Modeling Introduction

Eduardo Camponogara

Department of Automation and Systems Engineering
Federal University of Santa Catarina

October 10^{th}-14^{th}, 2016

Linear Programming

Quadratic Programming

Convex Programming

Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

General Formulation

$$
\begin{array}{ll}
\text{Minimize} & f(x) \\
\text{Subject to :} & \\
 & g(x) \leqslant 0 \\
 & h(x) = 0 \\
 & x \in \mathbb{R}^n
\end{array}
$$

Summary

Linear Programming

Quadratic Programming

Convex Programming

Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

Linear Programming

- Objective function and constraints are all linear (actually affine).
- Mathematically,
  - $f(x) = c^{\mathrm{T}}x$,
  - $g(x) = Ax - a$ e
  - $h(x) = Bx - b$

  in which c, a and b are vectors and A and B are matrices.

Linear Programming

Canonical Form:

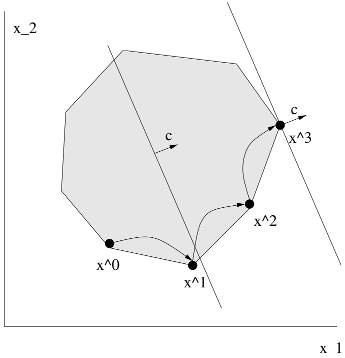
$$\begin{aligned} \max\ & c^{\mathrm{T}} x \\ \mathrm{s.a}:\ & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Problems can be reformulated in the canonical form to deal with:

1. equations/equalities;
2. real variables (which can assume negative values); and
3. minimization rather than maximization.

Geometric View

Find the maximum of a linear function $f(x)$ inside a polyhedron $P = \{x \in \mathbb{R}^n : Ax \leq b\}$.



Under normal conditions, there exists a vertex with the optimal solution $\Longrightarrow$ combinatorial problem.

Linear Programming: Algorithms

Algorithms:

- Algorithms available off-the-shelf: Simplex and Interior-Point.
- Very large problems are solve efficientlhy: dozens of thousands of variables and constraints.

Example

- An enterprise has 4 plants where vehicles are assembled.
- Each plant has demands and limits on man-power, materials and emission to produce assemble a vehicle, as follows:

	man-power	materials	pollution
plant 1	2	3	15
plant 2	3	4	10
plant 3	4	5	9
plant 4	5	6	7

- At least 400 vehicles must be assembled at plant 3.
- 3300 man-hours and 4000 material units are available.
- At most 12000 units of pollutions can be emitted.
- The goal is to maximize the number of vehicles produced.

Example

- An enterprise has 4 plants where vehicles are assembled.
- Each plant has demands and limits on man-power, materials and emission to produce assemble a vehicle, as follows:

	man-power	materials	pollution
plant 1	2	3	15
plant 2	3	4	10
plant 3	4	5	9
plant 4	5	6	7

- At least 400 vehicles must be assembled at plant 3.
- 3300 man-hours and 4000 material units are available.
- At most 12000 units of pollutions can be emitted.
- The goal is to maximize the number of vehicles produced.

Example

- An enterprise has 4 plants where vehicles are assembled.
- Each plant has demands and limits on man-power, materials and emission to produce assemble a vehicle, as follows:

	man-power	materials	pollution
plant 1	2	3	15
plant 2	3	4	10
plant 3	4	5	9
plant 4	5	6	7

- At least 400 vehicles must be assembled at plant 3.
- 3300 man-hours and 4000 material units are available.
- At most 12000 units of pollutions can be emitted.
- The goal is to maximize the number of vehicles produced.

Example

- An enterprise has 4 plants where vehicles are assembled.
- Each plant has demands and limits on man-power, materials and emission to produce assemble a vehicle, as follows:

	man-power	materials	pollution
plant 1	2	3	15
plant 2	3	4	10
plant 3	4	5	9
plant 4	5	6	7

- At least 400 vehicles must be assembled at plant 3.
- 3300 man-hours and 4000 material units are available.
- At most 12000 units of pollutions can be emitted.
- The goal is to maximize the number of vehicles produced.

