Mixed-Integer Nonlinear Optimization: Introduction, Modeling, and Applications

GIAN Short Course on Optimization: Applications, Algorithms, and Computation

Sven Leyffer

Argonne National Laboratory

September 12-24, 2016

Collaborators

Pietro Belotti and Ashutosh Mahajan

Christian Kirches, Jeff Linderoth, and Jim Luedtke

Outline

Mixed-Integer Nonlinear Optimization

Mixed-Integer Nonlinear Program (MINLP)

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & c(x) \le 0 \\ & x \in \mathcal{X} \\ & x_i \in \mathbb{Z} \text{ for all } i \in \mathcal{I} \quad \text{set of integers} \end{aligned}$$

- $f : \mathbb{R}^n \to \mathbb{R}$, $c : \mathbb{R}^n \to \mathbb{R}^m$ smooth (often convex) functions
- $\mathcal{X} \in \mathbb{R}^n$ bounded, polyhedral set, e.g. $\mathcal{X} = \{x : l \le A^T x \le u\}$
- $\mathcal{I} \subset \{1, \ldots, n\}$ subset of integer variables
- $x_i \in \mathbb{Z}$ for all $i \in \mathcal{I}$ ... combinatorial problem
- Combines challenges of handling nonlinearities with combinatorial explosion of integer variables
- More general constraints possible, e.g. $l \le c(x) \le u$ etc.

Complexity of MINLP

Mixed-Integer Nonlinear Program (MINLP)

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & c(x) \leq 0 \\ & x \in \mathcal{X} \\ & x_i \in \mathbb{Z} \text{ for all } i \in \mathcal{I} \end{aligned}$$

Complexity of MINLP

- MINLP is NP-hard: includes MILP, which are NP-hard [Kannan and Monma, 1978]
- Worse: MINLP are undecidable [Jeroslow, 1973]: quadratically constrained IP for which no computing device can compute the optimum for all problems in this class
  ... but we're OK if $\mathcal{X}$ is compact!

Notation

Some notation used throughout the course ...

- $f^{(k)} = f(x^{(k)})$ evaluated at $x = x^{(k)}$
- $\nabla f^{(k)} = \nabla f(x^{(k)})$ gradient
- Hessian of Lagrangian $\mathcal{L}(x, \lambda) = f(x) - \sum \lambda_i c_i(c)$ is $\nabla^2 \mathcal{L}^{(k)}$
  ... assumes $\mathcal{X}$ polyhedral
- Subscripts denote components, e.g. x_i is component i of x
- If $\mathcal{J} \subset \{1, \ldots, n\}$ then $x_{\mathcal{J}}$ are components of x corres. to $\mathcal{J}$
- $x_{\mathcal{I}}$ integer and $x_{\mathcal{C}}$ are the continuous variables, $p = |\mathcal{I}|$
- Floor and ceiling operators: $\lfloor x_i \rfloor$ and $\lceil x_i \rceil$:
  - $\lfloor x_i \rfloor$ largest integer smaller than or equal to x_i
  - $\lceil x_i \rceil$ smallest integer larger than or equal to x_i

Recall: Convexity of Nonlinear Functions

MINLP techniques distinguish convex and nonconvex MINLPs.
For our purposes, we define convexity as ...

Definition (Convex Functions)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex, iff $\forall x^{(0)}, x^{(1)} \in \mathbb{R}^n$ we have:

$$f(x^{(1)}) \geq f(x^{(0)}) + (x^{(1)} - x^{(0)})^T \nabla f^{(0)}$$

In a slight abuse of notation, we say that ...

Definition (Convexity of MINLP)

MINLP is a *convex* if the problem functions $f(x)$ and $c(x)$ are convex functions. If either $f(x)$ or any $c_i(x)$ is a nonconvex function, then MINLP is *nonconvex*.

Recall Convexity (cont.)

We also define the convex hull of a set S as ...

