Aalto University
School of Business

Linear programming (LP) – basic concepts

- *LP: A definition*
- *Graphical solution procedure for LP*
- *Binding and redundant constraints*
- *Special cases: Infeasible and unbounded problems; alternate optima*
- *Computer solution of LPs*
- *Application Examples*

Mathematical programming (optimization) problem

- A math. programming/optimization problem is defined by
  1. decision variables ($x = (x_1, \ldots, x_m)$)
  2. objective function (min $f(x)$ or max $f(x)$)
  3. constraints (equalities $h(x) = c$, or inequalities $g(x) \leq c$)

- A solution = some values for the decision variables
  - A <u>feasible solution</u> satisfies all the problem's constraints
    - The set of all feasible solution is called the <u>feasible region</u>
  - An <u>optimal solution</u> is a feasible solution that results in the best possible value for the objective function
    - The best value =
      - The lowest in minimization problems
      - The highest in maximization problems

Linear Programming (LP) Problem

- If both the objective function and the constraints are linear, the problem is referred to as a linear programming (LP) problem

- Linear functions are functions in which each (decision) variable appears in a separate term raised to the first power and is multiplied by a constant (which could be zero).

- Linear constraints are linear functions that are restricted to be "*less than or equal to*“ (≤), "*equal to*“ (=), or "*greater than or equal to*“ (≥) a constant

- Question: Why do you think strict inequalities (>, <) are not that relevant for LP, or optimization in general? Consider e.g., the problem

$$\max 2x$$
$$x < 1$$

Thus, on this course we only have ≤, ≥, and = constraints

Example: Iron Works, Inc.

- Iron Works, Inc. manufactures two grades of steel and receives *19* kilotons of iron ore per day
  - It takes *2* kt of ore to make one kt of grade 1 steel
  - It takes *3* kt of ore to make one kt of grade 2 steel
  - Facilities allow to produce at most *8* kt of steel daily
  - At most 6 kt of grade 1 steel can be produced daily due to labor restrictions
  - Revenue from one kt of steel is 50000 and 70000 euros for grades 1 and 2, respectively.
- Question:
  - Provide an interpretation of the decision variables (x_1, x_2), constraints and objective function.

LP Formulation

$$\text{Max} \quad 5x_1 + 7x_2$$

Obj. func.

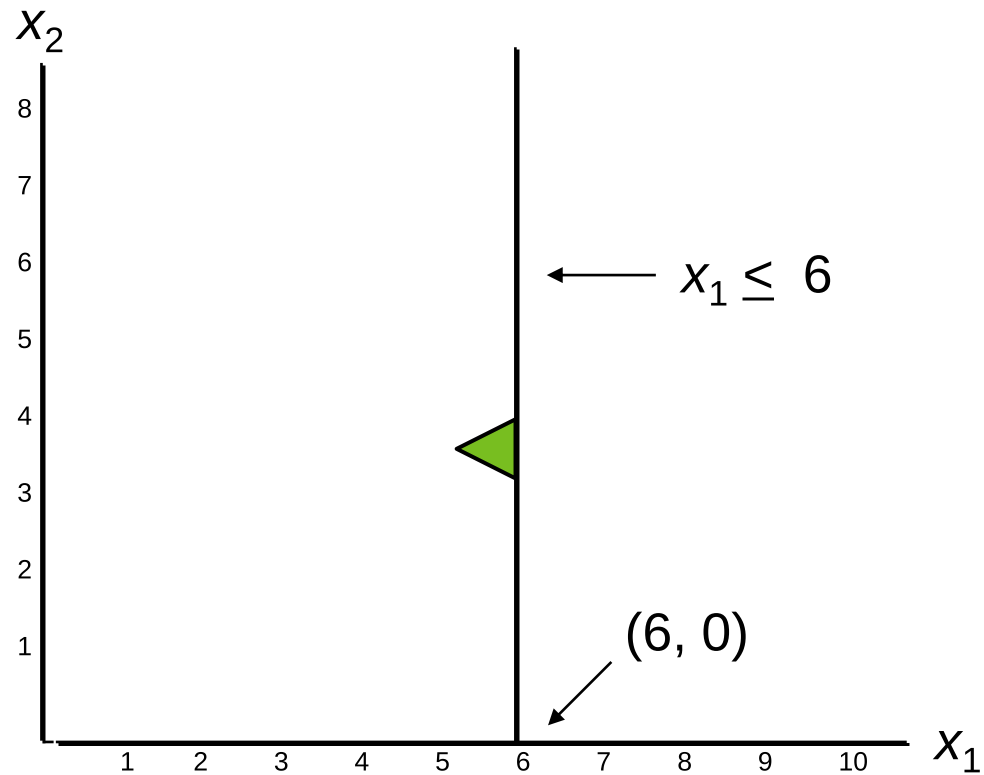
$$\begin{aligned} x_1 &\le 6 \\ 2x_1 + 3x_2 &\le 19 \\ x_1 + x_2 &\le 8 \\ x_1, x_2 &\ge 0 \end{aligned}$$

constraints

Example 1: Graphical Solution

At most 6 tons of grade 1 steel can be produced

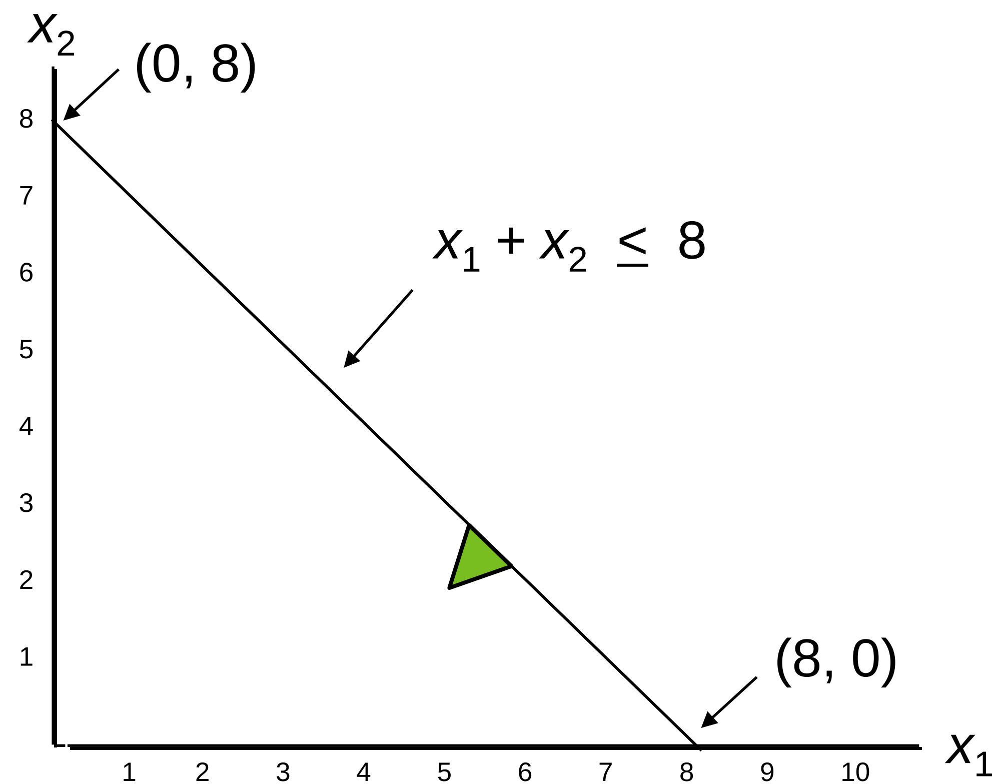
- x_1 = tons of grade 1 steel produced
- x_2 = tons of grade 2 steel produced