Example

- An enterprise has 4 plants where vehicles are assembled.
- Each plant has demands and limits on man-power, materials and emission to produce assemble a vehicle, as follows:

	man-power	materials	pollution
plant 1	2	3	15
plant 2	3	4	10
plant 3	4	5	9
plant 4	5	6	7

- At least 400 vehicles must be assembled at plant 3.
- 3300 man-hours and 4000 material units are available.
- At most 12000 units of pollutions can be emitted.
- The goal is to maximize the number of vehicles produced.

Example

1. What are the variables?

   x_1, x_2, x_3, x_4 in which x_i denotes the number of vehicles manufacture at plant i.

2. What is the goal?

   Maximize $x_1 + x_2 + x_3 + x_4$.

3. What are the constraints?

$$
\begin{aligned}
x_i &\geq 0, \forall i, \\
x_3 &\geq 400, \\
2x_1 + 3x_2 + 4x_3 + 5x_4 &\leq 3300, \\
3x_1 + 4x_2 + 5x_3 + 6x_4 &\leq 4000, \\
15x_1 + 10x_2 + 9x_3 + 7x_4 &\leq 12000
\end{aligned}
$$

Example

1. What are the variables?

   x_1, x_2, x_3, x_4 in which x_i denotes the number of vehicles manufacture at plant i.

2. What is the goal?

   Maximize $x_1 + x_2 + x_3 + x_4$.

3. What are the constraints?

$$
\begin{aligned}
x_i &\geq 0, \forall i, \\
x_3 &\geq 400, \\
2x_1 + 3x_2 + 4x_3 + 5x_4 &\leq 3300, \\
3x_1 + 4x_2 + 5x_3 + 6x_4 &\leq 4000, \\
15x_1 + 10x_2 + 9x_3 + 7x_4 &\leq 12000
\end{aligned}
$$

Example

1. What are the variables?

   x_1, x_2, x_3, x_4 in which x_i denotes the number of vehicles manufacture at plant i.

2. What is the goal?

   Maximize $x_1 + x_2 + x_3 + x_4$.

3. What are the constraints?

$$
\begin{aligned}
x_i &\geq 0, \forall i, \\
x_3 &\geq 400, \\
2x_1 + 3x_2 + 4x_3 + 5x_4 &\leq 3300, \\
3x_1 + 4x_2 + 5x_3 + 6x_4 &\leq 4000, \\
15x_1 + 10x_2 + 9x_3 + 7x_4 &\leq 12000
\end{aligned}
$$

Example

1. What are the variables?

   x_1, x_2, x_3, x_4 in which x_i denotes the number of vehicles manufacture at plant i.

2. What is the goal?

   Maximize $x_1 + x_2 + x_3 + x_4$.

3. What are the constraints?

$$
\begin{aligned}
x_i &\geq 0, \forall i, \\
x_3 &\geq 400, \\
2x_1 + 3x_2 + 4x_3 + 5x_4 &\leq 3300, \\
3x_1 + 4x_2 + 5x_3 + 6x_4 &\leq 4000, \\
15x_1 + 10x_2 + 9x_3 + 7x_4 &\leq 12000
\end{aligned}
$$

Example

1. What are the variables?

   x_1, x_2, x_3, x_4 in which x_i denotes the number of vehicles manufacture at plant i.

2. What is the goal?

   Maximize $x_1 + x_2 + x_3 + x_4$.

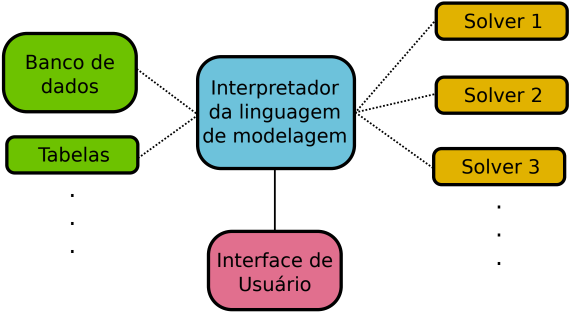
3. What are the constraints?

$$\begin{aligned} x_i &\geq 0, \forall i, \\ x_3 &\geq 400, \\ 2x_1 + 3x_2 + 4x_3 + 5x_4 &\leq 3300, \\ 3x_1 + 4x_2 + 5x_3 + 6x_4 &\leq 4000, \\ 15x_1 + 10x_2 + 9x_3 + 7x_4 &\leq 12000 \end{aligned}$$

AMPL

- AMPL is an algebraic modeling language that allows specification of math-programming problems at a high level.
- AMPL advocates the separation of model and data.
- It is very similar to the way we express optimization problems in math.
- The user is free from data manipulation.

