})\, |V_k||V_m|(\cos\theta_{km} + j\sin\theta_{km})$$

$\theta_{km} \doteq \theta_k - \theta_m$

Single-period ACOPF

Minimize $\sum_{i \in \mathcal{G}} F_i(\, \boldsymbol{P_i^g})$

Convex function

Real power generated at generator i

with constraints:

Complex power injected into branch km at k :

$\forall$ branch km: $\boldsymbol{S_{km}} =$

$(G_{kk} - jB_{kk})\, |\boldsymbol{V_k}|^2 + (G_{km} - jB_{km})\, |\boldsymbol{V_k}||\boldsymbol{V_m}|(\cos \boldsymbol{\theta_{km}} + \boldsymbol{j} \sin \boldsymbol{\theta_{km}}$

Complex power flow balance at k:

$\sum_{km \in \delta(k)} \boldsymbol{S_{km}} = (\sum_{i \in \mathcal{G}(k)} \boldsymbol{P_i^g} - P_k^d) + j(\sum_{i \in \mathcal{G}(k)} \boldsymbol{Q_i^g} - Q_k^d)$

Total real power generated at bus **k**

Real power demand at **k**

Power flow limit on branch km:

$|\boldsymbol{S_{km}}|^2 = \mathcal{R}\mathrm{e}(\, \boldsymbol{S_{km}})^2 + Im(\, \boldsymbol{S_{km}})^2 \leq U_{km}$

Voltage limit on bus k:

$V_k^{\min} \leq |\boldsymbol{V_k}| \leq V_k^{\max}$

Generator output limits:

$P_i^{\min} \leq \boldsymbol{P_i^g} \leq P_i^{\max}$

HOW DO WE **SOLVE** ACOPF ??

This is a tale of **two unrelated** questions:

- Do we want **a really good feasible solution**?
- Do we want **a tight lower bound**?
- And … what is **feasible**?

First question: **finding a good solution.**

Only one answer: log barrier methods.

Knitro, Ipopt, home-cooked versions

Sample runtimes

Case	# buses	# branches	# generators	Runtime (sec)	Solver
118	118	186	54	0.76	MIPS
1354pegase	1354	1991	260	1.95	MIPS
ACTIVSg2000	2000	3206	544	2.96	MIPS
3120sp	3120	3693	505	4.25	MIPS
9241pegase	9241	16049	1445	11.78	MIPS
ACTIVSg70k	70000	88287	10390	177.83* *Apple M2	**Knitro**

Optimal? Near optimal? Any guarantees?

ACOPF as a **QCQP** (Quadratically Constrained Quadratic Program)

$V_k = |V_k| e^{j\theta_k}$

$V_m = |V_m| e^{j\theta_m}$

$\theta_{km} \doteq \theta_k - \theta_m$

$$\boldsymbol{S_{km}} = (G_{kk} - jB_{kk})\,|\boldsymbol{V_k}|^2 + (G_{km} - jB_{km})\,|\boldsymbol{V_k}||\boldsymbol{V_m}|(\cos\boldsymbol{\theta_{km}} + \boldsymbol{j}\sin\boldsymbol{\theta_{km}})$$

$$S_{km} = V_k \left[Y \begin{pmatrix} V_k \\ V_m \end{pmatrix} \right]^*$$

$$Y = \begin{pmatrix} G_{kk} + jB_{kk} & G_{km} + jB_{km} \\ G_{mk} + jB_{mk} & G_{mm} + jB_{mm} \end{pmatrix}$$

Admittance matrix for line km

$$V_i = e_i + jf_i$$

Use rectangular coordinates for voltages

ACOPF as a **QCQP**

$$S_{km} = (G_{kk} - jB_{kk})\,|V_k|^2 + (G_{km} - jB_{km})\,|V_k||V_m|(\cos\theta_{km} + j\sin\theta_{km})$$

$$V_i = e_i + j f_i$$

Use rectangular coordinates for voltages

Real ("active") part

$$P_{km} = G_{kk}(e_k^2 + f_k^2) + G_{km}(e_k e_m + f_k e_m) + B_{km}(-e_k f_m + f_k e_m)$$

Imaginary ("reactive") part

$$Q_{km} = -B_{kk}(e_k^2 + f_k^2) - B_{km}(e_k e_m + f_k e_m) + G_{km}(-e_k f_m + f_k e_m)$$

Minimize $\sum_{i\in\mathcal{G}} F_i(\ \boldsymbol{P_i^g})$

with constraints:

$\sum_{km\in\delta(k)}\ \boldsymbol{S_{km}}\ =\ (\sum_{i\in\mathcal{G}(k)} \boldsymbol{P_i^g} - P_k^d)\ +\ j(\sum_{i\in\mathcal{G}(k)} \boldsymbol{Q_i^g} - Q_k^d)$

