Practical Solution Approaches to Optimization and Engineering: Case Studies in Mine Planning and Electrical Power

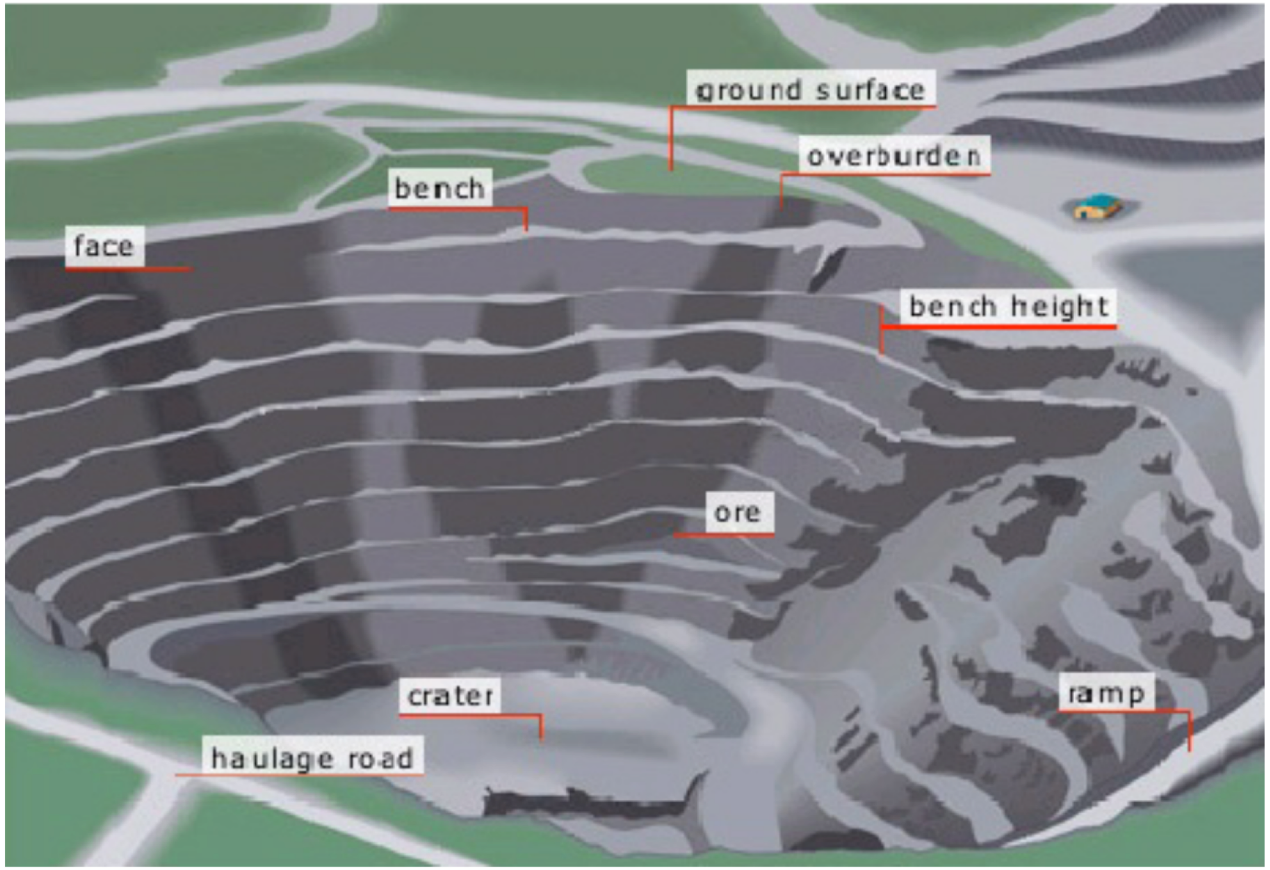
Daniel Bienstock, Columbia U. and Gurobi

RMIC, November 2022

Practical Optimization at a Crossroads

- Current and **past** areas of interest: logistics, transportation, supply chain
- These areas will remain relevant, but …
- The **future**: heavy engineering and hard science
- Very complex models that embody hard, inflexible rules
- Very large scale, high level of modeling detail, myriad details in complex systems
- Demanding performance requirements: must get **good** solutions **fast**
- Are our algorithms up to the task?

The open pit mining production scheduling problem



- Material to be removed in “blocks” – a lot of them.
- Each block has known physical properties.
- The blocks must be removed following a carefully planned order dictated by structural stability.
- Before a block can be extracted, blocks ”above it” must have been extracted first.

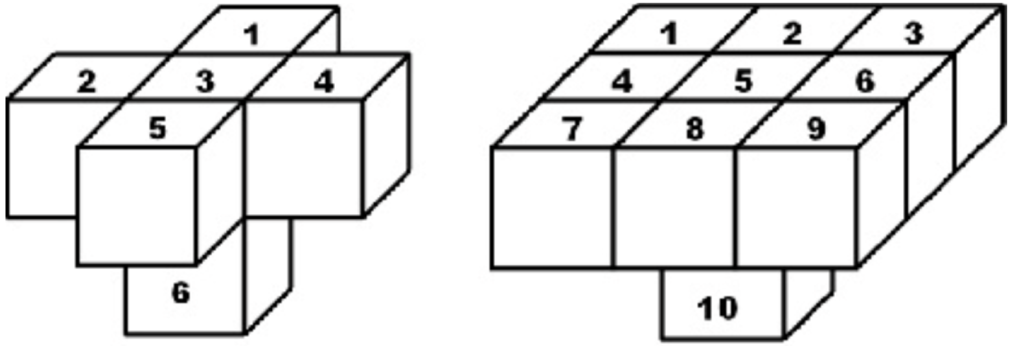


Fig. 2 Sequencing rules can be based, for example, on the removal of five blocks above a given block, block 6 (left) or on the removal of nine blocks above a given block, block 10 (right).