AMPL


Banco de dados
Tabelas
Interpretador da linguagem de modelagem
Solver 1
Solver 2
Solver 3
Interface de Usuário

AMPL

- AMPL needs a model in mathematical programming which states the variables, objective and constraints.
- Further, it needs a data instance.
- The model and data files are fed into an optimization solver.
- AMPL acts as a parser and compiler.

AMPL Example

Implement in AMPL the vehicle assembly problem.

Summary

Linear Programming

Quadratic Programming

Convex Programming

Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

Quadratic Programming

Canonical Formulation:

$$QP: \min \frac{1}{2}x^{\mathrm{T}}Qx + c^{\mathrm{T}}x$$
$$\text{s.t.}: Ax \leq b$$

Remarks:

- Extension to linear programming with a quadratic term.
- $Q \succ 0$ is a positive definite (or semidefinite) matrix.
- Convex Problem: minimize a convex objective within a convex feasible set.

Quadratic Programming: Algorithms

Algorithms:

- Active-set algorithms are efficient for QP.
  Simplex for LP is a particular kind of active-set algorithm.
- Interior-point method is another class of efficient algorithms, with polynomial time.
- Large problems are solved efficiently.

Quadratic Programming: Algorithms

Algorithms:

- Active-set algorithms are efficient for QP.
  Simplex for LP is a particular kind of active-set algorithm.
- Interior-point method is another class of efficient algorithms, with polynomial time.
- Large problems are solved efficiently.

Quadratic Programming: Algorithms

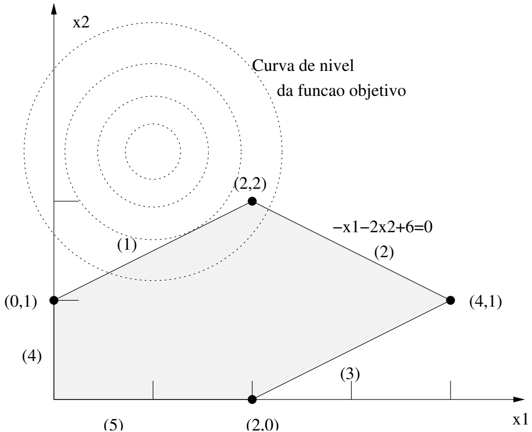
Algorithms:

- Active-set algorithms are efficient for QP.
  Simplex for LP is a particular kind of active-set algorithm.
- Interior-point method is another class of efficient algorithms, with polynomial time.
- Large problems are solved efficiently.

Quadratic Programming: Example

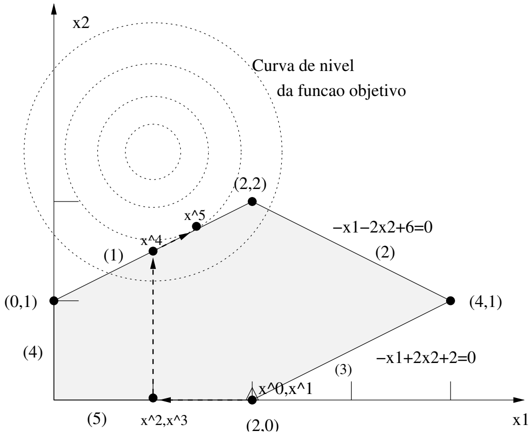
$$
\begin{aligned}
&\min\ q(x) = (x_1 - 1)^2 + (x_2 - 2.5)^2 \\
&\text{s.t.}: \\
&\qquad \begin{array}{rcl}
x_1 - 2x_2 + 2 & \geq & 0 \\
-x_1 - 2x_2 + 6 & \geq & 0 \\
-x_1 + 2x_2 + 2 & \geq & 0 \\
x_1, x_2 & \geq & 0
\end{array}
\end{aligned}
$$

Example


x2
Curva de nivel
da funcao objetivo
(2,2)
−x1−2x2+6=0
(1)
(2)
(0,1)
(4,1)
(4)
(3)
(5)
(2,0)
x1

Example

Behavior of the active-set algorithm.



Summary

Linear Programming

Quadratic Programming

Convex Programming

Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

Convex Problem

It consists of minimizing a convex function within a feasible convex set:

$$\min \ f(x)$$
$$s.t. : x \in \mathcal{X}$$

in which $X \subseteq \mathbb{R}^n$ is a convex set.