$\forall$ branch km : $\boldsymbol{S_{km}} = \boldsymbol{P_{km}}\ +\ \boldsymbol{jQ_{km}}$

$$\boldsymbol{P_{km}} = G_{kk}\ \boldsymbol{v_k^{(2)}} + G_{km}\ \boldsymbol{c_{km}} + B_{km}\ \boldsymbol{s_{km}}$$

$$\boldsymbol{Q_{km}} = -B_{kk}\ \boldsymbol{v_k^{(2)}}\ - B_{km}\ \boldsymbol{c_{km}} + G_{km}\ \boldsymbol{s_{km}}$$

$$\rightarrow \boldsymbol{v_k^{(2)} = e_k^2 + f_k^2,\ c_{km} = e_k e_m + f_k f_m,\ s_{km} = -e_k f_m + f_k e_m}$$

$$\boldsymbol{P_{km}}^2 +\ \boldsymbol{Q_{km}}^2\ \leq\ U_{km}$$

$V_k^{\min} \leq \boldsymbol{v_k^{(2)}} \leq V_k^{\max}\quad \forall$ bus k

$P_k^{\min} \leq \boldsymbol{P_k^g} \leq P_k^{\max}\quad \forall$ generator k

How well does this work on ACOPF?

Case	Root relaxation	300 seconds	Log barrier (Knitro)	Interior point time (s)
9	2264.30	5301.40*	5296.69	0.24
30	0.00	154.08	576.89	0.47
118	0.00	0.00	129660.69	0.24
1354pegase	23037.69	23037.69	74069.35	2.45
ACTIVSg2000	649917.91	649917.91	1228892.08	3.01

(Gurobi 10 on QCQP)

*Why is the **lower bound** so bad?* ***Especially at the root??***

*How about **upper bounds** … using spatial branching?*

Key task: improve the lower bounds

An important observation

Replace nonlinearities with new variables:

$|V_k||V_m|\cos\theta_{km}$ ← c_{km}

$|V_k||V_m|\sin\theta_{km}$ ← s_{km}

$|V_k|^2$ ← $v_k^{(2)}$

$|V_m|^2$ ← $v_m^{(2)}$

$$\longrightarrow \quad c_{km}^2 + s_{km}^2 = v_k^{(2)} v_m^{(2)}$$

$$c_{km}^2 + s_{km}^2 \leq v_k^{(2)} v_m^{(2)}$$

Jabr inequality

Minimize $\sum_{i \in \mathcal{G}} F_i(\ \boldsymbol{P_i^g})$

with constraints:

$$\sum_{km \in \delta(k)} \boldsymbol{S_{km}} \ = \ (\sum_{i \in \mathcal{G}(k)} \boldsymbol{P_i^g} - P_k^d) + j(\sum_{i \in \mathcal{G}(k)} \boldsymbol{Q_i^g} - Q_k^d)$$

$\forall$ branch km : $\quad \boldsymbol{S_{km}} = \ \boldsymbol{P_{km}} \ + \ \boldsymbol{jQ_{km}}$

$$\boldsymbol{P_{km}} = G_{kk}\ \boldsymbol{v_k^{(2)}} + G_{km}\ \boldsymbol{c_{km}} + B_{km}\ \boldsymbol{s_{km}}$$

$$\boldsymbol{Q_{km}} = -B_{kk}\ \boldsymbol{v_k^{(2)}} \ -B_{km}\ \boldsymbol{c_{km}} + G_{km}\ \boldsymbol{s_{km}}$$

$$\rightarrow \boldsymbol{v_k^{(2)} = e_k^2 + f_k^2,\ c_{km} = e_k e_m + f_k f_m,\ s_{km} = -e_k f_m + f_k e_m}$$

$$\boldsymbol{P_{km}}^2 + \ \boldsymbol{Q_{km}}^2 \ \leq \ U_{km}$$

$V_k^{\min} \leq \boldsymbol{v_k^{(2)}} \leq V_k^{\max} \quad \forall$ bus k

$P_k^{\min} \leq \boldsymbol{P_k^g} \leq P_k^{\max} \quad \forall$ generator k

Minimize $\sum_{i \in \mathcal{G}} F_i(\boldsymbol{P_i^g})$

with constraints:

$$\sum_{km \in \delta(k)} \boldsymbol{S_{km}} = (\sum_{i \in \mathcal{G}(k)} \boldsymbol{P_i^g} - P_k^d) + j(\sum_{i \in \mathcal{G}(k)} \boldsymbol{Q_i^g} - Q_k^d)$$

$\forall$ branch km : $\boldsymbol{S_{km}} = \boldsymbol{P_{km}} + \boldsymbol{jQ_{km}}$

$$\boldsymbol{P_{km}} = G_{kk}\, \boldsymbol{v_k^{(2)}} + G_{km}\, \boldsymbol{c_{km}} + B_{km}\, \boldsymbol{s_{km}}$$

$$\boldsymbol{Q_{km}} = -B_{kk}\, \boldsymbol{v_k^{(2)}} - B_{km}\, \boldsymbol{c_{km}} + G_{km}\, \boldsymbol{s_{km}}$$

$$\rightarrow \boldsymbol{c_{km}^2} + \boldsymbol{s_{km}^2} \leq \boldsymbol{v_k^{(2)}} \boldsymbol{v_m^{(2)}}$$

