Introduction to Optimization, DIKU 2007-08
Monday 3 December
David Pisinger

Program of the day

- Introduction to Integer Programming
- Modeling (Williams, chapter 9)
- Applications: Opencast mining
- Ladies or tigers

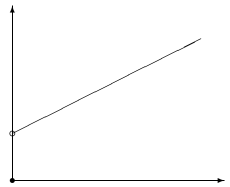
Purpose of the course

- To learn to build complex models from real life using Mathematical Programming
- To know techniques for solving Mathematical Programming models
- To understand that some problems can be solved efficiently and some cannot
- To learn that the same problem may be formulated in different ways, which are easier/harder to solve
- To know a number of techniques for decreasing solution times (or turn a problem from practically "unsolvable" to "solvable")

Integer Programming

In first part of course: continuous variables, linear constraints

- Most products are integral (apart from liquids)
  Airplane production, Tomato Soups
- Structure of problem leads to IP
  Graph problems
- Nonlinear objective functions or constraints occur frequently



- Logical conditions
  "If I use vegetable oil in the blend, then I must also add 5ml of preservatives"

Integer Programming

General formulation:

$$\begin{aligned} \text{maximize} \quad & \sum_{j=1}^{n} c_j x_j \\ \text{subject to} \quad & \sum_{j=1}^{n} a_{1j} x_j \le b_1 \\ & \vdots \\ & \sum_{j=1}^{n} a_{mj} x_j \le b_m \\ & x_j \ge 0, \quad j = 1, \ldots, n, \qquad x \text{ integer} \end{aligned}$$

where

- A is a $m \times n$ matrix
- b is a m-vector
- c is a n-vector

- IP: integer programming model
- ILP: integer *linear* model (all constraints linear)
- PIP: pure integer programming model
- MIP: mixed integer programming model

Integer Programming, why?

- IP is much more expressible than LP
- As IP is NP-hard, all NP-hard problems can be formulated as IP-models

But

- IP is not an ideal model
- Many problems cannot be formulated as IP-models in a simple way
- All NP-complete problems are “equivalent” and hence equally good
- IP proposed by Edmonds: Expressibility and LP-bounds

Integer Programming

IP powerful method for modeling

- LP easy to solve by e.g. Simplex (polynomial time by interior-point methods).
- General IP is NP-hard
- Many concrete problems may be solved despite NP-hardness
- Specific techniques for individual problems

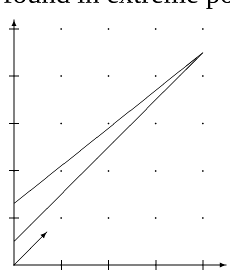
Special problems

- traveling salesman problem
- project selection
- transportation problem
- assignment problem
- assembly line balancing
- set partitioning problem
- aircrew scheduling
- depot location problem
- sequencing problem
- job-shop scheduling

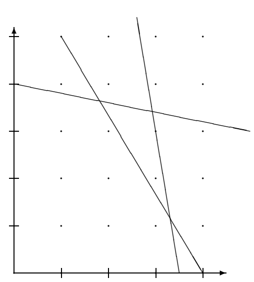
Hardness of IP

$$\begin{array}{ll} \text{maximize} & x_1 + x_2 \\ \text{subject to} & -2x_1 + \quad 2x_2 \geq 1 \\ & -8x_1 + 10x_2 \leq 13 \\ & x_1, x_2 \geq 0, \text{integer} \end{array}$$

Solutions are not found in extreme points (or nearby)



Find convex hull



Model building

- Indicator variables
- Non-convex problems
- Nonlinear functions
- Logical expressions
- Transformation of “human text” to ILP

Indicator variables

- Most important modeling tool!
- $\delta \in \{0,1\}$
- $\delta = 1$ if and only if some event happens.