Convex Problem: Explicit Representation

Standard Problem:

$$\begin{aligned} \min\ & f_0(x) \\ s.t.:\ & f_i(x) \le 0,\ i = 1, \ldots, m, \\ & a_i^{\mathrm{T}} x = b_i,\ i = 1, \ldots, p \end{aligned}$$

in which $f_i, i = 0, \ldots, m,$ are convex functions.

Notice that
$X = \{x : f_i(x) \le 0,\ i = 1, \ldots, m, a_i^{\mathrm{T}} x = b_i,\ i = 1, \ldots, p\}$ is a convex set.

Convex Problem: Explicit Representation

Standard Problem:

$$\begin{aligned} \min \ & f_0(x) \\ s.t.: \ & f_i(x) \leq 0, \ i = 1, \ldots, m, \\ & a_i^{\mathrm{T}} x = b_i, \ i = 1, \ldots, p \end{aligned}$$

in which $f_i, i = 0, \ldots, m,$ are convex functions.

Notice that
$X = \{x : f_i(x) \leq 0, \ i = 1, \ldots, m, a_i^{\mathrm{T}} x = b_i, \ i = 1, \ldots, p\}$ is a convex set.

Convex Problem: Algorithms

Algorithms:

- Efficient algorithms based on the interior-point method, with polynomial time.
- It is technology that enables the solution of large problems.
- There exist a range of direct application such as in nonlinear programming.

Classification: Support Vector Machines (SVM)

- Let D be training set, in the form:

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p,\ y_i \in \{-1, 1\}\}_{i=1}^n$$

in which y_i indicates the class to which point x_i belongs.

- We want to find a hyperplane $w^{\mathrm{T}}x - b = 0$ of maximum margin that separates the data points labeled with $y_i = -1$ from those labeled with $y_i = +1$.
- The vectors that are on the separating hyperplane are said to be supporting vectors.

Classification: Support Vector Machines (SVM)

- Let D be training set, in the form:

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p,\ y_i \in \{-1, 1\}\}_{i=1}^{n}$$

  in which y_i indicates the class to which point x_i belongs.
- We want to find a hyperplane $w^{\mathrm{T}}x - b = 0$ of maximum margin that separates the data points labeled with $y_i = -1$ from those labeled with $y_i = +1$.
- The vectors that are on the separating hyperplane are said to be supporting vectors.

SVM

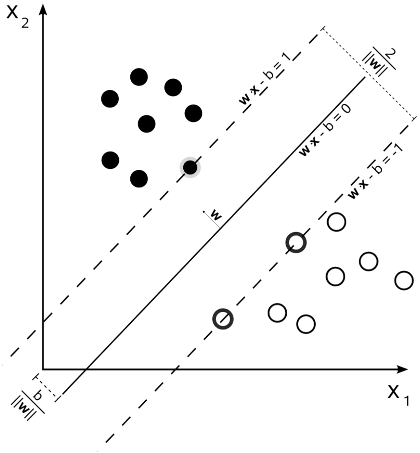
- If the training set D is linearly separable, we can choose two hyperplanes:

$$w^{\mathrm{T}}x - b = 1,$$
$$w^{\mathrm{T}}x - b = -1.$$

such that they separate the data, without any points between them.

- We see to maximize the distance between them, which is given by $2/\|w\|$.

SVM



SVM: Convex Formulation

Formulation:

$$\underset{(w,b)}{\arg\min}\ \frac{1}{2}\|w\|^2$$
$$\text{s.t.}: y_i(w^{\mathrm{T}}x_i - b) \geq 1,\ i = 1, \ldots, n$$

Summary

Linear Programming

Quadratic Programming

Convex Programming

Mixed-Integer Linear Programming

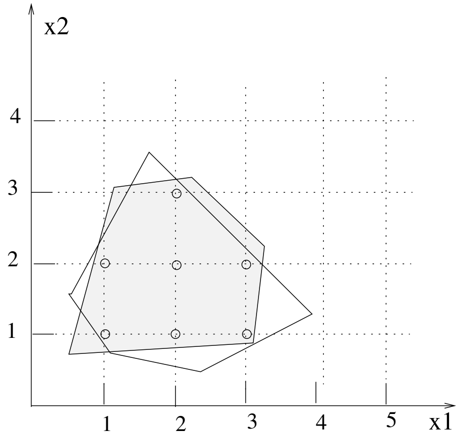
Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