$$\boldsymbol{P_{km}}^2 + \boldsymbol{Q_{km}}^2 \leq U_{km}$$

$V_k^{\min} \leq \boldsymbol{v_k^{(2)}} \leq V_k^{\max}$ $\forall$ bus k

$P_i^{\min} \leq \boldsymbol{P_i^g} \leq P_i^{\max}$ $\forall$ generator i

20-core Xeon

SOLVERS' PERFORMANCE ON JABR SOCP

Case	Objective			Time (s)		
	Gurobi	Knitro	Mosek	Gurobi	Knitro	Mosek
9241pegase	-	309234.16	-	82.11	34.68	31.11
9241pegase-api	-	6840612.84	-	116.32	23.39	72.29
9241pegase-sad	-	6083747.85	-	111.05	26.01	75.99
9591goc-api	1346480.71	1348107.89	1345869.72	38.25	23.74	36.60
9591goc-sad	1055698.54	1058606.56	1054379.58	49.29	32.83	37.61
ACTIVSg10k	-	2468172.93	2466666.10	40.18	21.48	26.08
10000goc-api	-	2507034.94	2498948.00	48.63	35.19	30.13
10000goc-sad	1387288.49	1388679.63	1386041.07	23.58	26.27	23.68
10192epigrids-api	-	1849684.14	1848873.47	75.82	42.69	29.09
10192epigrids-sad	-	1672989.96	1672534.72	83.85	28.33	28.63
10480goc-api	-	2708973.58	2707828.26	75.94	27.21	56.82
10480goc-sad	-	2286454.3	2285547.23	149.93	38.17	59.48
13659pegase	379135.73	379144.11	-	33.61	43.26	34.92
13659pegase-api	-	9198542.14	-	162.21	30.64	105.11
13659pegase-sad	8826902.31	8826958.23	8787429.86	83.75	31.84	108.74
19402goc-api	-	2449020.25	2447799.72	158.12	152.89	103.04
19402goc-sad	-	1954331.70	1952550.06	203.56	155.89	104.88
20758epigrids-api	-	-	3040421.02	143.99	TLim	93.46
20758epigrids-sad	-	-	2610196.94	98.30	TLim	75.88
24464goc-api	2548335.96	-	2558631.63	603.95	TLim	129.90
24464goc-sad	-	-	2603525.46	333.50	TLim	128.50
ACTIVSg25k	5956787.54	5964417.54	5955368.56	169.66	87.14	87.18
30000goc-api	-	1531256.65	1529197.81	207.60	118.80	123.38
30000goc-sad	-	-	1130868.71	191.22	TLim	84.90
ACTIVSg70k	-	16221577.73	16217263.66	553.26	320.98	232.47
78484epigrids-api	-	-	-	756.00	TLim	637.48
78484epigrids-sad	15180775.21	-	15169401.54	463.17	TLim	601.04

Primal bound	
Objective	Time
315911.56	96.74
7068721.98	73.85
6318468.57	33.92
1570263.74	42.85
1167400.79	28.15
2485898.75	76.71
2678659.51	23.46
1490209.66	103.06
1977687.11	117.15
1720194.13	23.74
2863484.4	38.71
2314712.14	27.93
386108.81	1184.15
9385711.45	44.43
9042198.49	42.08
2583627.35	87.33
1983807.59	64.01
3126508.3	61.39
2638200.23	58.11
2683961.9	533.03
2653957.66	73.87
6017830.61	56.69
1777930.63	134.71
1317280.55	565.05
16439499.83	240.55
16140427.68	1079.03
15315885.86	343.45

Why are the bounds so good?

Why are SOCPs so difficult?

Experiment:

100 rounds of

1. Select **one** Jabr constraint **at random**
2. Remove it
3. Solve remaining SOC

Value averaged over 100 rounds

Case	[redacted]	Jabr	weak Jabr
case14		8075.12	6292.78
case118		129340	126982.72
case300		718654	714858.26

Why?

$$S_{km} = (G_{kk} - jB_{kk})\,|V_k|^2 + (G_{km} - jB_{km})\,|V_k||V_m|(\cos\theta_{km} + j\sin\theta_{km})$$

Complex power flow definition

$$P_{km} = G_{kk}\,|V_k|^2 + G_{km}\,|V_k||V_m|\cos\theta_{km} + B_{km}\,|V_k||V_m|\sin\theta_{km}$$

Real (active) component

$$P_{km} + P_{mk} \geq 0$$

Loss inequality:

(Physics)

B. and Munoz (2015): use **this**

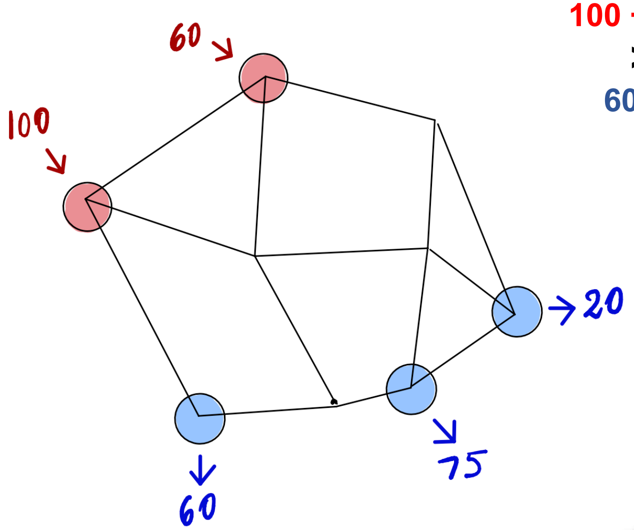
Not **that**

$$c_{km}^2 + s_{km}^2 \leq v_k^{(2)} v_m^{(2)}$$

Linearly constrained relaxation yields good bounds!