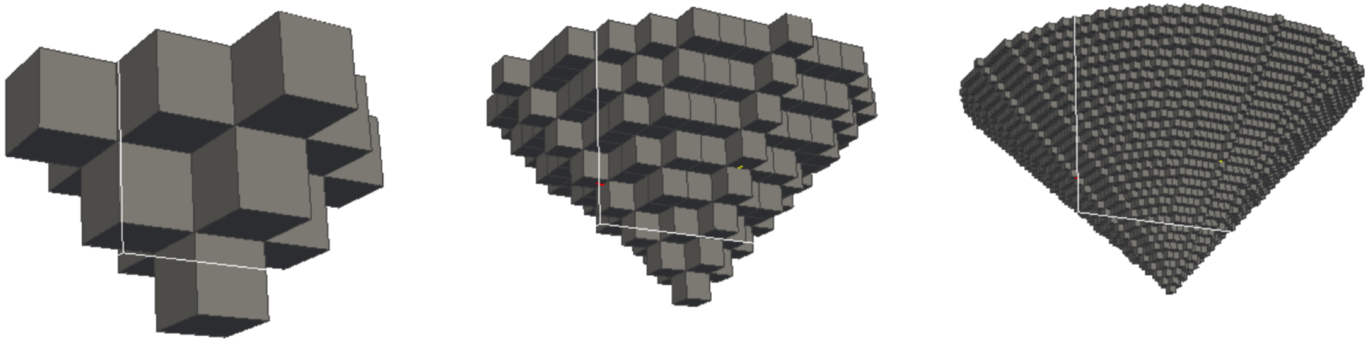


Fig. 3 Sequencing approximation based on the removal of all blocks at a 45-degree angle above a given block, for three, eight and thirty levels

Direct Optimization (Math Programming)

Thys Johnson, 1968

OPTIMUM OPEN-PIT MINE PRODUCTION SCHEDULING

Thys B. Johnson*

Abstract

Traditional mine planning concepts are discussed and suggestions for improvement through use of the developed model are proposed. The approach developed in this paper allows optimal planning of the complete mining-concentrating-refining system over the entire planning horizon and permits the system to dictate how and when to process a block of material, i.e., a dynamic cutoff.

$$\max \sum_{b\in\mathcal{B}} \sum_{d\in\mathcal{D}} \sum_{t\in T} \frac{1}{(1+\alpha)^t} p_{b,d,t} y_{b,d,t}$$

s.t.

$$x_{b,t} = \sum_{d\in\mathcal{D}} y_{b,d,t} \qquad \text{for all } b\in\mathcal{B},\ t\in T$$

$$\sum_{t=1}^{\tau} x_{b,t} \le \sum_{t=1}^{\tau} x_{a,t} \qquad \text{for all } (a,b)\in\mathcal{A},\ \tau\in T$$

$$\sum_{b\in\mathcal{B}} q_b x_{b,t} \le M_t \qquad \text{for all } t\in T$$

$$\sum_{b\in\mathcal{B}} q_b y_{b,d,t} \le U_t^d \qquad \text{for all } d\in\mathcal{D},\ t\in T$$

$$x_{b,t}, y_{b,d,t} \in \{0,1\} \qquad \text{for all } b\in\mathcal{B}, d\in\mathcal{D}, t\in T$$

- Typical number of periods : 10 – 20.
- Typical number of destinations : 2 – 5.
- Typical number of side constraints : 20 – 200.
- Typical number of scheduling units: 10,000 – 10 million.
- Typical number of precedences : 1 million – 4 billion.

Taking a step back ...

$$\max c^T x \text{ subject to}$$

$$Ax \leq b$$

$$Dx \leq d$$

Precedence, nonnegativity

Structured

General/messy, **bad**

Lagrangian relaxation/column generation?

$$\max c^T x \text{ subject to}$$
$$Ax \leq b$$
$$Dx \leq d$$

Idea used in **multicommodity flows** (~2002)

- Subgradient optimization can tail-off or diverge badly.
- Why?
- The solution to the Lagrangian is very bad.
- What to do?
- *Force structure of the Lagrangian solution into a "restricted primal".*

- Column generation **on roids.**

$$(P1) \quad \max c^T x \text{ subject to}$$

$$Ax \leq b$$

$$Dx \leq d \tag{1}$$

- Assume that $\{x : Ax \leq b\} \neq \emptyset$.
- Let $L(P1, \mu)$ be the lagrangian relaxation in which constraints (1) are dualized with penalties μ.

Algorithm Template:

1. Set $\mu^0 = 0$ and set $k = 1$.
2. Solve $L(P1, \mu^{k-1})$ to obtain optimal solution w^k.
3. Find some nontrivial constraint $H^k(x) = h^k$ that is satisfied by w^k.
4. Define the restricted problem

$$(P2(k)) \quad \max c^T x \text{ subject to}$$

$$Ax \leq b \tag{1}$$

$$Dx \leq d \tag{2}$$

$$H^k(x) = h^k \tag{3}$$

5. Solve $P2(k)$ to get optimal primal x^k with value z^k and optimal dual μ^k (corresponding to constraints (2)).
6. Set k=k+1 and GOTO step 2.

$$(P1) \quad \max c^T x \text{ subject to}$$

$$Ax \leq b$$

$$Dx \leq d \qquad (1)$$

- Assume that $\{x : Ax \leq b\} \neq \emptyset$.
- Let $L(P1, \mu)$ be the lagrangian relaxation in which constraints (1) are dualized with penalties μ.