Definition (Convex Hull)

For a set S, the *convex hull of S* is conv(S):

$$\left\{x|x = \lambda x^{(1)} + (1-\lambda)x^{(0)},\ \forall 0 \le \lambda \le 1,\ \forall x^{(0)}, x^{(1)} \in S\right\}.$$

- If $\mathcal{X} = \{x \in \mathbb{Z}^p : l \le x \le u\}$ and $l \in \mathbb{Z}^p$, $u \in \mathbb{Z}^p$, then $\text{conv}(\mathcal{X}) = [l, u]^p$
- Finding convex hull is hard, even for polyhedral $\mathcal{X}$.
- Convex hull important for MILP ...

Theorem (LP Relaxations of MILP)

MILP can be solved as LP over the convex hull of feasible set.

MILP $\neq$ MINLP

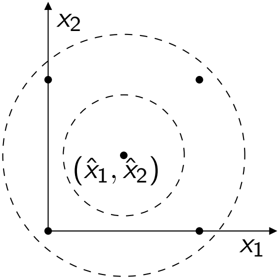
Important difference between MINLP and MILP

$$\underset{x}{\text{minimize}} \sum_{i=1}^{n} (x_i - \tfrac{1}{2})^2, \quad \text{subject to } x_i \in \{0, 1\}$$

... solution is not extreme point (lies in interior)

Remedy: Introduce objective η and a constraint $\eta \geq f(x)$

$$\begin{cases} \underset{\eta, x}{\text{minimize}} & \eta, \\ \text{subject to} & f(x) \leq \eta, \\ & c(x) \leq 0, \\ & x \in \mathcal{X}, \\ & x_i \in \mathbb{Z}, \ \forall i \in \mathcal{I}. \end{cases}$$



Assume MINLP objective is linear

MILP $\neq$ MINLP

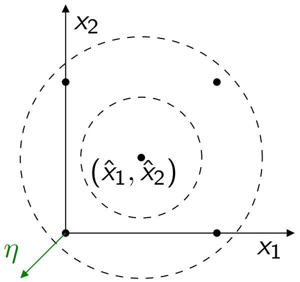
Important difference between MINLP and MILP

$$\underset{x}{\text{minimize}} \sum_{i=1}^{n} (x_i - \tfrac{1}{2})^2, \quad \text{subject to } x_i \in \{0, 1\}$$

... solution is not extreme point (lies in interior)

Remedy: Introduce objective η and a constraint $\eta \geq f(x)$

$$\begin{cases} \underset{\eta, x}{\text{minimize}} & \eta, \\ \text{subject to } & f(x) \leq \eta, \\ & c(x) \leq 0, \\ & x \in \mathcal{X}, \\ & x_i \in \mathbb{Z}, \ \forall i \in \mathcal{I}. \end{cases}$$



Assume MINLP objective is linear

Outline

Relaxation and Constraint Enforcement

Relaxation

- Used to compute a lower bound on the optimum
- Obtained by enlarging feasible set; e.g. ignore constraints
- Typically much easier to solve than MINLP

Constraint Enforcement

- Exclude solutions from relaxations not feasible in MINLP
- Refine or tighten of relaxation; e.g. add valid inequalities

Upper Bounds

- Obtained from any feasible point; e.g. solve NLP for fixed $x_{\mathcal{I}}$

Relaxations of Integrality

Definition (Relaxation)

Optimization problem $\min\{\breve{f}(x) : x \in \mathcal{R}\}$ is a relaxation of $\min\{f(x) : x \in \mathcal{F}\}$, iff $\mathcal{R} \supset \mathcal{F}$ and $\breve{f}(x) \leq f(x)$ for all $x \in \mathcal{F}$.