Example: Graphical Solution

At most 8 tons of steel can be produced

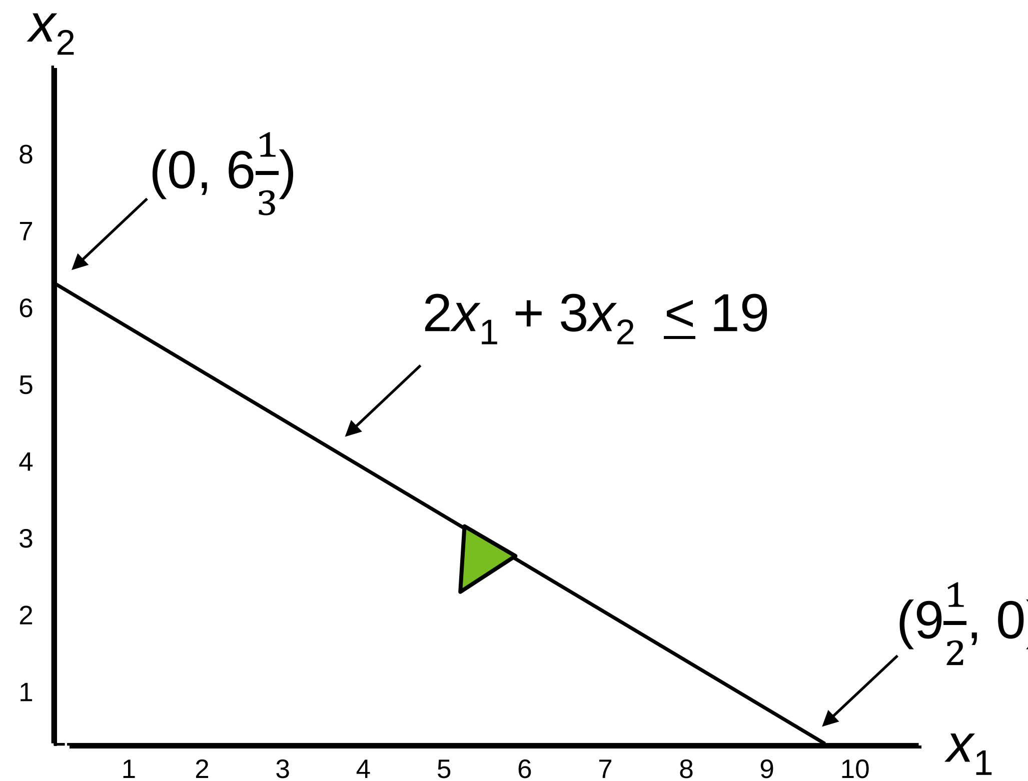
- x_1 = tons of grade 1 steel produced
- x_2 = tons of grade 2 steel produced