Lemma: loss inequality implies source-destination flow paths.
Every unit of load is accounted for by a unit of generation.

Why are losses important?

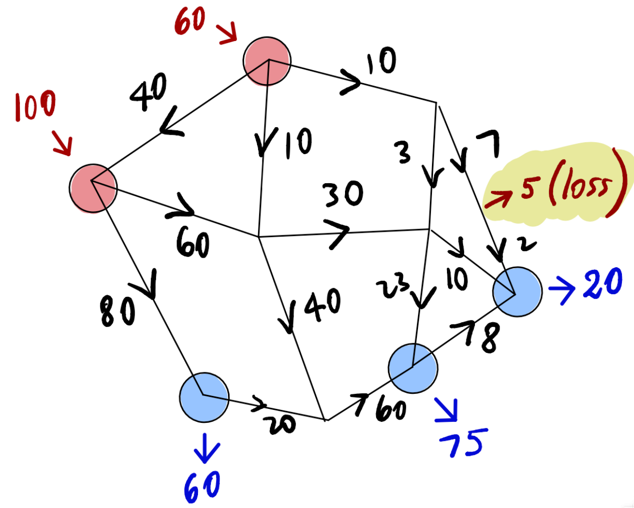


100 + 60 (generation)

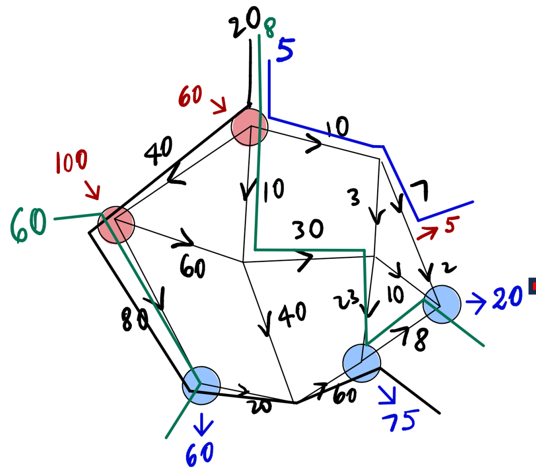
>

60 + 75 + 20 = 155

(load)



A partial flow decomposition:



Each unit of load and loss is accounted for by a corresponding unit of generation

Applying the loss inequality forces the relaxation to use more generation.

Recall that all costs in ACOPF are due to generation.

$$S_{km} = (G_{kk} - jB_{kk})\,|V_k|^2 + (G_{km} - jB_{km})\,|V_k||V_m|(\cos\theta_{km} + j\sin\theta_{km})$$

$$G_{kk} > 0 > G_{km} = G_{mk} \geq -G_{kk}, \quad B_{km} = B_{mk}$$

$$P_{km} = G_{kk}\,|V_k|^2 + G_{km}\,|V_k||V_m|\cos\theta_{km} + B_{km}\,|V_k||V_m|\sin\theta_{km}$$

$$P_{km} = G_{kk}\,v_k^{(2)} + G_{km}\,c_{km} + B_{km}\,s_{km}$$

$$P_{mk} = G_{mm}\,v_m^{(2)} + G_{mk}\,c_{km} - B_{km}\,s_{km}$$

$$P_{km} + P_{mk} = G_{kk}\,v_k^{(2)} + G_{mm}\,v_m^{(2)} + 2G_{km}\,c_{km} \geq \min\{G_{kk}, G_{mm}\}(\,v_k^{(2)} + v_m^{(2)} - 2c_{km})$$

$$c_{km}^2 + s_{km}^2 \leq v_k^{(2)} v_m^{(2)}$$

Jabr

$$v_k^{(2)} + v_m^{(2)} - 2c_{km} \geq 0$$

Outer-envelope approx. to Jabr

implies

$$P_{km} + P_{mk} \geq 0$$

Experiment:

100 rounds of

1. Select one Jabr constraint at random
2. Remove it
3. Solve remaining SOC

Value averaged over 100 rounds

	Formulation Value		Losses in Experiment			Formulation Value	
Case	AC	SOC	Avg Loss	Avg (br)	Min (br)	Jabr	weak Jabr
case14	0.0929	0.0918	-0.3808	-0.4906	-1.7443	8075.12	6292.78
case118	0.7740	0.7125	0.1084	-0.7046	-5.1803	129340	126982.72
case300	3.0274	2.8064	1.8485	-1.1652	-6.1421	718654	714858.26

Experiment:

100 rounds of

1. Select **100 Jabr constraints at random**
2. **Remove** them
3. **Solve** remaining SOC

		Weak Jabr				Jabr		AC
		Avg Losses		Min Losses				
Case	Load	Total	Sample	Sample	Objective	Objective	Losses	Losses
1354pegase	730.60	-150.38	-379.60	-764.52	58021.55	74008.58	9.49	10.09
2869pegase	1324.37	-197.83	-402.59	-1119.85	112663.19	133872.69	14.24	15.51
3375wp	483.63	-65.50	-117.11	-343.72	6467352.82	7385372.70	6.90	8.30