Model:

$$\delta = 1 \;\Leftrightarrow\; x > 0$$
$$\delta \in \{0,1\},\;\; x \geq 0$$

$\boxed{\delta = 1 \;\Rightarrow\; x > 0}$

$\delta = 1 \Rightarrow x \geq \varepsilon$ $\qquad$ ε level for x regarded as 0

$x - \varepsilon\delta \geq 0,\; \delta \in \{0,1\}$

$\boxed{x > 0 \;\Rightarrow\; \delta = 1}$

$\delta = 0 \;\Rightarrow\; x = 0$

$x - M\delta \leq 0,\; \delta \in \{0,1\}$ $\quad$ M upper bound on x

Indicator variables

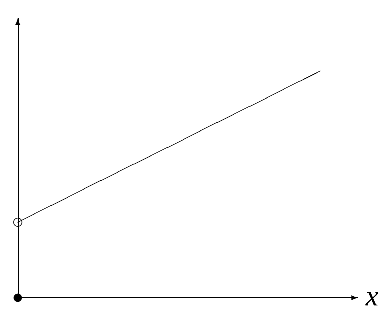
Logical implications $\boxed{X \;\Leftrightarrow\; Y}$

X	Y	$X \Rightarrow Y$	$X \Leftarrow Y$	$\neg X \Rightarrow \neg Y$
T	T	T	T	T
T	F	F	T	T
F	T	T	F	F
F	F	T	T	T

Fixed-charge problem

cost function

$$f(x) = \begin{cases} ax + b & \text{if } \; x > 0 \\ 0 & \text{if } \; x = 0 \end{cases}$$



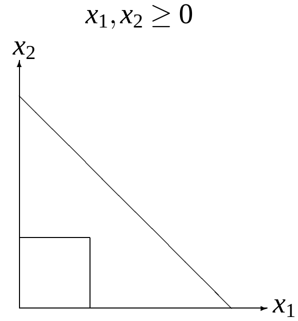
Model:

$$\begin{cases} \text{minimize} & ax + \delta b \\ \text{subject to} & x - M\delta \leq 0 \\ & x - \varepsilon\delta \geq 0 \\ & \delta \in \{0,1\},\;\; x \geq 0 \end{cases}$$

Non-convex problems

constraints:

$$x_1 + x_2 \leq b$$
$$x_1 \geq 1 \text{ or } x_2 \geq 1$$
$$x_1, x_2 \geq 0$$



Modeling tool

$$\delta = 1 \Rightarrow x \geq \varepsilon$$

Two indicator variables δ_1, δ_2:

$$\begin{cases} x_1 + x_2 & \leq & b \\ \delta_1 + \delta_2 & \geq & 1 \\ x_1 - 1\delta_1 & \geq & 0 \\ x_2 - 1\delta_2 & \geq & 0 \\ x_1, x_2 & \geq & 0, \\ \delta_1, \delta_2 & \in & \{0,1\} \end{cases}$$

Indicator variables

"if A is included in the blend then B is included in the blend"

can be modeled by using constraints

$x_A > 0 \;\Rightarrow\; \delta = 1$	$x_A - M\delta \leq 0, \;\; \delta \in \{0,1\}$ M upper bound on x_A
$\delta = 1 \;\Rightarrow\; x_B > 0$	$x_B - \varepsilon\delta \geq 0, \;\; \delta \in \{0,1\}$ ε level for x_B regarded as 0

Example

Assume that x_A and x_B are proportions in blend i.e. $x_A + x_B = 1$.

$$M = 1 \qquad \varepsilon = 0.01$$

Formulation:

$$\begin{cases} x_A - \delta & \leq \; 0 \\ x_B - 0.01\delta & \geq \; 0 \\ \delta & \in \; \{0,1\} \end{cases}$$

Indicator variables for linear inequalities

Example

"If resources needed for production of x_1, x_2 and x_3 are below the limit of one truck, then use the other truck for some other purpose."

General form

$$\boxed{\sum_{j=1}^{n} a_j x_j \leq b \;\;\Leftrightarrow\;\; \delta = 1, \quad \delta \in \{0,1\}}$$

- $\displaystyle\sum_{j=1}^{n} a_j x_j \leq b \Leftarrow \delta = 1$ has the MIP formulation

$$\boxed{\sum_{j=1}^{n} a_j x_j + M\delta \leq M + b}$$

where M is upper bound on $\sum_{j=1}^{n} a_j x_j - b$

$\delta = 1$: $\sum_{j=1}^{n} a_j x_j \leq b$

$\delta = 0$: $\sum_{j=1}^{n} a_j x_j - b \leq M$

Indicator variables for linear inequalities

- $\displaystyle\sum_{j=1}^{n} a_j x_j \leq b \Rightarrow \delta = 1$ has the MIP formulation

$$\boxed{\sum_{j=1}^{n} a_j x_j - (m - \varepsilon)\delta \geq b + \varepsilon}$$

where m is lower bound on $\sum_{j=1}^{n} a_j x_j - b$.