Algorithm Template:

1. Set $\mu^0 = 0$ and set $k = 1$.
2. Solve $L(P1, \mu^{k-1})$ to obtain optimal solution w^k.
   If $k > 1$ and $H^{k-1}(w^k) = h^{k-1}$ then STOP.
3. Find some nontrivial constraint $H^k(x) = h^k$ that is satisfied by w^k.
4. Define the restricted problem

$$(P2(k)) \quad \max c^T x \text{ subject to}$$

$$Ax \leq b \qquad (1)$$

$$Dx \leq d \qquad (2)$$

$$H^k(x) = h^k \qquad (3)$$

5. Solve $P2(k)$ to get optimal primal x^k with value z^k and optimal dual μ^k (corresponding to constraints (2)). If $\mu^k = \mu^{k-1}$ STOP.
6. Set k=k+1 and GOTO step 2.

Theorem:

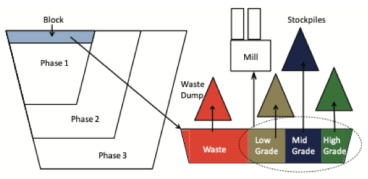
At termination we have solved the LP.

	Marvin	**Mine1B**	**Mine2**	**Mine3 small**	**Mine3 big**
Blocks	9400	29277	96821	675	108264
Parcels	9400	29277	96821	2975	177843
Block arcs	145640	1271207	1053105	1748	2762864
Periods	14	14	25	8	8
Destinations	2	2	2	8	8
Variables	199626	571144	3782250	18970	3503095
Variables Cplex presolved	197666	568890	—	17056	—
Constraints	2048388	17826203	26424496	9593	19935500
Constraints Cplex presolved	2047939	17822237	—	9353	—
Problem arcs	2229186	18338765	3001354	24789	23152350
Side constraints	28	28	50	120	132
Non-knapsack side constraints	0	0	0	10	13
Binding side const. at optimum	14	11	23	33	44
Cplex time (sec)	55141	—	—	52	—
Algorithm Performance					
Iters. to 10^{-5} optimality (sec)	8	8	9	14	30
Time to 10^{-5} optimality (sec)	10	60	344	1	1117
Iters. to comb. optimality	11	12	16	15	39
Time to comb. optimality (sec)	15	95	649	1	1592
Lagrangian time (sec)	13	83	621	0	725
Subproblem LP time (sec)	1	0	6	1	709

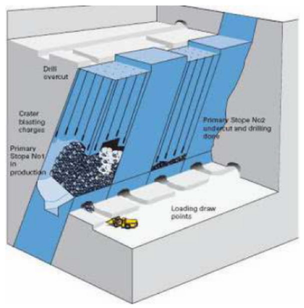
	Newman1	SM2	Marvin	ManySC	Coal1	Coal2	W23
Blocks	1059	18388	8516	3165	34174	33773	74260
Periods	6	30	20	6	9	9	20
Destinations	2	2	2	4	15	17	4
Variables	12708	1103280	340640	24504	1683393	1705498	3564480
Variables Cpx presol	12552	894090	2064496	20710	1677451	1699498	3476640
Constraints	24603	545008	1726636	36830	289329	291391	9251776
Constraints Cpx presol	24603	545008	337860	36738	249664	250001	9251776
Problem arcs	35181	1611452	2050204	48416	1977327	2001301	12667652
Side constraints	12	60	40	6832	3092	3573	84
Homogen side con	0	0	0	12	936	1278	48
Pos dual side con at opt	3	36	13	124	463	609	15
Gurobi sec	4	589	—	12	3580	3061	—*
Cplex sec	4	681	—	21	1460	1214	—
Algorithm Performance							
Tuned Gap at term.	1.4E-15	2.5E-14	1.3E-14	-7.9E-15	3.6E-7	1.3E-8	4.04E-13
Tuned Lagran,Subprob sec	0, 0	8, 1	4, 0	0, 2	11, 1702	9, 485	71, 5
Tuned Iters,Sec to 1e-5 opt	6, 0	10, 12	8, 5	16, 12	50, 1367	42, 586	15, 79
Tuned Iters,Sec to opt	7, 0	13, 16	9, 5	19, 15	54, 1485	47, 597	18, 94
Tuned Iters,Sec to term.	8, 0	14, 17	10, 6	20, 16	59, 1835	50, 612	19, 99

Many extensions/developments!

- **Underground mining.**



Open Pit Operation



Underground Stoping Operation

- **Cutoff grades.**

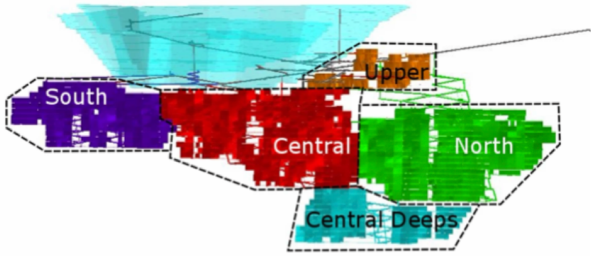
Cutoff grade is the minimum ratio of ore to rock in a block to be extracted.