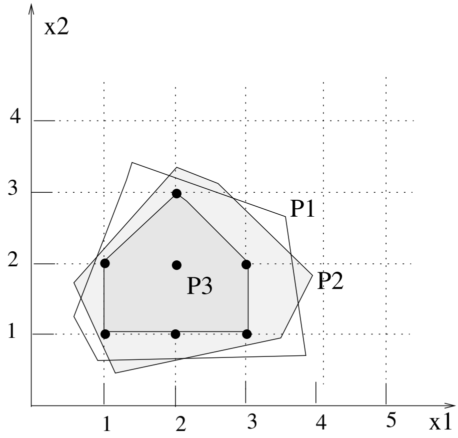
Generalization of linear programming in which variables can be continuous or discrete:

$$\begin{aligned} \max\ & c^{\mathrm{T}} x \\ \mathrm{s.t.}: & Ax \leq b \\ & x = (x_{\mathrm{C}}, x_{\mathrm{I}}) \geq 0, \\ & x_{\mathrm{I}} \in \mathbb{Z}^{n_{\mathrm{I}}} \end{aligned}$$

Geometric View



Ideal Formulation



Mixed-Integer Linear Programming

Properties:

- Highly expressive languages.
- Several problems of academic and industrial relevance can be formulated.
- Coupling to efficient off-the-shelf solvers.
- Mature optimization technology.

Facility Location

Properties:

- Given a number of regions, the problem is to decide where to install a set of emergency-response stations.
- For each station location, it is known the installation cost and the regions that can be served by the station (*e.g.*, in last than 8 minutes).
- Let $M = \{1, \ldots, m\}$ be the set of regions.
- Let $N = \{1, \ldots, n\}$ be the set of locations for emergency-response stations.

Facility Location

Properties:

- Let $S_j \subseteq M$ be the regions that can be served by station j.
- Let c_j be the installation cost of station j.

Facility Location

Problem Formulation:

- We build a matrix A such that $a_{ij} = 1$ if $i \in S_j$ and $a_{ij} = 0$ otherwise.
- Defining variables:

$$x_j = \begin{cases} 1 & \text{if a station } j \text{ will be installed} \\ 0 & \text{otherwise} \end{cases}$$

Facility Location

Defining Constraints:

a) At least one station must serve a region i:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \qquad \text{for } i = 1, \ldots, m$$

b) The variables are binary:

$$x_j \in \{0, 1\} \qquad \text{para } j = 1, \ldots, n$$

Facility Location

Defining the objective:

$$\min \quad \sum_{j=1}^{n} c_j x_j$$

Summary

- Linear Programming
- Quadratic Programming
- Convex Programming
- Mixed-Integer Linear Programming
- Mixed-Integer Linear Programming

Mixed-Integer Linear Programming

$$\begin{aligned}
&\text{Minimize} && f(x) \\
&\text{Subject to :} && \\
& && g(x) \leqslant 0 \\
& && h(x) = 0 \\
& && x = (x_{\mathrm{C}}, x_{\mathrm{I}}) \\
& && x_{\mathrm{C}} \in \mathbb{R}^{n_{\mathrm{C}}} \\
& && x_{\mathrm{I}} \in \mathbb{Z}^{n_{\mathrm{I}}}
\end{aligned}$$

Mixed-Integer Linear Programming

Propriedades:

- One of the most representative classes of mathematical programming.
- Global optimization algorithms are not, in general, efficient.
- Heuristics and exact algorithms are applied in combination. ,
- Need of expert knowledge on methods and problems.

Mixed-Integer Linear Programming

Propriedades:

- One of the most representative classes of mathematical programming.
- Global optimization algorithms are not, in general, efficient.
- Heuristics and exact algorithms are applied in combination. ,
- Need of expert knowledge on methods and problems.

Example: Gas Transportation Network

- Belgium does not produce gas!
- Natural gas is imported from Norway, Holland, and Algeria.
- Gas should be supplied to demand points at the lowest possible cost.
- Gas is injected into the network by compressors.
- There are pressure constraints in the gas pipelines.