$$\delta = 0 \Rightarrow \sum_{j=1}^{n} a_j x_j \geq b + \varepsilon$$

$\delta = 0$: $\sum_{j=1}^{n} a_j x_j \geq b + \varepsilon$

$\delta = 1$: $\sum_{j=1}^{n} a_j x_j - m + \varepsilon \geq b + \varepsilon$

$\qquad \sum_{j=1}^{n} a_j x_j - b \geq m$

Indicator variables for inequalities, example

Logical condition

$$\begin{array}{l} 2x_1 + 3x_2 \leq 1 \Leftrightarrow \delta = 1 \\ \delta \in \{0,1\} \\ 0 \leq x_1 \leq 1, \;\; 0 \leq x_2 \leq 1 \end{array}$$

We find

$$\begin{array}{rcl} M & = & \text{u.b.}(\sum_{j=1}^{n} a_j x_j - b) \\ & = & \text{u.b.}(2x_1 + 3x_2 - 1) = 4 \end{array}$$

and

$$\begin{array}{rcl} m & = & \text{l.b.}(\sum_{j=1}^{n} a_j x_j - b) \\ & = & \text{l.b.}(2x_1 + 3x_2 - 1) = -1 \end{array}$$

choose $\varepsilon = 0.01$, i.e. constraint broken when $2x_1 + 3x_2 \geq 1.01$

Constraints

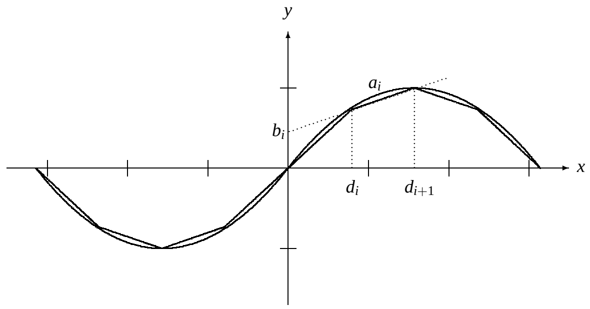
$$\begin{array}{l} 2x_1 + 3x_2 + 4\delta \leq 4 + 1 \\ 2x_1 + 3x_2 - (-1 - 0.01)\delta \geq 1 + 0.01 \end{array}$$

Which results in model:

$$\begin{cases} 2x_1 + 3x_2 + 4\delta \leq 5 \\ 2x_1 + 3x_2 + 1.01\delta \geq 1.01 \\ \delta \in \{0,1\} \\ 0 \leq x_1 \leq 1, \\ 0 \leq x_2 \leq 1 \end{cases}$$

Nonlinear functions

Frequently, the objective function or some of the constraints may contain nonlinear functions.



Approx. nonlinear function by piecewise linear function

- Split into m intervals
- For each interval $[d_i, d_{i+1}]$

$$d_i \le x \le d_{i+1} \Leftrightarrow y = a_i x + b_i$$

- Model as

$$\begin{cases} d_i \le x & \Leftrightarrow \quad \delta_1 = 1 \\ x \le d_{i+1} & \Leftrightarrow \quad \delta_2 = 1 \\ \delta_1 + \delta_2 = 2 & \Leftrightarrow \quad \delta = 1 \\ y = a_i x + b_i & \Leftrightarrow \quad \delta = 1 \end{cases}$$

- Many intervals m, better precision but much harder to solve!

Logical conditions and 0-1 variables

a) *If no depot is sited here then it will not be possible to supply any of the customers from the depot.*

b) *If we manufacture product A then we must also manufacture product B or at least one of products C and D.*

c) *If we do not place an electronic module in this position, then no wires can be connected into this position.*

Introduce an indicator variable $\delta_i \in \{0,1\}$ with each condition X_i

$$\boxed{\text{condition } X_i \text{ is true} \Leftrightarrow \delta_i = 1}$$

In this way we may formulate:

$X_1 \vee X_2$	$\delta_1 + \delta_2 \ge 1$
$X_1 \wedge X_2$	$\delta_1 = 1, \delta_2 = 1$
$X_1 \Rightarrow X_2$	$\delta_1 - \delta_2 \le 0$
$X_1 \Leftrightarrow X_2$	$\delta_1 - \delta_2 = 0$

Transformation to linear form

Write up the text in ordinary mathematical form

$$(\sin(x_1) \le \tfrac{1}{2} \vee x_1 x_2 \le x_3) \Rightarrow (x_3 = 1 \vee x_2 + x_1 \le 1)$$

Stepwise transformation

1 Arithmetic functions are replaced by piecewise linear approximations of the functions.

2 Products of decision variables are transformed into products of binary variables. Products of binary variables may easily be expressed as logical constraints, and thus put on binary form.