Low cutoff = longer operation for the mine, but more processing

High cutoff = extraction focuses on more valuable blocks, but lifetime of mine may be too short

Heuristics can be used to decompose a mine into a set of separate operations (using different cutoffs)

(Newman et al)

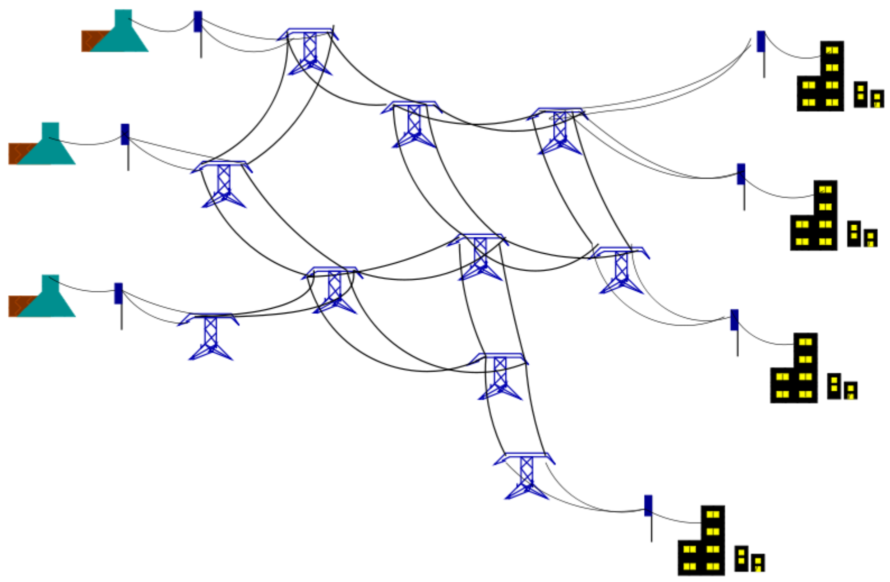


- **Commercialization.**

Standard ACOPF

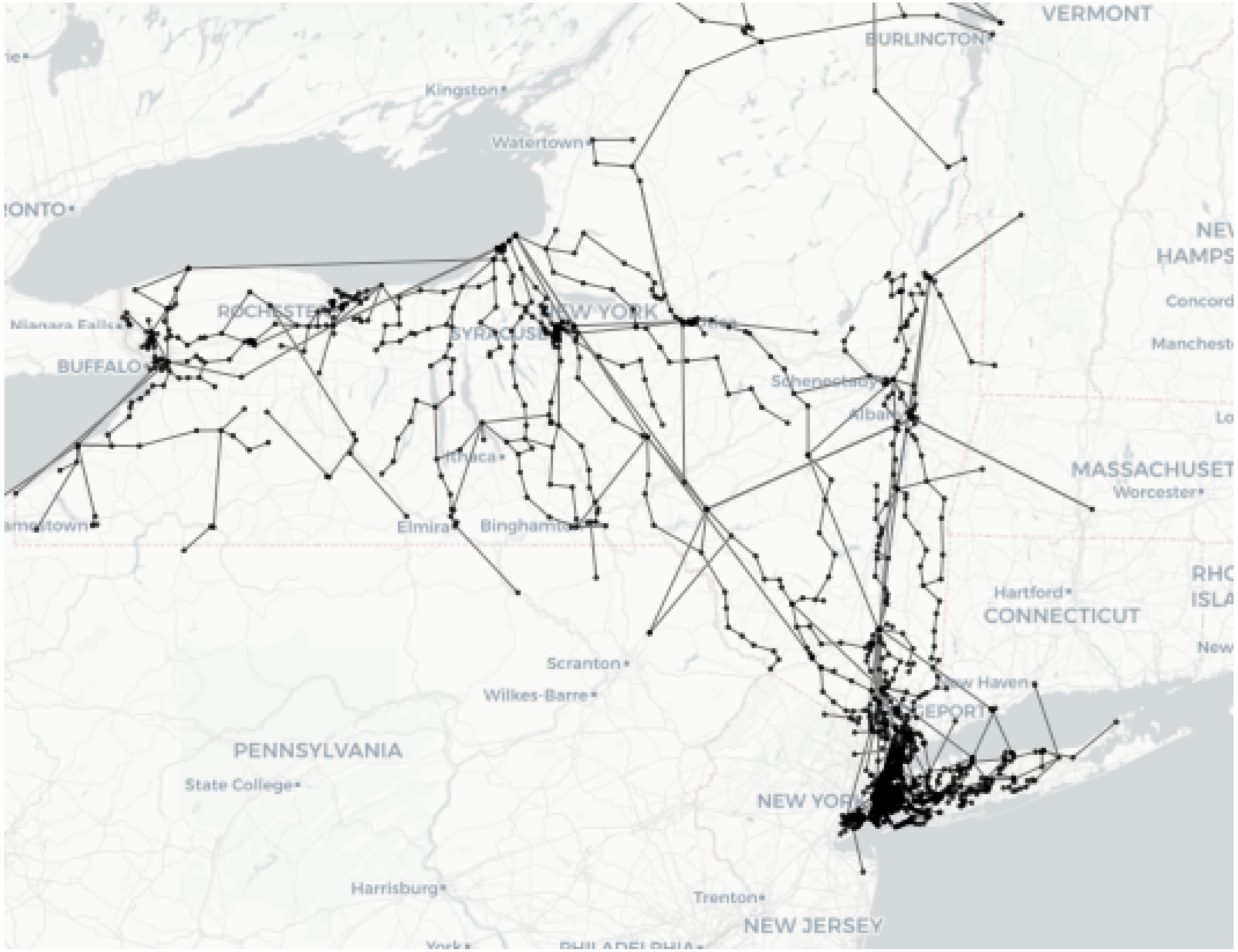
We are given a power system, i.e., a network of