Goal: relaxation easy to solve globally, e.g. MILP or NLP

Relaxing Integrality

- Relax Integrality $x_i \in \mathbb{Z}$ to $x_i \in \mathbb{R}$ for all $i \in \mathcal{I}$
- Gives *nonlinear relaxation* of MINLP, or NLP:

$$\begin{cases} \underset{x}{\text{minimize}} & f(x), \\ \text{subject to} & c(x) \leq 0, \\ & x \in \mathcal{X}, \quad \text{continuous} \end{cases}$$

- Used in branch-and-bound algorithms

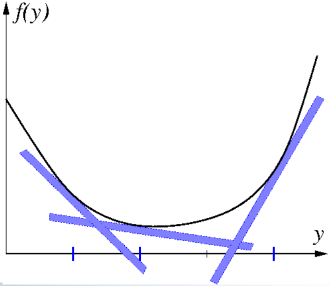
Relaxations of Nonlinear Convex Constraints

Relaxing Convex Constraints

- Convex $0 \geq c(x)$ and $\eta \geq f(x)f$ relaxed by supporting hyperplanes

$$\eta \geq f^{(k)} + \nabla f^{(k)^T}(x - x^{(k)})$$
$$0 \geq c^{(k)} + \nabla c^{(k)^T}(x - x^{(k)})$$

  for a set of points $x^{(k)}, \ k = 1, \ldots, K$.
- Obtain polyhedral relaxation of convex constraints.
- Used in the outer approximation methods.



Relaxations of Nonconvex Constraints

Relaxing Nonconvex Constraints

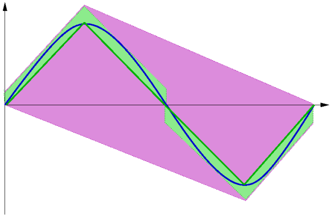
- Construct convex underestimators, $\check{f}(x)$ and $\check{c}(x)$ for nonconvex functions $c(x)$ and $f(x)$:

$$\check{f}(x) \leq f(x) \quad \text{and} \quad \check{c}(x) \leq c(x), \; \forall x \in \text{conv}(\mathcal{X}).$$

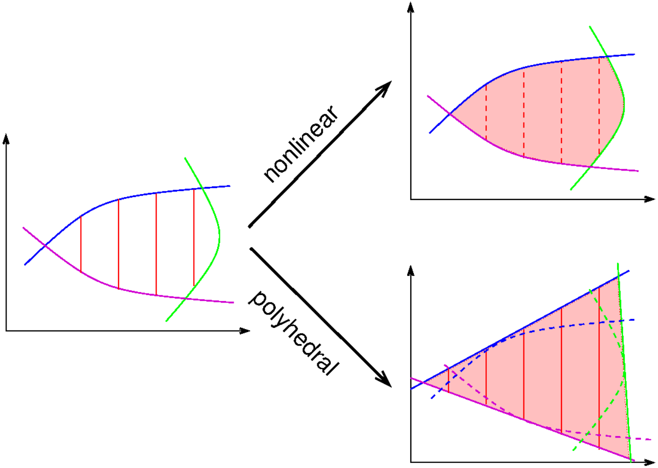
- Relax constraints $z \geq f(x)$ and $0 \geq c(x)$ as

$$z \geq \check{f}(x) \quad \text{and} \quad 0 \geq \check{c}(x).$$

- Used in spatial branch-and-bound.



Relaxations Summary



Nonlinear and polyhedral relaxation

Relaxations

Relaxations can be combined to produce better algorithms

- Relax convex underestimators via supporting hyperplanes.
- Relax integrality of polyhedral relaxation to obtain an LP.

Relaxations are useful because we have following result:

Theorem (Relaxation Property)

If the solution of the relaxation of the η-MINLP is feasible in the η-MINLP, then it solves the MINLP.

... but if solution of relaxation is not feasible, then need ...

Constraint Enforcement

Goal: Given solution of relaxation, $\hat{x}$, not feasible in MINLP, exclude it from further consideration to ensure convergence

Three constraint enforcement strategies

1. Relaxation refinement: tighten the relaxation
2. Branching: disjunction to exclude set of non-integer points
3. Spatial branching: divide region into sub-regions

Strategies can be combined ...