3 Relations are transformed into linear inequalities with boolean variables.

$$(ax \le b) \Leftrightarrow (\delta = 1)$$

4 Boolean logics are transformed into linear form.

$$(B_1 \vee B_2) \Leftrightarrow (\delta' = 1)$$

5 The resulting expression should be true $\delta_{all} = 1$

6 Domains of variables are defined.

$$\delta_i \in \{0,1\}$$

Transformation of general constraints to linear form

Step 3:

Relation	ILP-constraints	
$Ax \le b$	$Ax + (\delta - 1)M \le b,$	$Ax + \delta M \ge b + \varepsilon$
$Ax < b$	$Ax + (\delta - 1)M \le b - \varepsilon,$	$Ax + \delta M \ge b$
$Ax > b$	$Ax + (1 - \delta)M \ge b + \varepsilon,$	$Ax - \delta M \le b$
$Ax \ge b$	$Ax + (1 - \delta)M \ge b,$	$Ax - \delta M \le b - \varepsilon$
$Ax = b$	$Ax \ge b \wedge Ax \le b,$	

Step 4:

Relation	Meaning	ILP-constraints
$B_1 \vee B_2$	$\delta = 1 \Leftrightarrow \delta_1 = 1 \vee \delta_2 = 1$	$\delta - \delta_1 - \delta_2 \le 0,\ \delta_1 + \delta_2 - 2\delta \le 0$
$B_1 \wedge B_2$	$\delta = 1 \Leftrightarrow \delta_1 = 1 \wedge \delta_2 = 1$	$2\delta - \delta_1 - \delta_2 \le 0,\ \delta_1 + \delta_2 - \delta \le 1$
$B_1 \Rightarrow B_2$	$\delta = 1 \Leftrightarrow (\delta_1 = 1 \Rightarrow \delta_2 = 1)$	$\delta_1 - \delta_2 + \delta \le 1,\ \delta_1 - \delta_2 + 2\delta \ge 1$
$B_1 \Leftrightarrow B_2$	$\delta = 1 \Leftrightarrow (\delta_1 = 1 \Leftrightarrow \delta_2 = 1)$	use: $(B_1 \Rightarrow B_2) \wedge (B_2 \Rightarrow B_1)$
$\neg B_1$	$\delta = 1 \Leftrightarrow \neg(\delta_1 = 1)$	$\delta = 1 - \delta_1$

Opencast mining (Williams example 12.15)

Concentration c_{ijk} of pure metal for each block

- Level 1 (surface)

1.5	1.5	1.5	0.75
1.5	2.0	1.5	0.75
1.0	1.0	0.75	0.50
1.5	1.5	1.5	0.25

- Level 2 (25 ft depth)

4.0	4.0	2.0
3.0	3.0	1.0
2.0	2.0	0.5

- Level 3 (50 ft depth)

12.0	6.0
5.0	4.0

- Level 4 (75 ft depth)

6.0

Cost of extraction

Level 1: $e_1 = 3.000$ pounds
Level 2: $e_2 = 6.000$ pounds
Level 3: $e_3 = 8.000$ pounds
Level 4: $e_4 = 10.000$ pounds

Revenue from 100% block is $R = 200.000$ pounds

Opencast mining

Introduce variables δ_{ijk}

- Level 1 (surface)

δ_{111}	δ_{112}	δ_{113}	δ_{114}
δ_{121}	δ_{122}	δ_{123}	δ_{124}
δ_{131}	δ_{132}	δ_{133}	δ_{134}
δ_{141}	δ_{142}	δ_{143}	δ_{144}

- Level 2 (25 ft depth)

δ_{211}	δ_{212}	δ_{213}
δ_{221}	δ_{222}	δ_{223}
δ_{231}	δ_{232}	δ_{233}

- Level 3 (50 ft depth)