- Generators
- Power lines and transformers
- Buses (nodes)
- Each bus has a load, i.e., numerical demand for power generators, lines, transformers and buses (nodes) with power demands

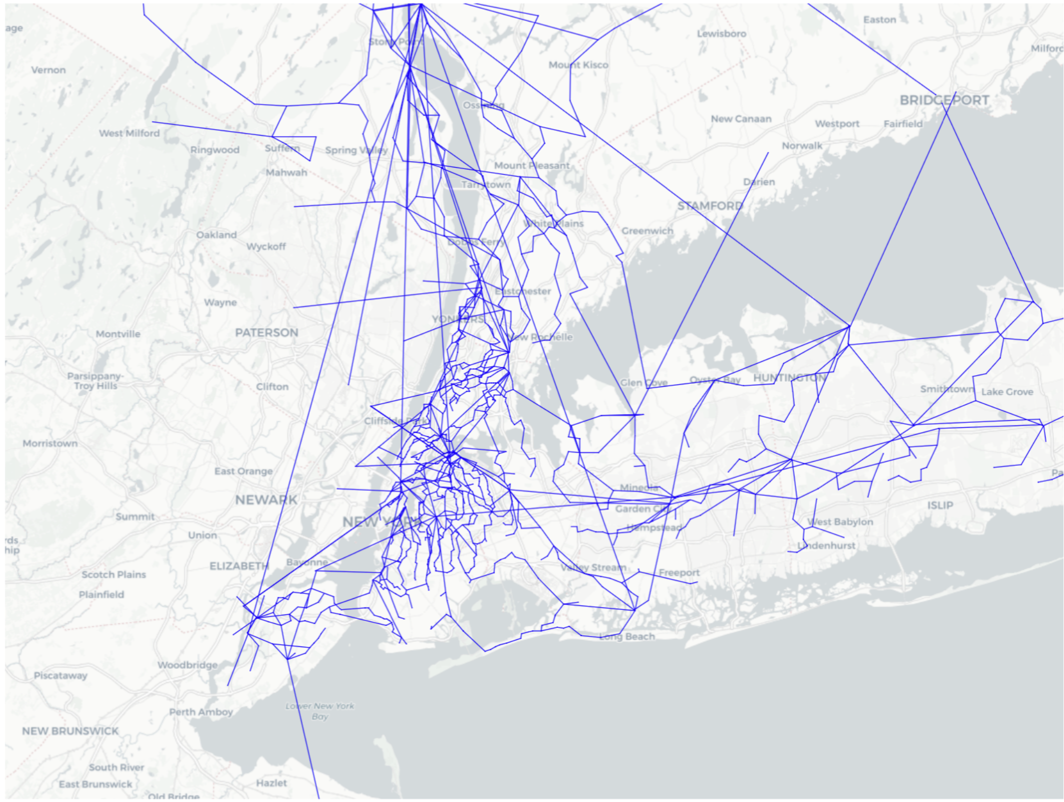


Objective: meet demands at minimum cost

Note: power flows following **laws of physics**



NY system:
1814 buses
500+ generators



A formal textbook statement of standard ACOPF

Minimize cost of generation: $\sum_{g \in \mathcal{G}} F_g(P^g)$

- Here, $\mathcal{G}$ is the set of generators
- P^g is the (active) power generated at g
- F_g is generation cost at g – convex, piecewise-linear or quadratic
  Example: $F_g(P) = 3P^2 + 2P$

Constraints:

- PF (power flow) constraints: choose voltages so that network delivers power from generators to the loads, following **AC power flow laws**
- Voltage magnitudes are constrained
- Power flow on any line km cannot be too large
- The output of any generator is limited

Minimize $\sum_{g\in\mathcal{G}} F_g(\boldsymbol{P^g})$

with constraints:

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

$$\sum_{km\in\delta(k)} \boldsymbol{S_{km}} = \left(\sum_{\boldsymbol{g\in G(k)}} \boldsymbol{P^g} - P_k^d\right) + j\left(\sum_{\boldsymbol{g\in G(k)}} \boldsymbol{Q^g} - Q_k^d\right)$$

Power flow limit on line km:

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

Voltage limit on node k:

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

Generator output limit on node k:

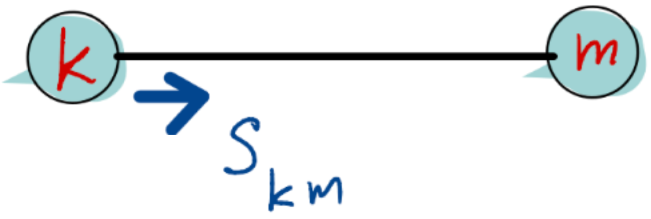
$$P_k^{\min} \leq \boldsymbol{P_k^g} \leq P_k^{\max}$$

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

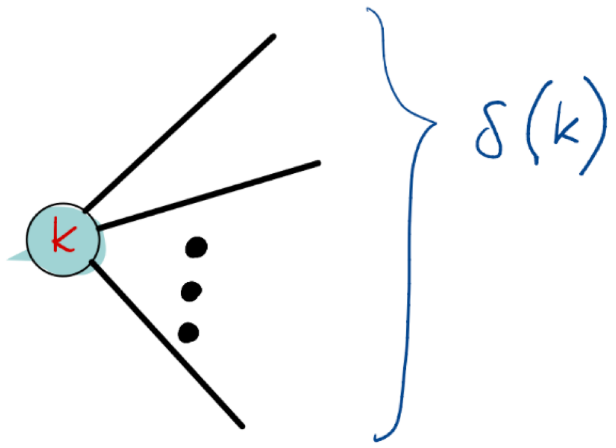
complex power **injected** into **km** at **k**