Basic Physics: $|S_{km}|^2 = |V_k|^2 |I_{km}|^2$ $\qquad$ $P_{km}^2 + Q_{km}^2 = v_k^{(2)} i_{km}^{(2)}$

$$i_{km}^2 \doteq \boldsymbol{\alpha_{km}}\, v_k^{(2)} + \boldsymbol{\beta_{km}}\, v_m^{(2)} + \boldsymbol{\gamma_{km}}\, c_{km} + \boldsymbol{\zeta_{km}}\, s_{km}$$

Data-dependent constants

$$P_{km}^2 + Q_{km}^2 \leq v_k^{(2)} i_{km}^{(2)}$$

i2 inequality

Basic Physics:

$$|S_{km}|^2 = |V_k|^2 |I_{km}|^2 \qquad P_{km}^2 + Q_{km}^2 = v_k^{(2)} i_{km}^{(2)}$$

$$i_{km}^2 = \boldsymbol{\alpha_{km}}\, v_k^{(2)} + \boldsymbol{\beta_{km}}\, v_m^{(2)} + \boldsymbol{\gamma_{km}}\, c_{km} + \boldsymbol{\zeta_{km}}\, s_{km}$$

$$P_{km}^2 + Q_{km}^2 \leq v_k^{(2)} i_{km}^{(2)} \qquad i_{km}^2 \leq \frac{|\boldsymbol{S_{km}}|^{\mathbf{2}}_{\max}}{|\boldsymbol{V_k}|^{\mathbf{2}}_{\min}}$$

If $\boldsymbol{\alpha_{km}}$ is *large*: use

$$v_k^{(2)} + \frac{\boldsymbol{\beta_{km}}}{\boldsymbol{\alpha_{km}}} v_m^{(2)} + \frac{\boldsymbol{\gamma_{km}}}{\boldsymbol{\alpha_{km}}} c_{km} + \frac{\boldsymbol{\zeta_{km}}}{\boldsymbol{\alpha_{km}}} s_{km} \geq 0,$$

$$v_k^{(2)} + \frac{\boldsymbol{\beta_{km}}}{\boldsymbol{\alpha_{km}}} v_m^{(2)} + \frac{\boldsymbol{\gamma_{km}}}{\boldsymbol{\alpha_{km}}} c_{km} + \frac{\boldsymbol{\zeta_{km}}}{\boldsymbol{\alpha_{km}}} s_{km} \leq \frac{|\boldsymbol{S_{km}}|^{\mathbf{2}}_{\max}}{|\boldsymbol{V_k}|^{\mathbf{2}}_{\min}}$$

Else: explicitly use

$$i_{km}^2 = \boldsymbol{\alpha_{km}}\, v_k^{(2)} + \boldsymbol{\beta_{km}}\, v_m^{(2)} + \boldsymbol{\gamma_{km}}\, c_{km} + \boldsymbol{\zeta_{km}}\, s_{km}$$

$$P_{km}^2 + Q_{km}^2 \leq v_k^{(2)} i_{km}^{(2)}$$

Large SOCP relaxations are **difficult** for our solvers. Why?

IN ACOPF:

- Many of the Jabr (SOC) constraints are tight at optimum – they are **needed.**
- Why? If not there, relaxation inaccurate.
- Loss inequalities, which are outer-approximation of the Jabr constraints, yield good bounds.

Our solution: a cutting-plane algorithm

- **Outer-approximation** of Jabr inequalities (**and i2 SOC inequalities**)
- **Cut management:**
  - **Remove** old cuts that are slack
  - **Reject** new cuts if too parallel to existing cuts.

Uses Gurobi → Cutting-Plane Algorithm

Knitro → Primal bound

Case	Cutting-Plane Algorithm					Primal bound	
	Objective	Time	Computed	Added	Rounds	Objective	Time
9241pegase	309221.81	378.82	135599	29875	23	315911.56	96.74
9241pegase-api	6924650.57	277.32	128316	30230	21	7068721.98	73.85
9241pegase-sad	6141202.28	386.51	113686	27273	21	6318468.57	33.92
9591goc-api	1346373.10	187.26	87812	22469	22	1570263.74	42.85
9591goc-sad	1055493.25	246.87	90153	20514	27	1167400.79	28.15
ACTIVSg10k	2476851.62	132.16	60803	18183	19	2485898.75	76.71
10000goc-api	2502026.03	147.12	73084	19666	24	2678659.51	23.46
10000goc-sad	1387303.02	114.97	58984	18528	17	1490209.66	103.06
10192epigrids-api	1849488.30	152.87	97921	24882	22	1977687.11	117.15
10192epigrids-sad	1672819.53	185.02	95740	23726	23	1720194.13	23.74
10480goc-api	2708819.18	200.48	114967	29805	21	2863484.4	38.71
10480goc-sad	2287314.69	270.38	118122	28004	24	2314712.14	27.93
13659pegase	379084.55	841.83	176962	37297	22	386108.81	1184.15
13659pegase-api	9270988.77	326.57	147479	34390	19	9385711.45	44.43
13659pegase-sad	8868216.24	301.87	130682	32662	19	9042198.49	42.08
19402goc-api	2448812.41	440.67	213564	52388	22	2583627.35	87.33
19402goc-sad	1954047.79	488.33	218291	49749	25	1983807.59	64.01
20758epigrids-api	3042956.88	464.17	189436	46124	25	3126508.3	61.39
20758epigrids-sad	2612551.03	379.36	180790	44624	24	2638200.23	58.11
24464goc-api	2560407.12	471.14	226595	57162	22	2683961.9	533.03
24464goc-sad	2605128.51	506.39	222908	55242	23	2653957.66	73.87
ACTIVSg25k	5993266.85	592.39	156285	43851	28	6017830.61	56.69
30000goc-api	1531110.84	464.16	142385	41840	24	1777930.63	134.71
30000goc-sad	1130733.51*	147.74	76546	76546	6	1317280.55	565.05
ACTIVSg70k	16326225.66	1065.76	350572	123431	13	16439499.83	240.55
78484epigrids-api	15877674.54	1007.99	556893	240576	10	16140427.68	1079.03
78484epigrids-sad	15175077.19	1062.55	501202	313587	8	15315885.86	343.45