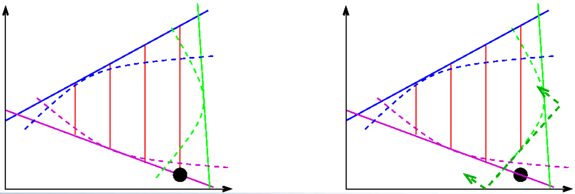
Constraint Enforcement: Refinement

Tighten the relaxation to remove current solution $\hat{x}$ of relaxation

- Add a valid inequality to relaxation, i.e. an inequality that is satisfied by all feasible solutions of MINLP
- Valid inequality is called a cut if it excludes $\hat{x}$
- Example: $c(x) \leq 0$ convex, and $\exists i : c_i(\hat{x}) > 0$, then

$$0 \geq \hat{c}_i + \nabla \hat{c}^T (x - \hat{x})$$

  cuts off $\hat{x}$. Proof: Exercise.
- Used in Benders decomposition and outer approximation.
- MILP: cuts are basis for branch-and-cut techniques.



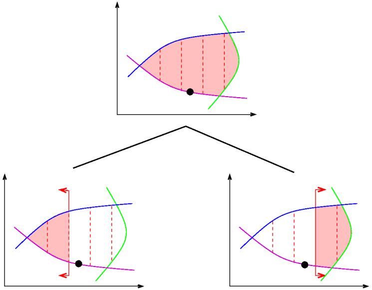
Constraint Enforcement: Branching

Eliminate current $\hat{x}$ solution by branch on integer variables:

1. Select fractional $\hat{x}_i$ for some $i \in \mathcal{I}$
2. Create two new relaxations by adding

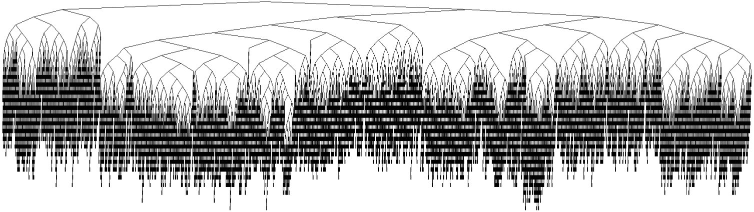
$$x_i \leq \lfloor \hat{x}_i \rfloor \quad \text{and} \quad x_i \geq \lceil \hat{x}_i \rceil \qquad \text{respectively}$$

... solution to MINLP lies in one of the new relaxations.



... creates branch-and-bound tree

Branch-and-Bound Trees can be Huge



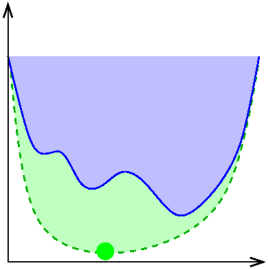
Tree after 360 s CPU time has more than 10,000 nodes

Constraint Enforcement: Spatial Branching

Enforcement for relaxed nonconvex constraints

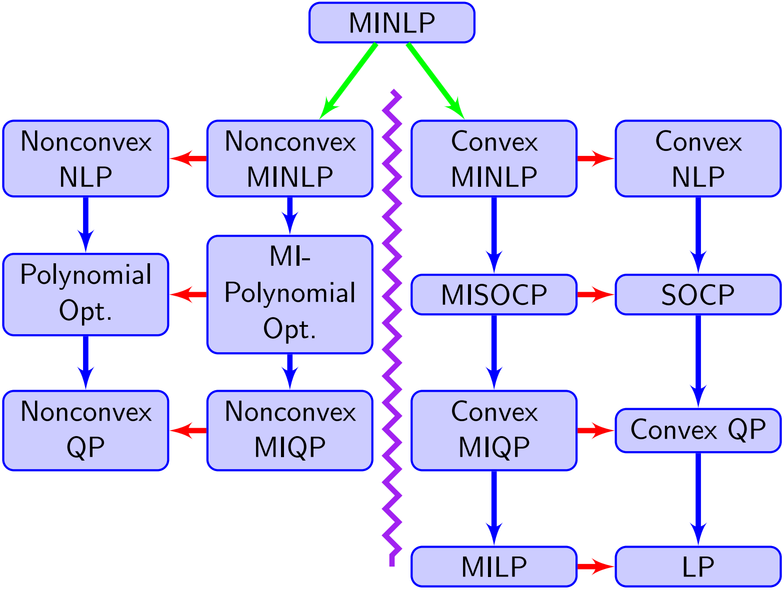
- Combine branching and relaxation refinement
- Branch on continuous variable and split domain in two parts.
- Create new relaxation over (reduced) sub-domains.
- Generates tree similar to integer branching.
- Mix with interval techniques to eliminate sub-domains.