δ_{311}	δ_{312}
δ_{321}	δ_{322}

- Level 4 (75 ft depth)

δ_{411}

Maximize net profit

$$\sum (R \cdot c_{ijk} - e_i)\delta_{ijk}$$

Constraints

$$\delta_{ijk} = 1 \Rightarrow \begin{cases} \delta_{i-1,j,k} & = 1 \\ \delta_{i-1,j+1,k} & = 1 \\ \delta_{i-1,j,k+1} & = 1 \\ \delta_{i-1,j+1,k+1} & = 1 \end{cases}$$

All variables $\delta_{ijk} \in \{0, 1\}$.

Opencast mining

The constraints

$$\delta_{ijk} = 1 \Rightarrow \begin{cases} \delta_{i-1,j,k} & = 1 \\ \delta_{i-1,j+1,k} & = 1 \\ \delta_{i-1,j,k+1} & = 1 \\ \delta_{i-1,j+1,k+1} & = 1 \end{cases}$$

can be expressed as

$$\begin{aligned} \delta_{ijk} - \delta_{i-1,j,k} & \leq 0 \\ \delta_{ijk} - \delta_{i-1,j+1,k} & \leq 0 \\ \delta_{ijk} - \delta_{i-1,j,k+1} & \leq 0 \\ \delta_{ijk} - \delta_{i-1,j+1,k+1} & \leq 0 \end{aligned}$$

Model with 30 binary variables, 56 constraints
Brute-force solution: 2^{30} steps

Opencast mining

$$\begin{pmatrix} -1 & & & & & 1 & & & & & & \\ & -1 & & & & 1 & & & & & & \\ & & -1 & & & 1 & & & & & & \\ & & & -1 & & 1 & & & & & & \\ & -1 & & & & & 1 & & & & & \\ & & -1 & & & & 1 & & & & & \\ & & & -1 & & & 1 & & & & & \\ & & & & -1 & & 1 & & & & & \\ & & & & & & & & & & & \\ & & & & & & & -1 & & & & 1 \\ & & & & & & & & -1 & & & 1 \\ & & & & & & & & & -1 & & 1 \\ & & & & & & & & & & -1 & 1 \end{pmatrix}$$

The constraint matrix A is totally unimodular (TU)

Solving

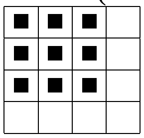
$$\begin{aligned} \max \quad & c\delta \\ \text{s.t.} \quad & A\delta \leq b \\ & \delta \geq 0 \end{aligned}$$

gives integer solutions for any integer vector b and any c

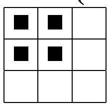
Opencast mining

The optimal solution

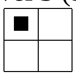
- Level 1 (surface)



- Level 2 (25 ft depth)



- Level 3 (50 ft depth)



- Level 4 (75 ft depth)

Net profit is 17.500 pounds.

Ladies or tigers (Smullyan 1982)

Prisoner in castle, meets nine doors:

- Lady (immediately marry her)
- Tiger (immediately eaten)

Prefers to marry lady than be eaten

Statements on the nine doors are:

Door 1: The lady is in an odd-numbered room
Door 2: This room is empty
Door 3: Either sign 5 is right or sign 7 is wrong
Door 4: Sign 1 is wrong
Door 5: Either sign 2 or sign 4 is right
Door 6: Sign 3 is wrong
Door 7: The lady is not in room 1
Door 8: This room contains a tiger and room 9 is empty
Door 9: This room contains a tiger and sign 6 is wrong

Ladies or tigers (Smullyan 1982)

In addition

- Only one lady
- Each other room: Tiger or empty
- Sign on door of lady is true
- Sign on door of every tiger is false
- Sign on door of empty room can be either true or false.

No unique solution until the prisoner is told whether or not room eight is empty. Then unique solution

Ladies or tigers (Smullyan 1982)

Define subscripts $i = 1, \dots, 9$ and $j = 1, \dots, 3$ where (1 lady, 2 tiger, 3 empty) and as above variables are

$$x_{i,j} = \begin{cases} 1 & \text{if door } i \text{ hides prize } j \\ 0 & \text{otherwise} \end{cases}$$

$$t_i = \begin{cases} 1 & \text{if statement on door } i \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

“Either-or” means “ordinary or”