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

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



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



$$\sum_{km \in \delta(k)} \boldsymbol{S_{km}} = \left(\sum_{\boldsymbol{g \in G(k)}} \boldsymbol{P^g} - P_k^d \right) + j\left(\sum_{\boldsymbol{g \in G(k)}} \boldsymbol{Q^g} - Q_k^d \right)$$

LHS = complex power injected into grid at **k**

Total real power generated at **k**

Real power demand at **k**

Minimize $\sum_{g \in \mathcal{G}} F_g(\ \boldsymbol{P^g})$

with constraints:

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

$$\sum_{km \in \delta(k)} \boldsymbol{S_{km}}\ =\ (\ \sum_{\boldsymbol{g} \in \boldsymbol{G(k)}} \boldsymbol{P^g} - P_k^d)\ +\ j(\ \sum_{\boldsymbol{g} \in \boldsymbol{G(k)}} \boldsymbol{Q^g} - Q_k^d)$$

Power flow limit on line km:

$$|\boldsymbol{S_{km}}|^{\mathbf{2}}\ =\ \Re\mathrm{e}(\ \boldsymbol{S_{km}})^2 + Im(\ \boldsymbol{S_{km}})^2\ \leq\ U_{km}$$

Voltage limit on node k:

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

Generator output limit on node k:

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

But there is an equivalent formulation as a **QCQP** (Quadratically Constrained Quadratic Program)

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

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

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

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

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

Admittance matrix for line km

$$V_i = e_i + j f_i$$

Use rectangular coordinates for voltages

- QCQPs are hard!

- Numerically challenging.
- It is difficult to certify nearness to feasibility of a nearly feasible solution.
- It is difficult to certify infeasibility of a model.
- How do we explain infeasibility of a model? IISs, anyone?
- Real-world cases can be at the boundary of infeasibility.
- Nonlinear != linear

Exploring the Power Flow Solution Space Boundary

Ian A. Hiskens, *Senior Member* and Robert J. Davy

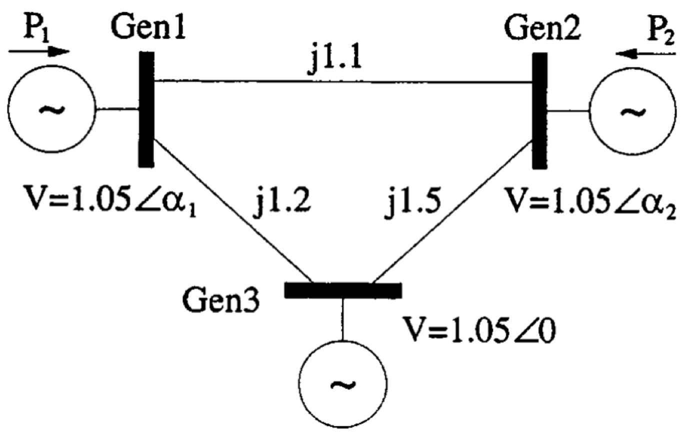
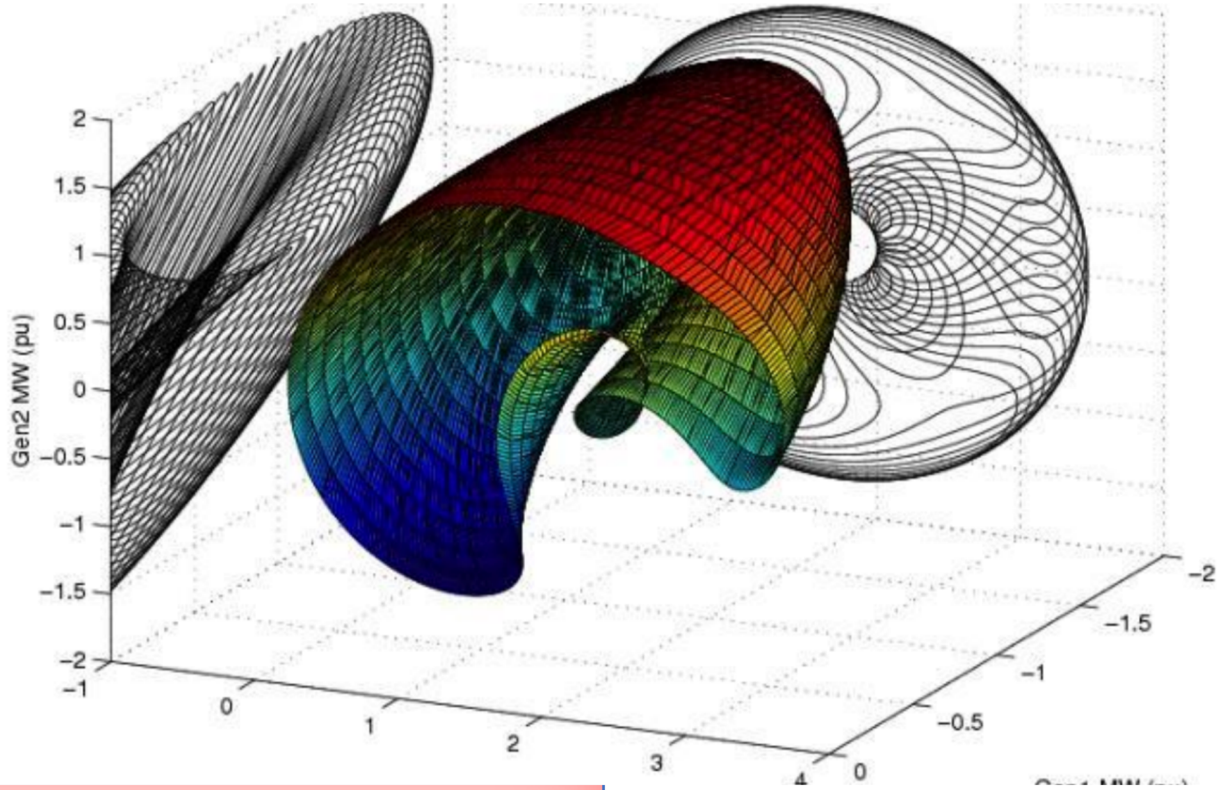