20-core Xeon, recent

Jabr SOCP statistics:

```
Presolved: 1407109 rows, 882912 columns, 3013953 nonzeros
Presolved model has 216224 second-order cone constraints
```

	SOCP value			**SOCP time**		
	Gurobi	Knitro	Mosek	Gurobi	Knitro	Mosek
Jabr SOCP (ACTIVSg70k)	-	16221577.73	16217263.66	553.26	320.98	232.47
i2 SOCP	16333807.38†	-	-	300.98	TLim	209.3

An essential feature of power systems operation

ACOPF problems, in the real-world **are ~~never~~ rarely run "from scratch".**
The power grid never stops …
Ideally, a new computation every 5 minutes.

There is a **'prior'** solution – ACOPF **was recently run on similar data.**

Most data sets include information about this prior solution.

How can we take advantage of this information?

It seems difficult to appropriately warm-start SOCP solvers.

Our linear cutting-plane algorithm can take advantage of this mode of operation.

We assume that the cuts from the prior run are available.

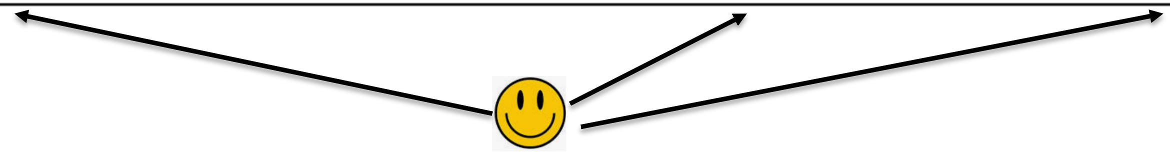
They remain **valid** most of the time. Why?

We use those cuts as a starting formulation.

Our favorite solvers are designed to handle this functionality well.

Warm-Started Relaxations, Loads perturbed Gaussian $(\mu, \sigma) = (0.01 \cdot P_d, 0.01 \cdot P_d)$