Example: Graphical Solution

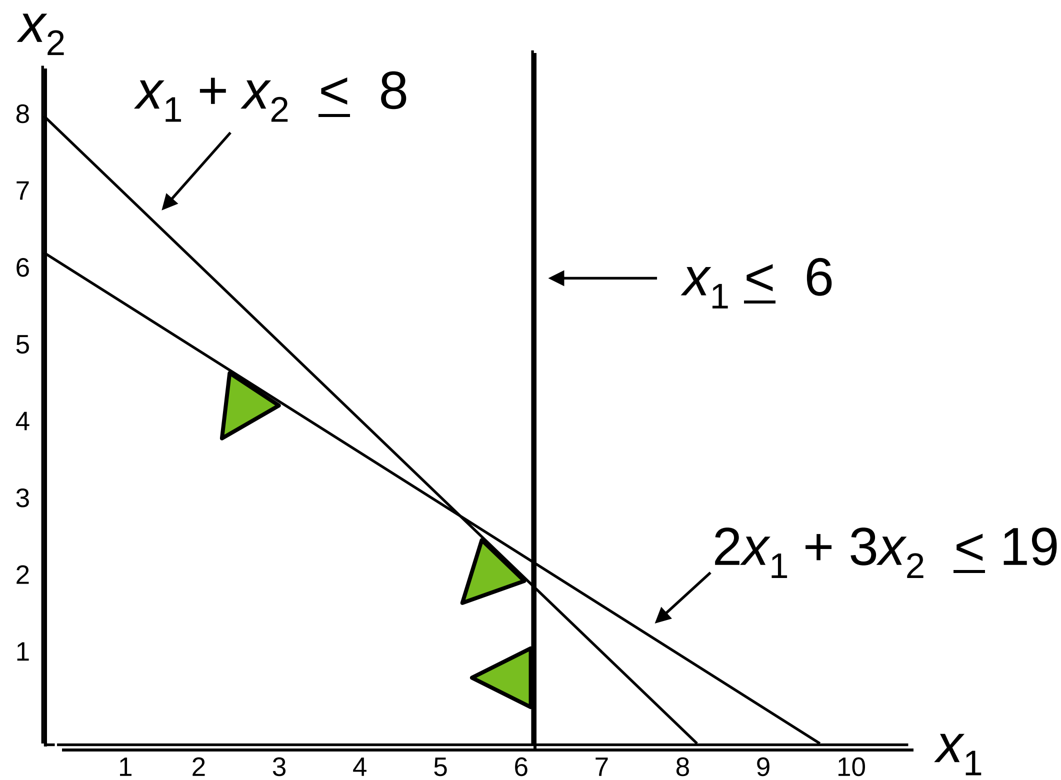
19 tons of iron ore available

- x_1 = tons of grade 1 steel produced
- x_2 = tons of grade 2 steel produced



Example: Graphical Solution

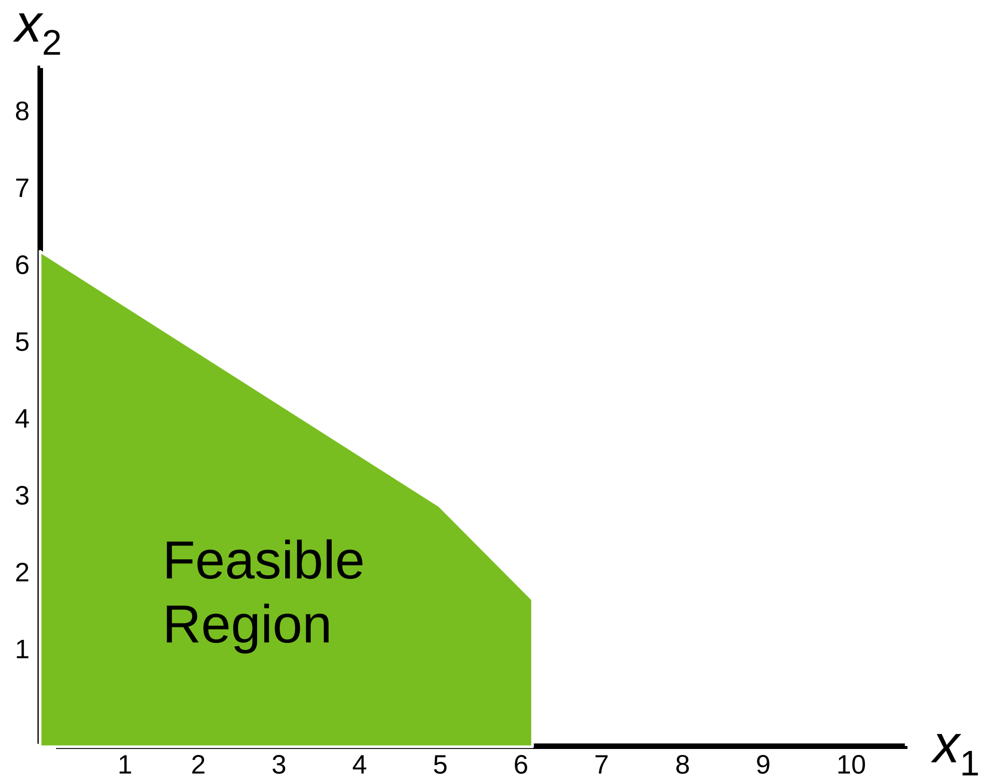
Combined-Constraint Graph



Example: Graphical Solution

feasible region

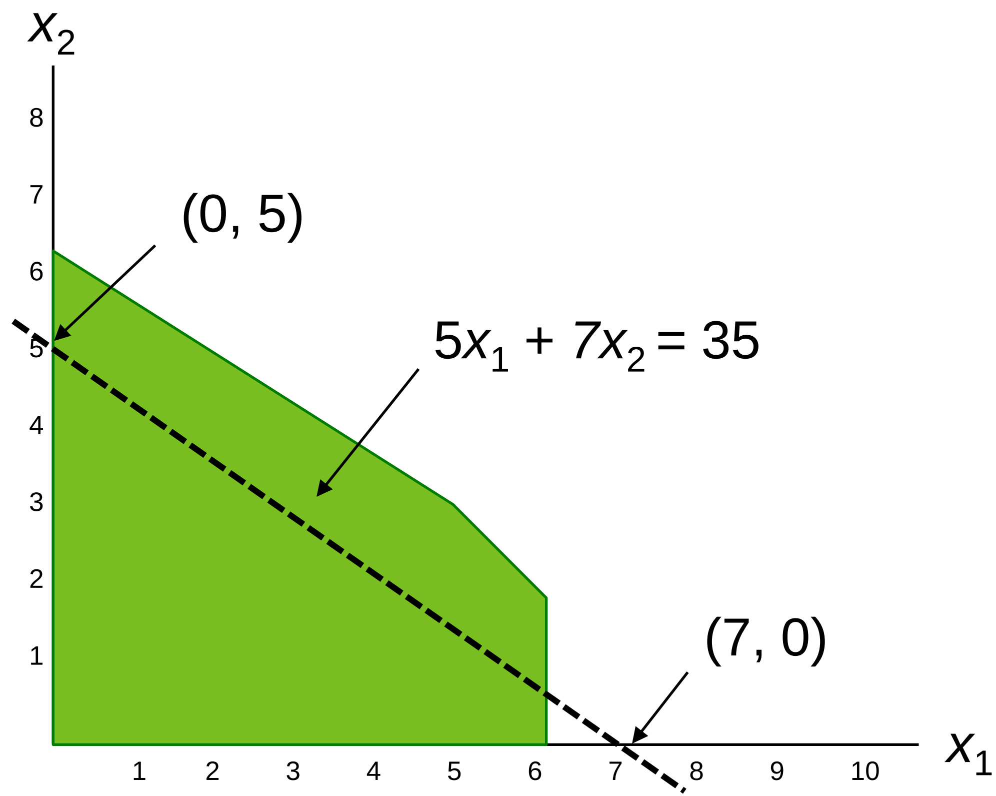
(=the set of feasible solutions)