Nonconvex MINLPs combine all 3 enforcement techniques.



Outline

MINLP Tree


MINLP
Nonconvex NLP
Nonconvex MINLP
Convex MINLP
Convex NLP
Polynomial Opt.
MI-Polynomial Opt.
MISOCP
SOCP
Nonconvex QP
Nonconvex MIQP
Convex MIQP
Convex QP
MILP
LP

How Big Is It?

The Leyffer-Linderoth-Luedtke (LLL) Measure of Complexity

You have a problem of class X that has Y decision variables? What is the largest value of Y for which one of Sven, Jim, and Jeff would be willing to bet \$50 that a "state-of-the-art" solver could solve the problem?

How Big Is It?

The Leyffer-Linderoth-Luedtke (LLL) Measure of Complexity

You have a problem of class X that has Y decision variables? What is the largest value of Y for which one of Sven, Jim, and Jeff would be willing to bet \$50 that a "state-of-the-art" solver could solve the problem?

Convex		Nonconvex	
Prob. Class (X)	**# Var** (Y)	**Prob. Class** (X)	**# Var** (Y)
MINLP	500	MINLP	100
NLP	5×10^4	NLP	100
MISOCP	1000	MIPP	150
SOCP	10^5	PP	150
MIQP	1000	MIQP	300
QP	5×10^5	QP	300
MILP	2×10^4		
LP	5×10^7		

MINLP Modeling Practices

Modeling plays a fundamental role in MILP see [Williams, 1999]
... even more important in MINLP

- MINLP combines integer and nonlinear formulations
- Reformulations of nonlinear relationships can be convex
- Interactions of nonlinear functions and binary variables
- Sometimes we can linearize expressions

MINLP Modeling Preference

We prefer linear over convex over nonconvex formulations.

MINLP Modeling Practices

Modeling plays a fundamental role in MILP see [Williams, 1999]
... even more important in MINLP

- MINLP combines integer and nonlinear formulations
- Reformulations of nonlinear relationships can be convex
- Interactions of nonlinear functions and binary variables
- Sometimes we can linearize expressions

MINLP Modeling Preference

We prefer linear over convex over nonconvex formulations.

The great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.
- R. Tyrrell Rockafellar

Linearization of Constraints

Assum $x_2 \neq 0$. A simple transformation (a constant parameter):

$$\frac{x_1}{x_2} = a \;\Leftrightarrow\; x_1 = a x_2$$

Linearization of bilinear terms $x_1 x_2$ with:

- Binary variable $x_2 \in \{0, 1\}$
- Variable upper bound: $0 \leq x_1 \leq U x_2$

... introduce new variable x_{12} to replace $x_1 x_2$ and add constraints

$$0 \leq x_{12} \leq x_2 U \quad \text{and} \quad -U(1 - x_2) \leq x_1 - x_{12} \leq U(1 - x_2),$$

Never Multiply a Nonlinear Function by a Binary

Previous example generalizes to nonlinear functions
Often binary variables "switch" constraints on/off

Warning

Never model on/off constraints by multiplying by a binary variable.