	Cutting-Plane				Jabr SOCP							
	First Round		Last Round		Objective			Time (s)			Primal bound	
Case	Objective	Time (s)	Objective	Time (s)	Gurobi	Knitro	Mosek	Gurobi	Knitro	Mosek	Objective	Time (s)
9241pegase	309288.32	13.78	309299.97	160.28	-	309302.67	-	73.12	32.21	36.04	315979.53	101.48
9241pegase-api	INF	23.10	INF	23.10	-	-	-	134.53	TLim	72.96	LOC INF	1845.92
9241pegase-sad	6153913.91	16.18	6154117.59	136.78	-	6096743.03	-	97.51	26.07	83.43	6333763.92	43.71
9591goc-api	1343642.47	11.06	1343670.62	56.36	1343767.43	1345384.57	1343190.29	39.36	25.36	35.30	1571582.59	54.16
9591goc-sad	1058124.48	12.62	1058157.44	65.37	1058337.76	1061275.83	1057323.31	51.85	34.04	37.52	1178895.53	29.53
ACTIVSg10k	2475041.43	9.52	2475078.69	50.51	-	2466383.20	-	42.31	21.75	29.33	2484093.15	57.24
10000goc-api	2502049.28	8.51	2502098.01	36.48	2501946.30	2507074.78	2499373.75	31.91	43.44	32.33	LOC INF	1677.21
10000goc-sad	1388833.86	8.70	1388859.09	44.50	1388824.91	1390230.41	1387588.17	25.96	29.31	23.67	1493481.44	93.72
10192epigrids-api	1848085.36	10.27	1848133.48	45.84	-	1848285.26	1847120.93	65.38	41.17	25.99	LOC INF	1458.35
10192epigrids-sad	1672358.89	10.33	1672398.61	53.37	-	1672533.02	1671364.67	73.64	28.61	35.66	1717429.36	23.89
10480goc-api	2704157.29	12.43	2704252.95	58.45	-	2704373.73	2703432.85	197.17	27.57	55.92	2868495.28	36.89
10480goc-sad	2294908.37	12.81	2294990.69	70.93	-	2294080.35	2292830.56	185.22	35.90	58.31	2322198.81	27.34
13659pegase	379742.62	60.74	379794.51	426.88	379799.37	379804.43	-	34.21	43.17	32.75	386765.25	370.23
13659pegase-api	9253539.07	21.25	9253773.43	109.20	9181205.93	9181269.20	-	97.11	30.41	118.31	9368277.57	62.20
13659pegase-sad	8865733.59	21.28	8865892.49	113.04	8824442.20	8824486.03	-	86.49	33.19	102.59	9039904.52	40.02
19402goc-api	2452185.69	23.55	2452270.83	120.10	-	2452448.33	2451708.50	146.87	120.39	103.32	LOC INF	4440.99
19402goc-sad	1956255.19	23.28	1956313.91	113.89	-	1956570.60	1955018.07	231.90	172.82	102.19	1986936.95	66.02
20758epigrids-api	3043006.76	22.34	3043076.56	104.06	-	-	3032919.24	134.60	TLim	78.32	LOC INF	12425.89
20758epigrids-sad	2610197.53	20.46	2610261.88	93.09	-	-	2608090.26	143.69	TLim	72.19	2635892.81	49.25
24464goc-api	2561680.14	26.28	INF	50.38	-	LOC INF	-	223.07	573.37	118.6	-	19444.54
24464goc-sad	2606391.76	26.78	2606473.78	133.54	-	-	2604708.86	423.12	TLim	128.84	2655942.01	72.48
ACTIVSg25k	5988886.18	28.24	5989016.75	198.58	5952404.50	5960068.30	5949381.04	138.01	73.75	109.39	6013477.05	57.87
30000goc-api	1527412.96	25.35	1527487.45	151.75	-	1528338.73	1525625.64	243.61	369.83	119.92	LOC INF	3407.47
30000goc-sad	-	46.33	-	46.33	-	-	1132715.53	257.94	TLim	75.20	1318389.55	620.27
ACTIVSg70k	16316572.42	102.25	16317886.35	536.51	-	16210682.53	16206290.43	498.80	309.56	229.07	16428367.50	243.84
78484epigrids-api	15862318.24	115.76	15865624.98	883.93	-	-	15859950.52	757.64	TLim	642.24	-	8113.53
78484epigrids-sad	15176866.00	151.77	15180592.27	1118.02	15182602.75	-	15174716.43	420.56	TLim	589.46	15316872.94	353.13



Multi-period cases

- T time periods
- Loads are given for each period
- Generator ramping rates link successive periods

Solution strategy

- Use single-period cuts to warm-start
- Cuts propagated to all periods
- Relaxation sees all periods at once
- Primal Heuristic for nonconvex AC

Warm-Started Relaxation versus i2 SOCP+, $T = 4$ and loads perturbed $\mathrm{Unif}(0, 0.025) + 0.01 \cdot t$ for $t \in \{1, 2, 3\}$, ramping rates 50%

	Cutting-plane algorithm						Solvers								
							i2 SOCP+								ACOPF
				First Iteration					Obj			Time			
Case	#Vars	#Cons	FTime	Obj	Time	#Cuts	#Vars	#Cons	Gurobi	Knitro	Mosek	Gurobi	Knitro	Mosek	PBound
1354pegase	66897	72564	1.41	301493.85	3.42	6000	56326	84155	303424.53	-	-	41.94	TLim	7.86	303727.82
3375wp	148217	159692	3.17	30132477.23	8.76	10448	121822	179665	-	-	-	60.00	TLim	15.28	30410712.08
6468rte	319517	323396	6.81	352645.45	18.24	32960	269070	373246	355828.72	-	-	468.73	TLim	41.89	356969.89
9241pegase	524028	538308	11.36	1258422.42	48.99	68472	451546	643811	-	-	-	411.55	TLim	92.33	1297205.96
ACTIVSg10k	456684	490360	10.27	10117471.23	30.65	28184	364824	437972	-	-	-	243.69	TLim	79.58	10265810.55
10000goc-api	465676	495636	9.88	9129427.52	28.01	35856	390042	567215	-	-	-	559.76	TLim	148.77	10019708.68
10000goc-sad	465676	495636	10.21	5414194.06	30.94	35224	390042	567215	-	-	-	419.79	TLim	104.86	5866821.40
13659pegase	723245	735740	15.67	-	365.52	97036	618066	859791	-	-	-	457.45	TLim	117.18	1584844.81
13659pegase-api	723245	735740	15.28	35162915.42	74.46	85584	618066	859791	-	-	-	1093.04	TLim	155.84	36163923.41
13659pegase-sad	723245	735740	15.60	34093210.18	52.16	78612	618066	859791	-	-	-	891.76	TLim	150.4	35199898.78
20758epigrids-api	1106271	1127168	23.34	-	195.65	80904	950270	1340191	-	-	-	1529.93	TLim	356.82	12088587.29
20758epigrids-sad	1106271	1127168	23.64	10138699.01	92.13	77228	950270	1340191	-	-	-	1397.93	TLim	247.43	10317105.84
ACTIVSg25k	1145783	1212392	25.46	24310520.83	93.95	68676	961854	1387817	-	-	-	589.52	TLim	200.22	24835784.33
30000goc-api	1253699	1366264	30.24	5185673.79	97.36	59732	1018714	1524129	-	-	-	1761.14	TLim	286.2	6563873.25
30000goc-sad	1253699	1366264	28.03	4316717.55	138.08	179704	1018714	1524129	-	-	-	727.32	TLim	247.6	-
ACTIVSg70k	3115935	3316972	68.7	67721489.41	350.58	258228	2584042	3765747	-	-	-	1312.16	TLim	405.88	70148300.24
78484epigrids-api	4179320	4225032	96.42	60947456.94	494.72	625780	3558322	4997073	-	-	-	TLim	TLim	1117.23	62011780.96
78484epigrids-sad	4179320	4225032	97.61	58930587.37	743.6	888916	3558322	4997073	-	-	-	TLim	TLim	1117.35	-