Example: Graphical Solution

Objective Function Line

- Cf. revenue
  - Grade 1: 50k euros
  - Grade 2: 70k euros



Example: Graphical Solution

<u>Optimal Solution:</u>

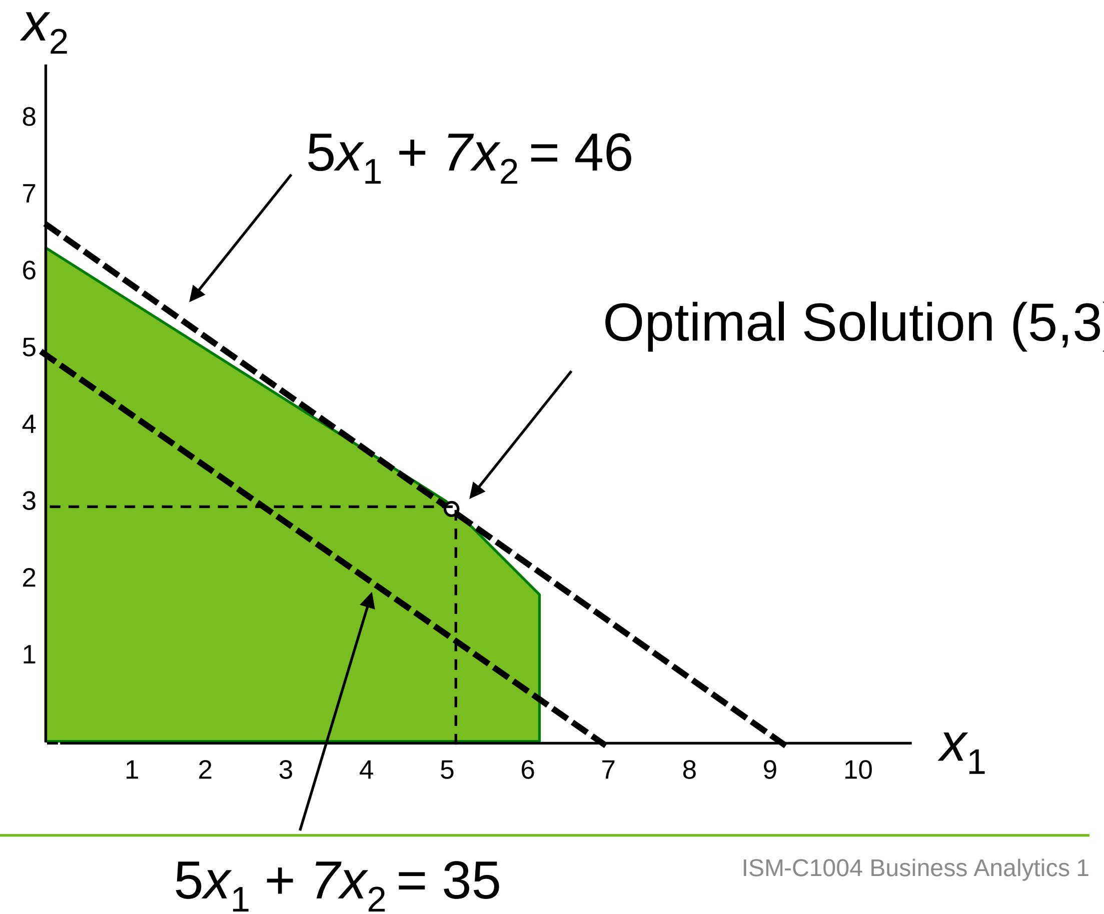
- $x_1 = 5$ tons of grade 1 steel produced
- $x_2 = 3$ tons of grade 2 steel produced

<u>Optimal objective function value:</u>

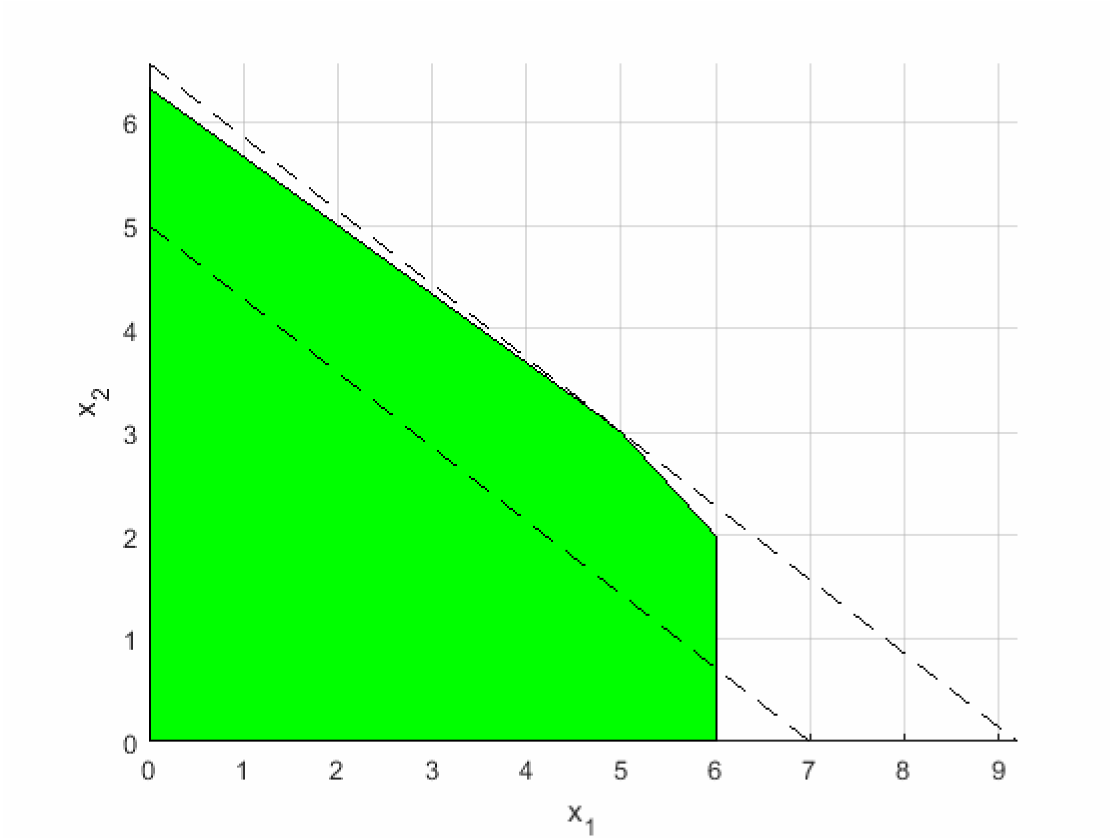
- 460000 euros

<u>Binding constraints</u> at optimum:

- $2x_1 + 3x_2 \leq 19$
- $x_1 + x_2 \leq 8$



Example: Graphical Solution



LP Formulation

$$\text{Max} \quad 5x_1 + 7x_2$$

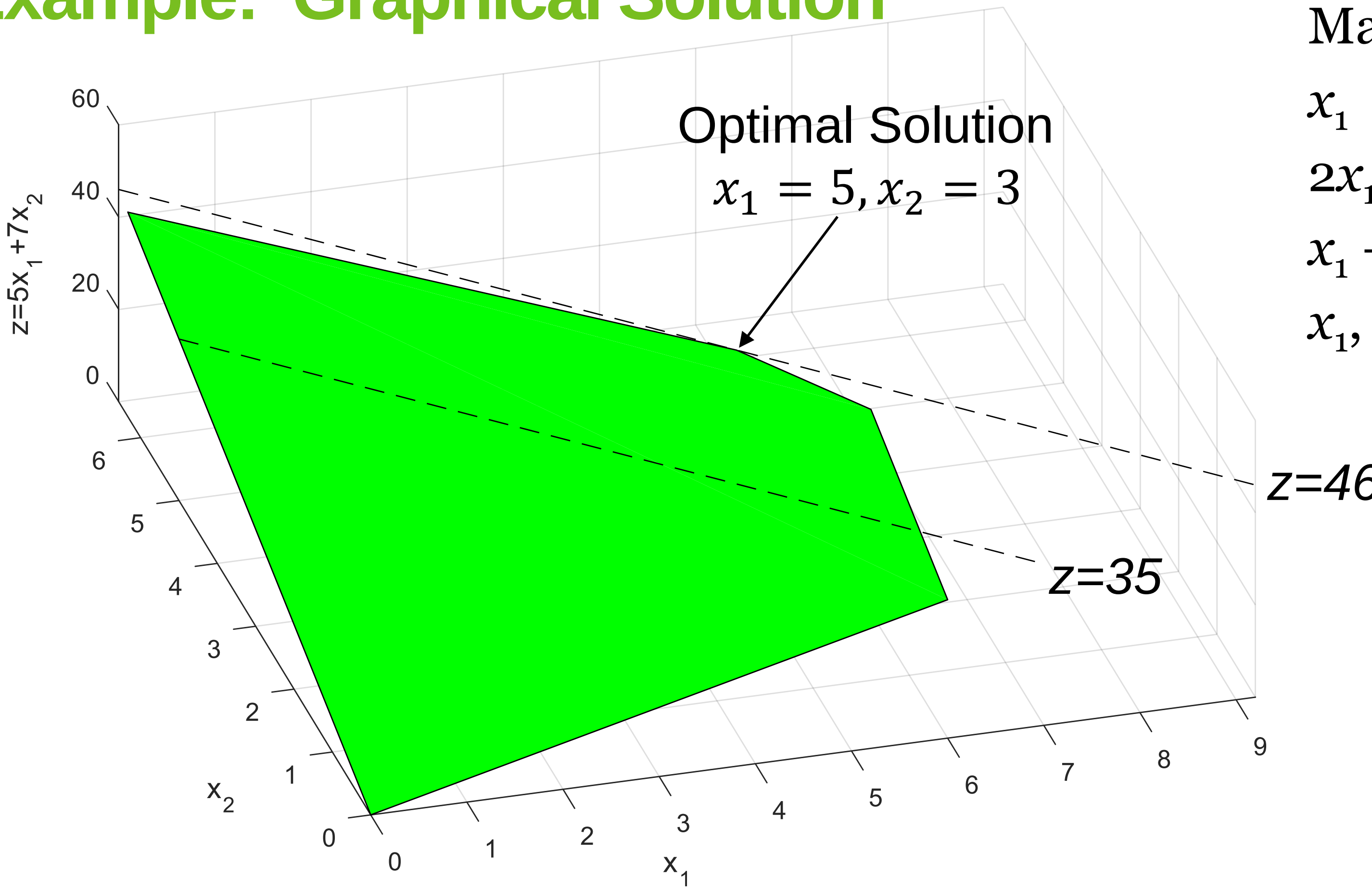
$$x_1 \leq 6$$

$$2x_1 + 3x_2 \leq 19$$

$$x_1 + x_2 \leq 8$$

$$x_1, x_2 \geq 0$$

Example: Graphical Solution



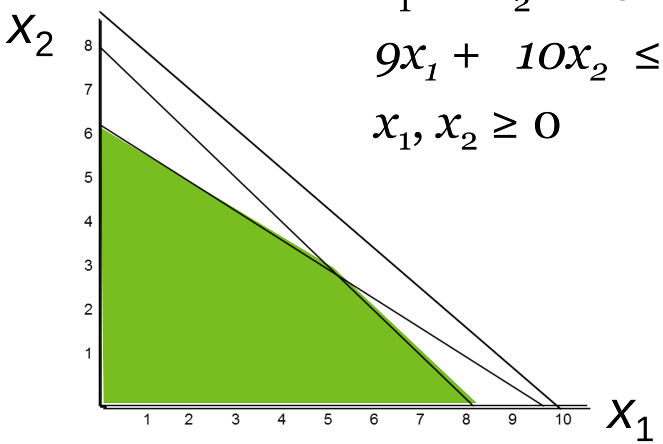
LP Formulation

$$\begin{aligned} \text{Max} \quad & 5x_1 + 7x_2 \\ x_1 \quad & \le 6 \\ 2x_1 + 3x_2 & \le 19 \\ x_1 + x_2 & \le 8 \\ x_1, x_2 & \ge 0 \end{aligned}$$

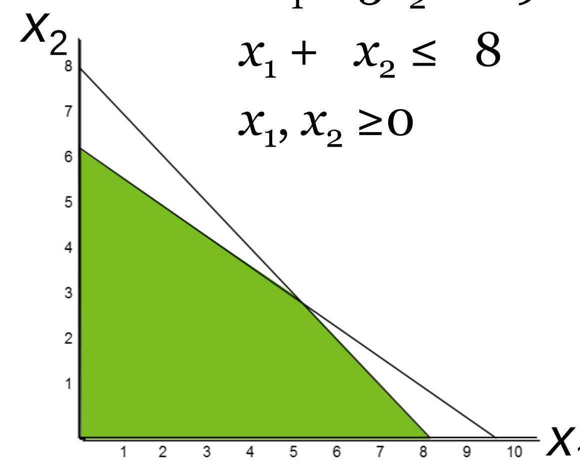
Redundant constraints

- A constraint is **redundant** if removing it does not change the feasible region
- Example:

$$\begin{aligned} 2x_1 + 3x_2 &\leq 19 \\ x_1 + x_2 &\leq 8 \\ 9x_1 + 10x_2 &\leq 90 \\ x_1, x_2 &\geq 0 \end{aligned}$$



$$\begin{aligned} 2x_1 + 3x_2 &\leq 19 \\ x_1 + x_2 &\leq 8 \\ x_1, x_2 &\geq 0 \end{aligned}$$



→ Constraint $9x_1 + 10x_2 < 90$ is redundant!