Three alternative approaches

- Disjunctive programming, see [Grossmann and Lee, 2003]
- Perspective formulations (not always), see [Günlük and Linderoth, 2012]
- Big-M formulation (weak relaxations)

Avoiding Undefined Nonlinear Expressions

MINLP solvers fail because NLP solver gets IEEE exception, e.g.

$$c(x_1) = -\ln(\sin(x_1)) \leq 0,$$

cannot be evaluated at $\sin(x_1) \leq 0$

Reformulate equivalently as

$$\tilde{c}(x_2) = -\ln(x_2) \leq 0, \;\; x_2 = \sin(x_1), \quad \text{and} \quad x_2 \geq 0.$$

IPM solvers never evaluate at $x_2 \leq 0$
Active-set method can also safeguard against $x_2 \leq 0$

- $x_2 \geq 0$ is s simple bound which can be enforced exactly
- $x_2 = 0$ get IEEE exception $\Rightarrow$ trap & reduce trust-region
- As $x_2 \to 0$, the constraint violation $c(x_2) \to \infty$

Modeling of Discrete Variables

We can model discrete variables such as

$$y \in \{Y_1, Y_2, \dots, Y_k\}$$

where Y_i are discrete parameters (e.g. pipe diameters) with special ordered sets (SOS):

$$y = \sum_{i=1}^{k} z_i Y_i, \quad 1 = \sum_{i=1}^{k} z_i, \quad z_i \in \{0,1\}$$

see [Beale and Tomlin, 1970, Beale and Forrest, 1976]

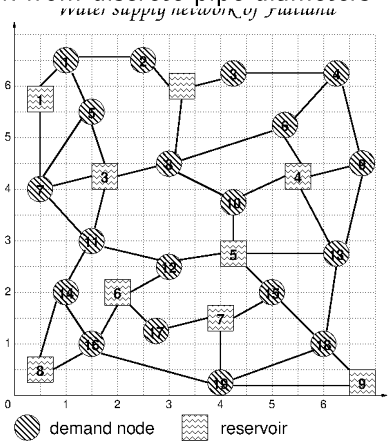
- Similarly linearize univariate functions $f(z)$, $z \in \mathbb{Z}$
- Generalizes to higher dimensions
- Solvers detect SOS structure and use special branching rules

Design of Water Distribution Networks

Model of water, gas, air networks

Goal: design minimum cost network from discrete pipe diameters

- $\mathcal{N}$ nodes in network
- $\mathcal{S}$ source nodes
- $\mathcal{A}$: arcs in the network



Design of Water Distribution Networks

Goal

Design minimum cost network from discrete pipe diameters

$\mathcal{N}$ nodes, $\mathcal{S}$ source nodes, $\mathcal{A}$: arcs in the network

Variables:

- q_{ij}: flow pipe $(i,j) \in \mathcal{A}$
- d_{ij}: diameter of pipe $(i,j) \in \mathcal{A}$, where $d_{ij} \in \{P_1, \ldots, P_r\}$
- h_i: hydraulic head at node $i \in \mathcal{N}$
- z_{ij}: binary variables model flow direction $(i,j) \in \mathcal{A}$
- a_{ij}: area of cross section $(i,j) \in \mathcal{A}$
- y_{ijk}: SOS-1 variables to model diameter

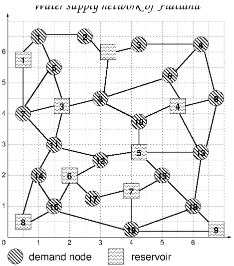
NB: Area $a_{ij} = \pi d_{ij}^2/4$ is redundant ... but useful!