Example: Gas Transportation Network

- Belgium does not produce gas!
- Natural gas is imported from Norway, Holland, and Algeria.
- Gas should be supplied to demand points at the lowest possible cost.
- Gas is injected into the network by compressors.
- There are pressure constraints in the gas pipelines.

Example: Gas Transportation Network

- Network (N, A). $A = A_p \cup A_a$.
  - A_a : active arcs model compressors. Flow can increase in the arc.
  - A_p : passive arcs ensure flow conservation.
- $N_s \subseteq N$: set of supplier nodes.
- $c_i,\ i \in N_s$: price of gas unit.
- $\underline{s}_i, \overline{s}_i$: lower and upper bounds on gas supply at node i.
- $\underline{p}_i, \overline{p}_i$: lower and upper bounds for pressure at node i.

Example: Gas Transportation Network

- Network (N, A). $A = A_p \cup A_a$.
  - A_a : active arcs model compressors. Flow can increase in the arc.
  - A_p : passive arcs ensure flow conservation.
- $N_s \subseteq N$: set of supplier nodes.
- c_i, $i \in N_s$: price of gas unit.
- $\underline{s}_i, \overline{s}_i$: lower and upper bounds on gas supply at node i.
- $\underline{p}_i, \overline{p}_i$: lower and upper bounds for pressure at node i.

Example: Gas Transportation Network

- Network (N, A). $A = A_p \cup A_a$.
  - A_a : active arcs model compressors. Flow can increase in the arc.
  - A_p : passive arcs ensure flow conservation.
- $N_s \subseteq N$: set of supplier nodes.
- c_i, $i \in N_s$: price of gas unit.
- $\underline{s}_i, \overline{s}_i$: lower and upper bounds on gas supply at node i.
- $\underline{p}_i, \overline{p}_i$: lower and upper bounds for pressure at node i.

Example: Gas Transportation Network

- $s_i,\ i \in N$: supply of node i:
  - $s_i > 0 \implies$ gas injected into the network at node i.
  - $s_i < 0 \implies$ gas drawn from the network at node i to meet the local demand.
- $f_{ij},\ (i,j) \in A$: flow in arc (i,j):
  - $f(i,j) > 0 \implies$ gas flow $i \to j$.
  - $f(i,j) < 0 \implies$ gas flow $j \to i$.

Example: Gas Transportation Network

- $s_i,\ i \in N$: supply of node i:
  - $s_i > 0 \implies$ gas injected into the network at node i.
  - $s_i < 0 \implies$ gas drawn from the network at node i to meet the local demand.
- $f_{ij},\ (i,j) \in A$: flow in arc (i,j):
  - $f(i,j) > 0 \implies$ gas flow $i \to j$.
  - $f(i,j) < 0 \implies$ gas flow $j \to i$.

Problem Formulation (Conceptual)

$$
\begin{aligned}
\min \; & \sum_{j \in N_s} c_j s_j \\
\text{s.t.}: \; & \sum_{j:(i,j)\in A} f_{ij} - \sum_{j:(j,i)\in A} f_{ji} = s_i, \; \forall i \in N, \\
& \mathrm{sign}(f_{ij}) f_{ij}^2 = \psi_{ij}(p_i^2 - p_j^2), \; \forall (i,j) \in A_p, \\
& \mathrm{sign}(f_{ij}) f_{ij}^2 \geq \psi_{ij}(p_i^2 - p_j^2), \; \forall (i,j) \in A_a, \\
& s_i \in [\underline{s}_i, \overline{s}_i], \; \forall i \in N, \\
& p_i \in [\underline{p}_i, \overline{p}_i], \; \forall i \in N, \\
& f_{ij} \geq 0, \; \forall (i,j) \in A_a
\end{aligned}
$$

Dealing with Function $\text{sign}(\cdot)$

- We can model the function $\text{sign}(\cdot)$ with a binary variable.
- Assuming that $|f_{ij}| \leq F_{ij}$ (constant), then:

$$
\begin{aligned}
f_{ij} &\leq F_{ij} z_{ij}, \\
f_{ij} &\geq -F_{ij}(1 - z_{ij}), \\
\text{sign}(f_{ij}) &= (2z_{ij} - 1), \\
z_{ij} &\in \{0, 1\}
\end{aligned}
$$

which transforms the conceptual problem into a concrete MINLP.

Fundamentals

- Thank you for attending this lecture!!!