Example: A Minimization Problem

LP Formulation

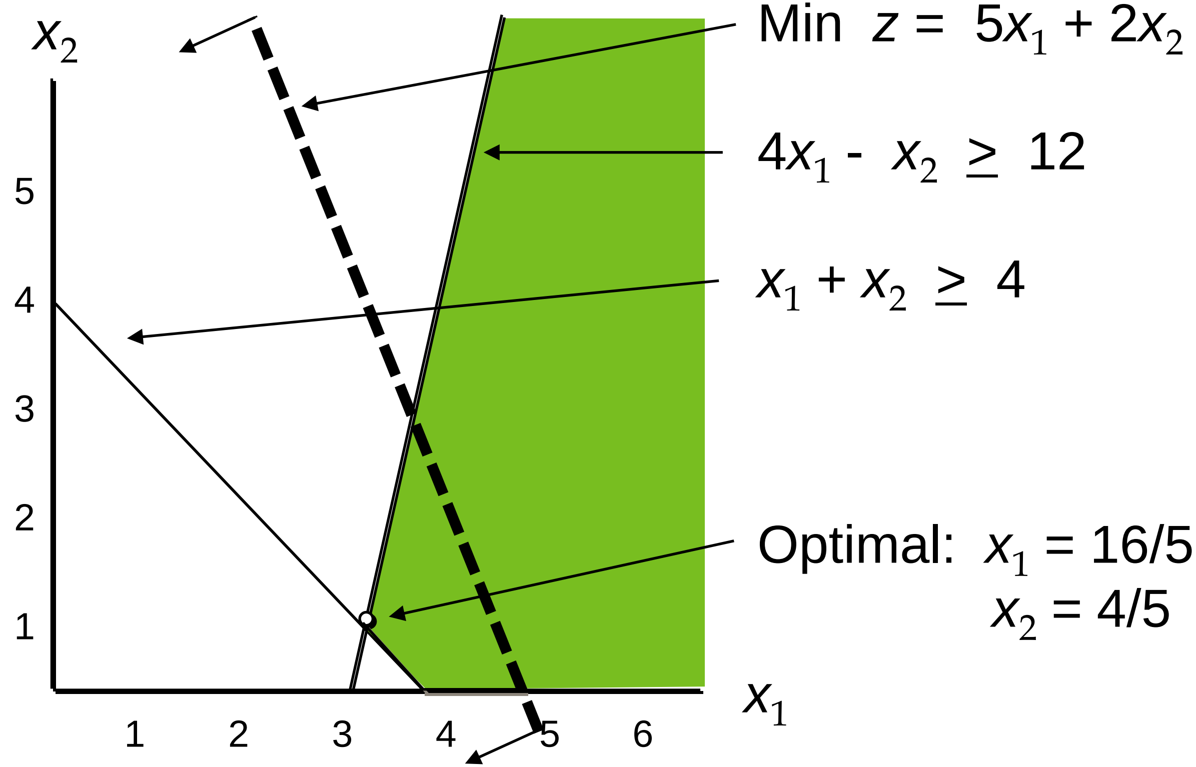
Min $z = 5x_1 + 2x_2$

$4x_1 - x_2 \geq 12$ (1)

$x_1 + x_2 \geq 4$ (2)

$x_1 \geq 0$ (3)

$x_2 \geq 0$ (4)



Question: Which constraints are binding?

Summary of the Graphical Solution Procedure

- Prepare a graph of the feasible solutions for each of the constraints
- Determine the feasible region
  - The set of those solutions that satisfy all the constraints
- Draw an objective function line
  - Move parallel objective function lines toward improved objective function values without entirely leaving the feasible region
  - Any feasible solution on the objective function line with the largest (smallest) value is an optimal solution

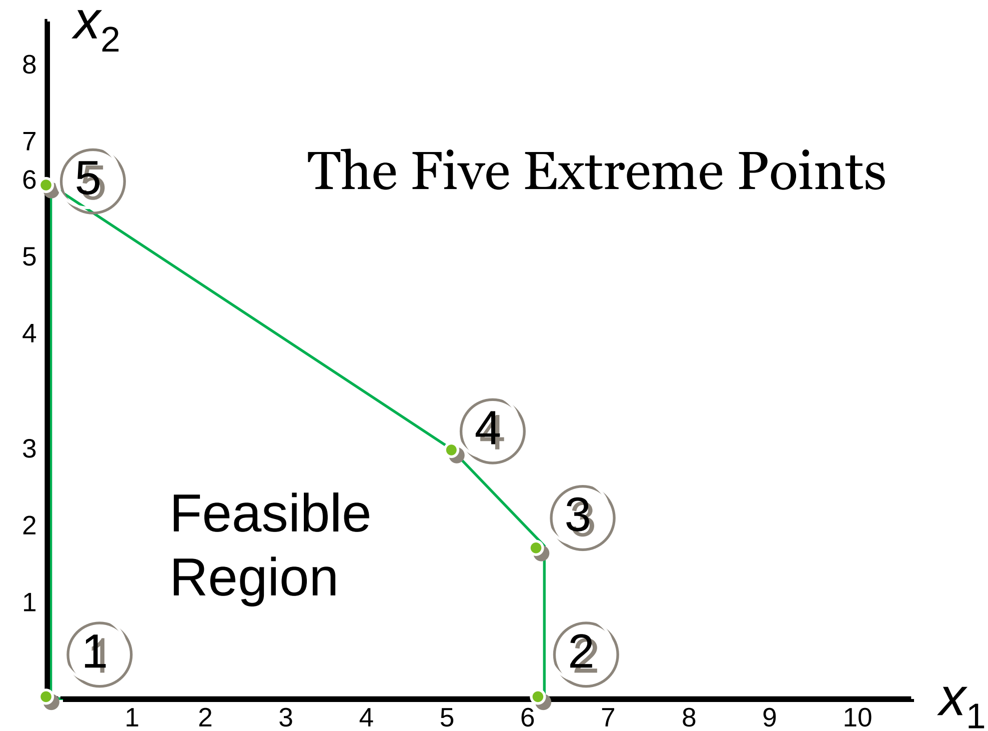
- Another example on the graphical solution procedure can be found at the end of this slide set

Extreme Points and the Optimal Solution

- The corners of the feasible region are referred to as the <u>extreme points</u>.
- At least one of the extreme points is an optimal solution*

→An alternative (graphical) solution method:

  - Compute the objective function value in each extreme point
  - The extreme point with the highest objective function value is an optimal solution



*) as long as the problem has an optimal solution (cf. unbounded and infeasible problems introduced soon)

LP special cases

- Each LP problem falls into one of the three categories:

1. The problem has one or more optimal solutions
   - Several alternative optimal solutions exists if all points of a line segment between two extreme points yield the optimal objective function value