Design of Water Distribution Networks

$\mathcal{N}$ nodes

$\mathcal{S} \subset \mathcal{N}$ source nodes

$\mathcal{A}$: arcs in the network



Equations for q_{ij} flow pipe $(i,j) \in \mathcal{A}$

- Conservation of flow at every node

$$\sum_{(i,j)\in\mathcal{A}} q_{ij} - \sum_{(j,i)\in\mathcal{A}} q_{ji} = D_i, \ \forall i \in \mathcal{N} - \mathcal{S}.$$

- Flow bounds are nonlinear in d_{ij} ... linear in a_{ij}:

$$-V_{\max} a_{ij} \le q_{ij} \le V_{\max} a_{ij}, \ \forall (i,j) \in \mathcal{A}.$$

Design of Water Distribution Networks

Modeling Trick: SOS & Nonlinear Expressions

Modeling discrete $d_{ij} \in \{P_1, \ldots, P_r\}$ and nonlinear $a_{ij} = \pi d_{ij}^2/4$:

1. Introduce SOS-1 variables $y_{ijk} \in \{0, 1\}$ for $k = 1, \ldots, r$
2. Model discrete choice as linear equation

$$\sum_{k=1}^{r} y_{ijk} = 1, \quad \text{and} \quad \sum_{k=1}^{r} P_k y_{ijk} = d_{ij}. \ \forall (i,j) \in \mathcal{A},$$

3. Model nonlinear relationship as linear equation

$$\sum_{k=1}^{r} (\pi P_k/4) y_{ijk} = a_{ij}, \ \forall (i,j) \in \mathcal{A}.$$

$\Rightarrow$ no longer need nonlinear equation $a_{ij} = \pi d_{ij}^2/4$!

Design of Water Distribution Networks

Nonsmooth pressure loss model along arc $(i,j) \in \mathcal{A}$

$$h_i - h_j = \frac{\operatorname{sgn}(q_{ij})|q_{ij}|^{c_1} c_2 L_{ij} K_{ij}^{-c_1}}{d_{ij}^{c_3}}$$

... introduce binary variables to model nonsmooth term $|q_{ij}|^{c_1}$

1. Add binary variables $z_{ij} \in \{0,1\}$, and $q_{ij} = q_{ij}^+ - q_{ij}^-$.

$$0 \le q_{ij}^+ \le Q_{\max} z_{ij}, \quad 0 \le q_{ij}^- \le Q_{\max}(1 - z_{ij}),$$

2. Pressure drop becomes

$$h_i - h_j = \frac{\left[\left(q_{ij}^+\right)^{c_1} - \left(q_{ij}^-\right)^{c_1}\right] c_2 L_{ij} K_{ij}^{-c_1}}{d_{ij}^{c_3}}, \ \forall (i,j) \in \mathcal{A}.$$

... can again linearize the $d_{ij}^{c_3}$ expression with SOS

... alternative uses complementarity: $0 \le q_{ij}^+ \ \perp \ q_{ij}^- \ge 0$

Optimization of IEEE 802.11 Broadband Networks

Optimize 802.11 broadband networks for resource sharing meshes

- objective: minimizing co-channel and inter-channel interference
- integrality: assign channels to basic nodes within a network
- 13 Direct Sequence Spread Spectrum (DSSS) overlapping channels
- co-channel interference: two access points with same channel
- inter-channel interference: cards with overlapping channels transmit simultaneously

$\Rightarrow$ general nonconvex MINLP

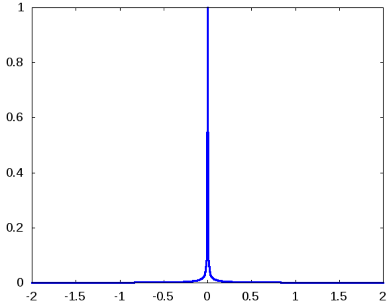
Original model has one horrible constraint ...