Fig. 6. Three bus system.



A knowledge of the solution boundary of the power flow problem is important for determining the robustness of operating points, and for evaluating strategies for improving robustness. A method of exploring that solution boundary has been developed.

Examples have demonstrated some of the possible forms that the solution boundary can exhibit. It appears that quite complicated behavior is possible. This could have a significant influence on the formulation of algorithms for optimally improving system robustness. It remains to fully explore these issues.

• McCormick relaxation - an important workhorse

$w = xy$

$x^L \leq x \leq x^U \quad y^L \leq y \leq y^U$ where x^L, x^U, y^L, y^U are upper and lower bound values for x and y respectively

Convex hull provided by under/over estimators

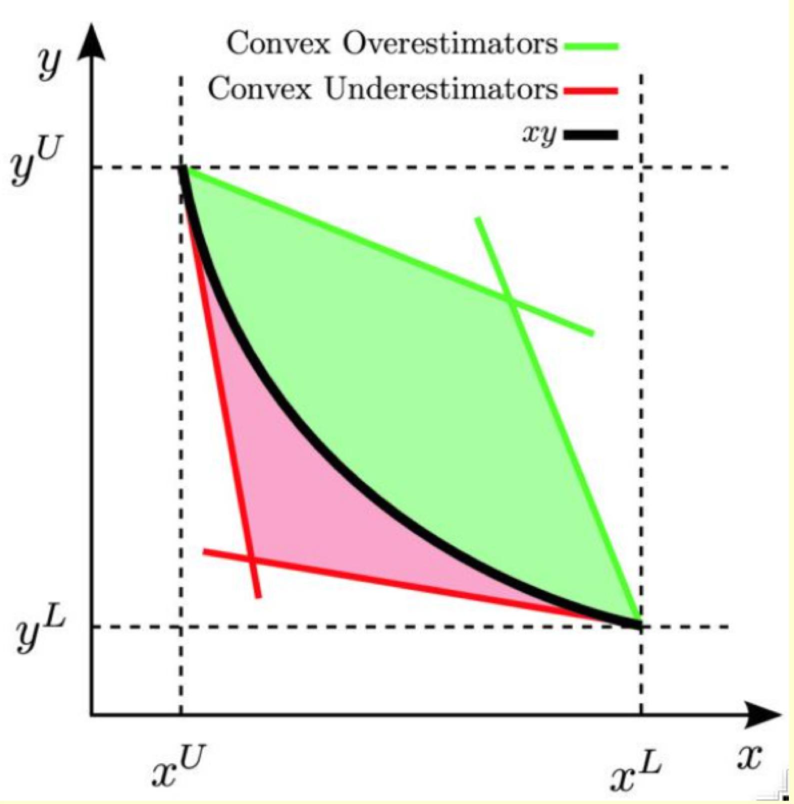
The underestimators of the function are represented by:

$w \geq x^L y + x y^L - x^L y^L \; ; \; w \geq x^U y + x y^U - x^U y^U$

The overestimators of the function are represented by:

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

Works well in tandem with spatial branching



(source: Wikipedia)

• Issues with McCormick relaxation and spatial b&b?

- On **hard** instances, e.g., **hard and large ACOPF**, bounds can be very weak and we will grow an immense tree

- **Numerical issues!** SOC and rotated cone constraints approximated with **many** outer envelope cuts

- **Numerical issues!** Nodes can be very iffy, in particular:

  Infeasibility fathoming! Mr. Solver, are you sure that node is infeasible?

- And what does infeasibility ***actually mean***, in light of our prior slides?

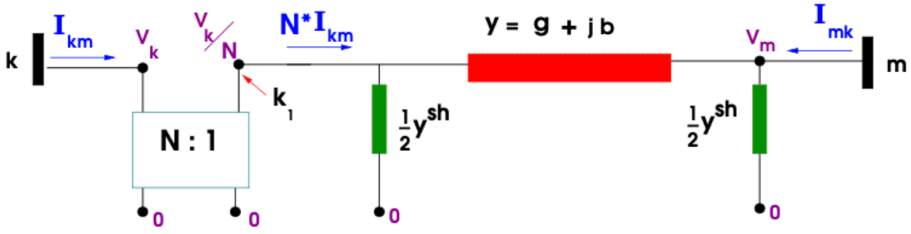
- Upper bounds: Mr. Solver, are you sure that solution is **feasible**?