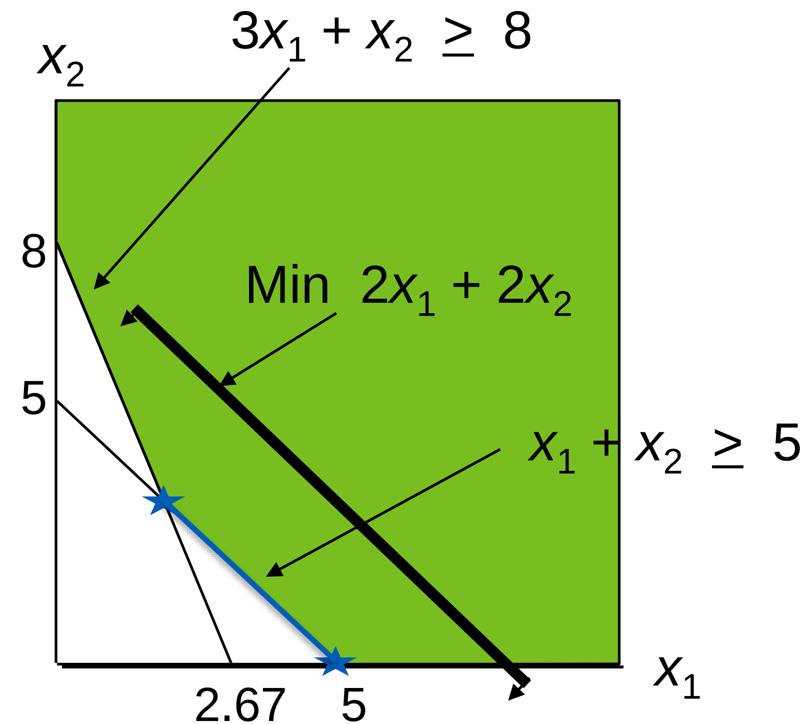
2. The problem is infeasible
   - An over constrained LP with no point that satisfies all the constraints (i.e., the feasible region is empty)

3. The problem is unbounded
   - The objective function value can be improved without a bound in the feasible region

Example: Alternate optimal solutions

$$\text{Min} \quad z = 2x_1 + 2x_2$$
$$x_1 + x_2 \geq 5$$
$$3x_1 + x_2 \geq 8$$
$$x_1, x_2 \geq 0$$

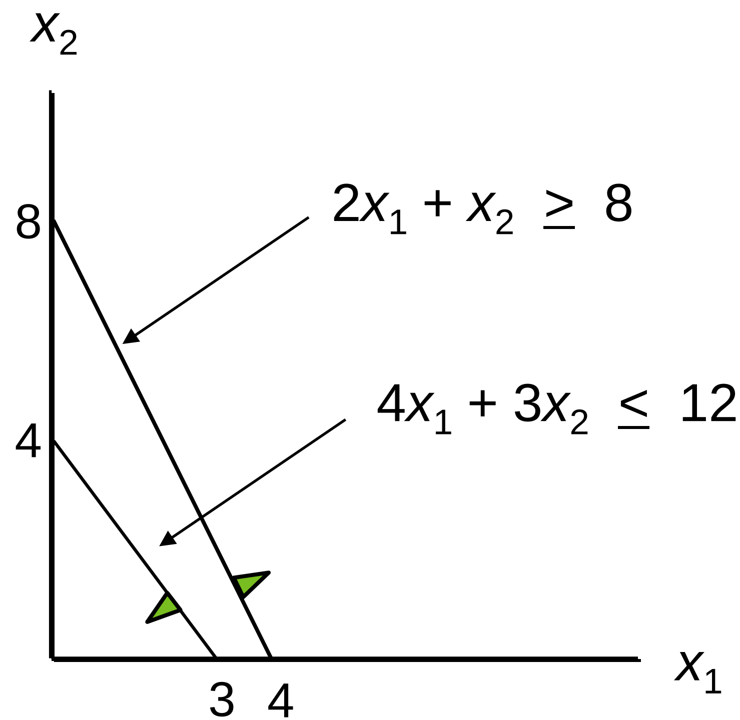
- The objective function line is parallel to a boundary constraint in the direction of optimization
- The points (5,0) and (1.5 , 3.5) and all points on the line segment in between are optimal
  - Objective function value is 10



Example: An Infeasible Problem

$$\text{Max} \quad z = 2x_1 + 6x_2$$
$$4x_1 + 3x_2 \leq 12$$
$$2x_1 + x_2 \geq 8$$
$$x_1, x_2 \geq 0$$

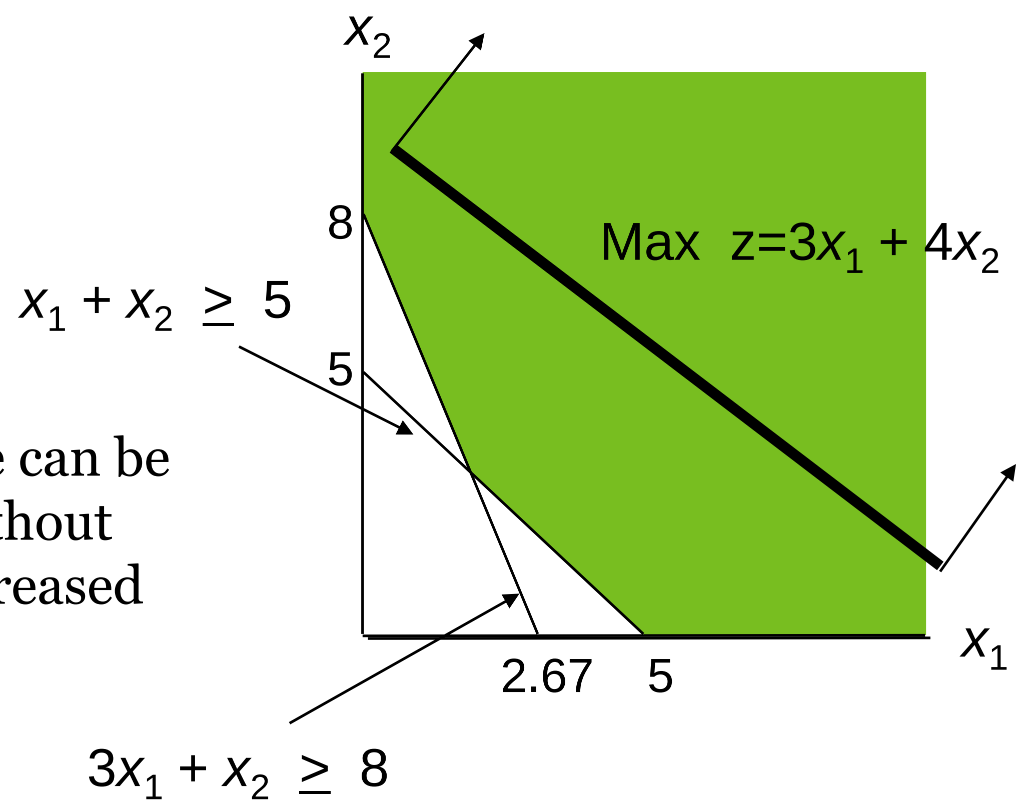
- There are no points that satisfy all constraints
- Hence, this problem has
  - An empty feasible region
  - No feasible solution
  - No optimal solution



Example: An Unbounded Problem

$$\text{Max} \quad z = 3x_1 + 4x_2$$
$$x_1 + x_2 \geq 5$$
$$3x_1 + x_2 \geq 8$$
$$x_1, x_2 \geq 0$$

- The objective function line can be moved parallel to itself without bound so that z can be increased infinitely