Optimization of IEEE 802.11 Broadband Networks

... and the horrible constraint is ...

$$z = \frac{1}{1 + 1000(x - y)^{10}}$$

- highly nonlinear/nonconvex

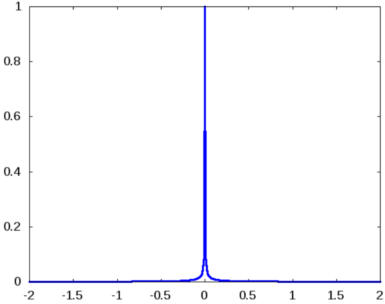


Optimization of IEEE 802.11 Broadband Networks

... and the horrible constraint is ...

$$z = \frac{1}{1 + 1000(x - y)^{10}}$$

- highly nonlinear/nonconvex
- $z = 1$, if $x = y$
- $z = 0$, if $x \neq y$

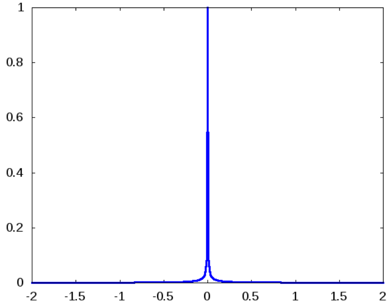


Optimization of IEEE 802.11 Broadband Networks

... and the horrible constraint is ...

$$z = \frac{1}{1 + 1000(x - y)^{10}}$$

- highly nonlinear/nonconvex
- $z = 1$, if $x = y$
- $z = 0$, if $x \neq y$
- x, y integer (channels)

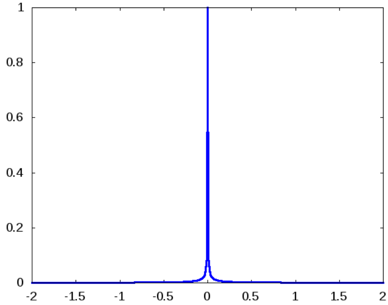


Optimization of IEEE 802.11 Broadband Networks

... and the horrible constraint is ...

$$z = \frac{1}{1 + 1000(x-y)^{10}}$$

- highly nonlinear/nonconvex
- $z = 1$, if $x = y$
- $z = 0$, if $x \neq y$
- x, y integer (channels)
- model as MIP not NLP



Collections of MINLP Test Problems

AMPL Collections of MINLP Test Problems

1. MacMINLP www.mcs.anl.gov/~leyffer/macminlp/
2. IBM/CMU collection egon.cheme.cmu.edu/ibm/page.htm

GAMS Collections of MINLP Test Problems

1. GAMS MINLP-world www.gamsworld.org/minlp/
2. MINLP CyberInfrastructure www.minlp.org/index.php

Solve MINLPs online on the NEOS server,
www.neos-server.org/neos/

... and there are even a few CUTEr problems in SIF!

Summary and Teaching Points

Basic building blocks of solvers

- Relaxation ... create easier to solve problem
- Constraint enforcement ... exclude relaxation solution
- Upper bounds ... prune search space

... more on these concepts over the next three days!

Modeling is very, very, very important

- Linearize, linearize, linearize as much as possible
- Prefer linear over convex and convex over nonconvex
  ... order or magnitude difference in solver capabilities

Beale, E. and Tomlin, J. (1970).
Special facilities in a general mathematical programming system for non- convex problems using ordered sets of variables.
In Lawrence, J., editor, *Proceedings of the 5th International Conference on Operations Research*, pages 447–454, Venice, Italy.

Beale, E. M. L. and Forrest, J. J. H. (1976).
Global optimization using special ordered sets.
Mathematical Programming, 10:52–69.

Grossmann, I. and Lee, S. (2003).
Generalized convex disjunctive programming: Nonlinear convex hull relaxation.
Computational Optimization and Applications, pages 83–100.

Günlük, O. and Linderoth, J. T. (2012).
Perspective reformulation and applications.
In *IMA Volumes*, volume 154, pages 61–92.

Jeroslow, R. G. (1973).
There cannot be any algorithm for integer programming with quadratic constraints.
Operations Research, 21(1):221–224.

Kannan, R. and Monma, C. (1978).
On the computational complexity of integer programming problems.
In Henn, R., Korte, B., and Oettli, W., editors, *Optimization and Operations Research,*, volume 157 of *Lecture Notes in Economics and Mathematical Systems*, pages 161–172. Springer.

Williams, H. P. (1999).
Model Building in Mathematical Programming.
John Wiley & Sons.