Upper bounds: log-barrier methods

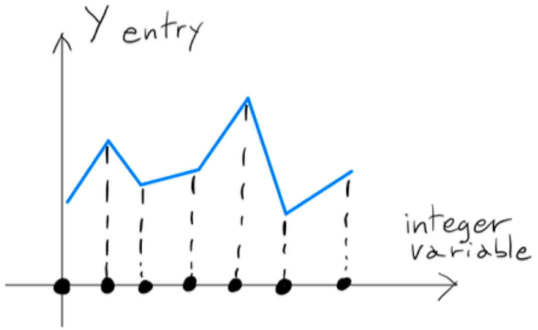
- A must-have tool!
- Knitro, IPOPT, LOQO, others?
- Knitro and IPOPT are excellent
- Also, very elegant theory!
- But, also, excellent implementations!
- Theory only guarantees convergence to (?) a critical point for the barrier function
- Works extremely well for standard ACOPF
- A (minor?) issue: solutions can exhibit small infeasibilities

GO competition: configurable transformers and shunts

- Example: tap ratio and angle in a transformer can be adjusted



- Impedance correction factors modeled using a piecewise-linear curve



- Function can be quite nonlinear with local optima

- Switched shunts, in blocks (at buses)

- Altogether, a large number of integer variables

And much, much more

- Contingencies (security-constrained ACOPF)
- Penalties for infeasibilities
- Migration from existing solution
- Tight timeframe available for computation
- MPI required: 4 boxes with 16 cores each
- …

GO competition: data sets

- Combination of industry and realistic synthetic data sets
- Both large and with many contingencies
- Industry example **C2T2N34363**:
  34,000 buses, **41,000** lines, **900** generators, **3000** contingencies
  thousands of integer variables
- A single ACOPF run on such networks is nontrivial: Example **C2FEN19402**:
  - 19,402 buses, 968 generators, 13,000 lines
  - 267904 variables, 200890 constraints, 6692 contingencies
  - **KNITRO solves Base case** in **51.05** seconds on **11** cores.
- Available time is limited: 5 minutes to one hour

The king of the hill – log barrier methods

Today, two implementations dominate:

- **KNITRO** (Waltz, Nocedal, 2003): A “merit function” method.
- **IPOPT** (Wachter and Biegler, 2004): A “filter” method.

KNITRO and IPOPT followed a long line of work due to many authors!

1. Log barrier methods can (very closely) optimize very large ACOPF problems in **minutes**
2. Nothing else comes close. Relaxation methods only prove bounds – don’t provide **solutions**
3. New kid on the block: **Gurobi**. Integrated log-barrier, integer programming and relaxations!

Datasets in Final Event: 22 networks

- Buses: 403 - 31,777
- Contingencies: 54 - 1800

PRIZES

For Challenge 2, the ARPA-E Benchmark team is not prize eligible and does not occupy a rank during the consideration of prize awards.

Total Prizes ($k) to be Awarded, Subject to Eligibility

Team	Trial Event 3	Final Event	FE + T3
GravityX	130	600	730
Artelys	170	360	530
GOT-BSI-OPF	0	420	420
Pearl Street Technologies	70	270	340
Electric Stampede	140	0	140
GMI-GO	60	60	120
Monday Mornings	0	60	60
GO-SNIP	0	30	30
Gordian Knot	30	0	30
total	600	1,800	2,400

Joint work with Gonzalo Muñoz (2015)

Arbitrarily close approximations of QCQPs using pure-binary integer programs

➡ Approximate continuous variables using binary variables. Why?

Given $0 \leq r \leq 1$ (quantity to approximate) and $0 < \gamma < 1$ (the tolerance)

set

$$L = L(\gamma) \doteq \lceil \log_2 \gamma^{-1} \rceil$$

Then there exist 0/1-values z_h, $1 \leq h \leq L$, with

$$\sum_{h=1}^{L} 2^{-h} z_h \leq r \leq \sum_{h=1}^{L} 2^{-h} z_h + 2^{-L} \leq \sum_{h=1}^{L} 2^{-h} z_h + \gamma \leq 1.$$

A generic polynomial-optimization problem

$$\text{(PO)}: \quad \min \; c^T x$$
$$\text{subject to:} \quad f_i(x) \geq 0 \quad 1 \leq i \leq m$$
$$x_j \in [0,1] \quad 0 \leq j \leq n.$$

$$f_i(x) = \sum_{\alpha \in I(i)} f_{i,\alpha} x^{\alpha}. \quad \text{(monomial notation)}$$

$$\mathbf{A} = \max\{\alpha\}, \qquad \boldsymbol{\delta} := \mathbf{1} - (\mathbf{1} - \boldsymbol{\gamma})^{\boldsymbol{A}}, \qquad \boldsymbol{\gamma} = \text{tolerance}$$

$$L = L(\gamma) \doteq \lceil \log_2 \gamma^{-1} \rceil$$

$$\text{PO}(\gamma): \quad \min \sum_{j=1}^{n} c_j \left(\sum_{h=1}^{L} 2^{-h} z_{j,h} \right)$$

$$\text{s.t.} \quad \sum_{\alpha \in I(i)} f_{i,\alpha} \left[\prod_{j=1}^{n} \left(\sum_{h=1}^{L} 2^{-h} z_{j,h} \right)^{\alpha_j} \right] \geq -\delta \, \|f_i\|_1, \quad 1 \leq i \leq m$$

$$z_{j,h} \in \{0,1\}, \; \forall j \in \{1,\ldots,n\}, \; 1 \leq h \leq L.$$

Formulation is both $O(\gamma)$-feasible and optimal. And sparsity-preserving.

Is this a crazy approach to QCQP?

- We start with a **bad**, large QCQP and we end up with a much, much larger and probably **badder** but linear binary IP

- But it is linear …

- Numerics should be less of an issue …

- And it has a lot of structure …

- This is ongoing work with Matías Villagra and Yuri Faenza