Warm-Started Relaxation versus i2 SOCP+, $T = 24$ hr active power load curve, ramping rates 50%

Case	#Vars	#Cons	FTime	First Iteration Obj	Time	#Cuts	DInfs
1354pegase	402677	440584	8.66	1656321.34	20.38	36000	2.46e-08
9241pegase	3151388	3258748	66.05	6865741.86	368.23	410832	4.33e-08
ACTIVSg10k	2752524	2991860	66.13	54565119.63	226.82	169104	7.5e-09
10000goc-api	2804496	3015596	62.15	51667906.99	208.71	215136	5.822e-07
10000goc-sad	2804496	3015596	60.63	32512183.58	284.34	211344	3.46e-06
ACTIVSg25k	6898863	7371032	150.16	132201219.46	816.18	412056	4.74e-07
30000goc-api	7539819	8268104	167.43	28586461.36	755.74	358392	1.7313e-06
30000goc-sad	7539819	8268104	169.75	25717079.58	1091.27	1078224	2.52e-06
ACTIVSg70k	18747555	20109632	410.49	350310736.78	2641.18	1549368	2.407e-07
78484epigrids-api	25110280	24855336	533.07	-	1521.64	3754680	-
78484epigrids-sad	25110280	24855336	532.93	344805634.00	4749.24	5333496	1.34e-04

Well …

- All solvers fail on every SOCP
- Our AC heuristic also fails

Warm-Started Relaxation versus DCOPF, $T = 12$ and $T = 24$

	$T = 12$										$T = 24$									
	Warm-Started Relaxation					DCOPF					Warm-Started Relaxation					DCOPF				
Case	#Vars	#Cons	Obj	Time	DInfs	#Vars	#Cons	Obj	Time	Dinfs	#Vars	#Cons	Obj	Time	DInfs	#Vars	#Cons	Obj	Time	Dinfs
1354pegase	201209	250008	860664.27	92.63	1.06e-08	62369	67828	850023.48	2.44	1.72e-10	402677	440584	1666609.97	180.61	2.97e-08	124997	136696	1646661.83	4.61	2.15e-10
9241pegase	1574972	1980693	3565252.49	3811.42	4.32e-04	447608	477952	3627501.88	18.06	3.76e-10	3151388	3258748	6925029.02	8419.66	1.25e-03	896660	961684	7013508.62	36.02	5.25e-10
ACTIVSg10k	1375020	1639853	28690135.28	1308.80	3.38e-07	449628	501812	28187885.77	19.50	2.34e-07	2752524	2991860	55184650.09	2869.25	7.18e-08	901740	1013564	54241448.25	37.77	5.44e-08
10000goc-api	1401204	1686928	28459989.11	1066.39	2.40e-08	446364	490232	INF	20.15	-	2804496	3015596	53800966.46	2273.58	3.18e-07	894816	988820	INF	40.25	-
10000goc-sad	1401204	1647684	16474320.51	1450.12	3.36e-06	446364	490232	16114986.72	19.96	2.21e-07	2804496	3015596	32678842.78	3099.33	7.97e-06	894816	988820	32120062.03	41.02	1.97e-06
ACTIVSg25k	3447015	4472632	69534077.06	4073.97	9.75e-07	1097943	1199456	67992943.20	48.82	7.88e-07	6898863	7371032	134142620.00	6991.16	2.08e-06	2200719	2418248	131325272.36	95.18	5.69e-09
30000goc-api	3768147	4484763	17079448.81	3667.40	1.06e-05	1225815	1299860	-	75.87	-	7539819	8268104	31616476.13	6799.92	2.06e-05	2455155	2613824	-	171.42	-
30000goc-sad	3768147	4373646	13190466.44	3901.76	2.45e-06	1225815	1299860	12846216.92	53.33	7.65e-08	7539819	8268104	25811075.96	6249.53	1.23e-05	2455155	2613824	25279287.29	108.86	2.22e-08
ACTIVSg70k	9368583	11913750	188736994.5	4702.30	3.66e-08	2977455	3195644	-	175.35	-	18747555	20109632	357083446.24	8496.71	2.98e-08	5965299	6432848	-	829.66	-
78484epigrids-api	12551704	14607420	INF	447.39	-	3555448	3699780	INF	4.19	-	25110280	24855336	-	2138.78	-	7117768	7427052	INF	380.67	-
78484epigrids-sad	12551704	16185978	177335668.05	5465.67	1.36e-04	3555448	3699780	-	175.91	-	25110280	30188832	344805634.00	5399.62	1.34e-04	7117768	7427052	-	513.08	-