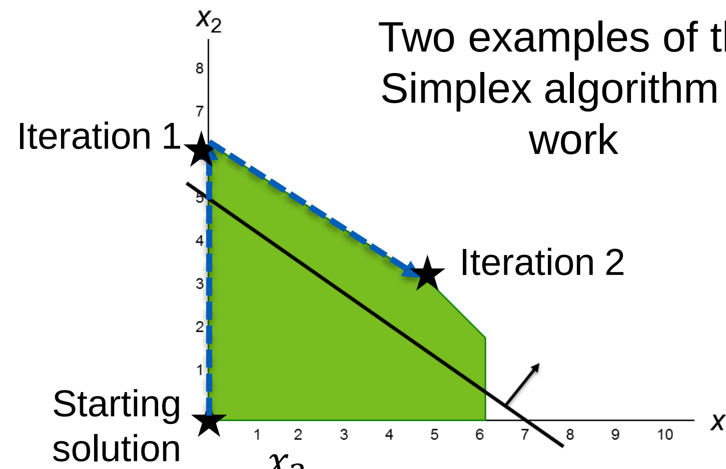
Note: Changing max to min would make the problem bounded (see two slides back).
This shows that in general, even if the feasible region contains solutions with infinitely large/small decision variable values, it does not necessarily mean that the LP problem is unbounded.

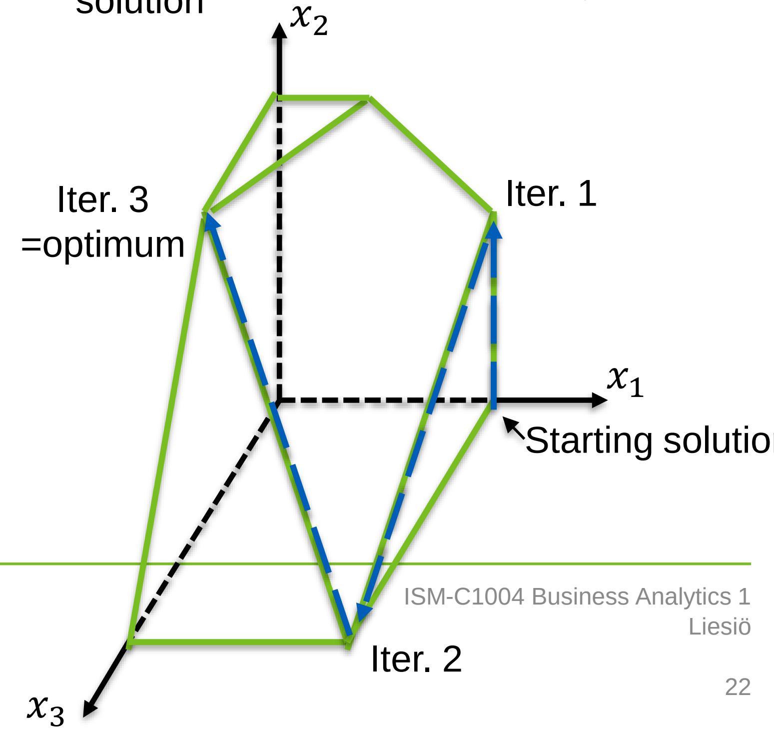
The Simplex Algorithm

- Solution Algorithms for LP
  - **Simplex** (G.B. Dantzig, 1947)
  - Interior point methods (Karmarkar, 1984)

- Simplex (of its modification) is implemented in many software
  - **Excel Solver**
  - lp_solve (open source)
  - CPLEX
  - Gurobi
  - XPress

Two examples of the Simplex algorithm at work





Excel solver

LP Formulation

$$\text{Max} \quad 5x_1 + 7x_2$$

$$x_1 \le 6$$

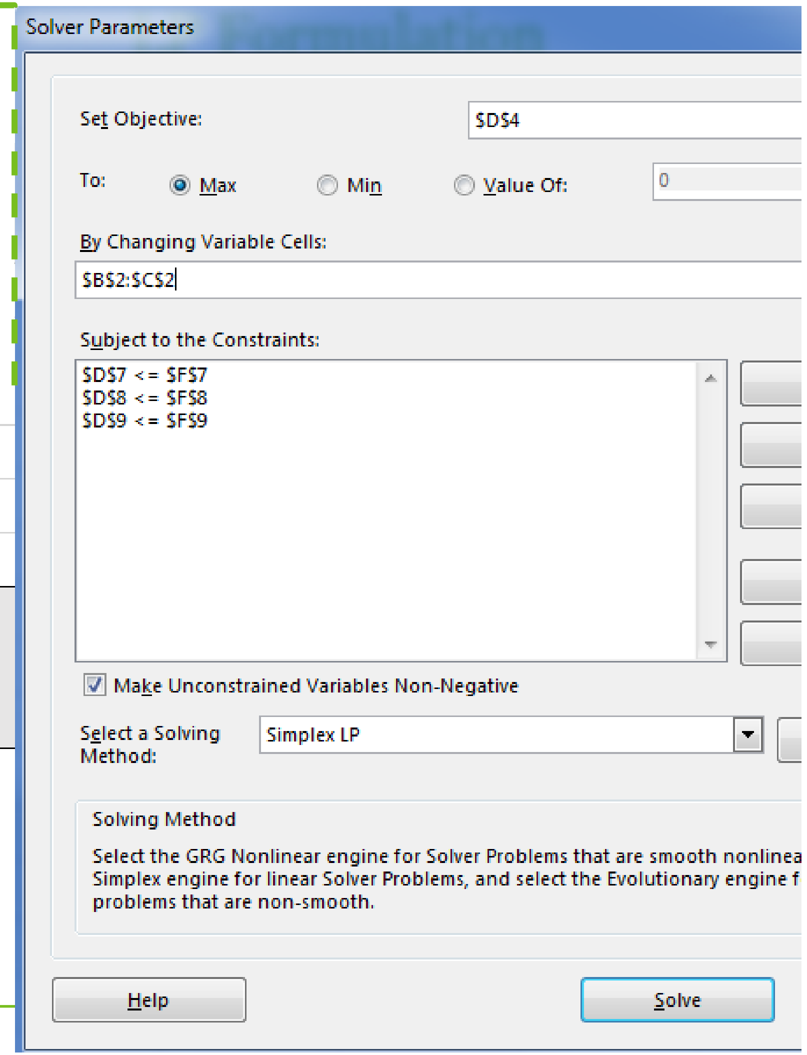
$$2x_1 + 3x_2 \le 19$$

$$x_1 + x_2 \le 8$$

$$x_1, x_2 \ge 0$$

D4 | =SUMPRODUCT(B$2:C$2;B4:C4)

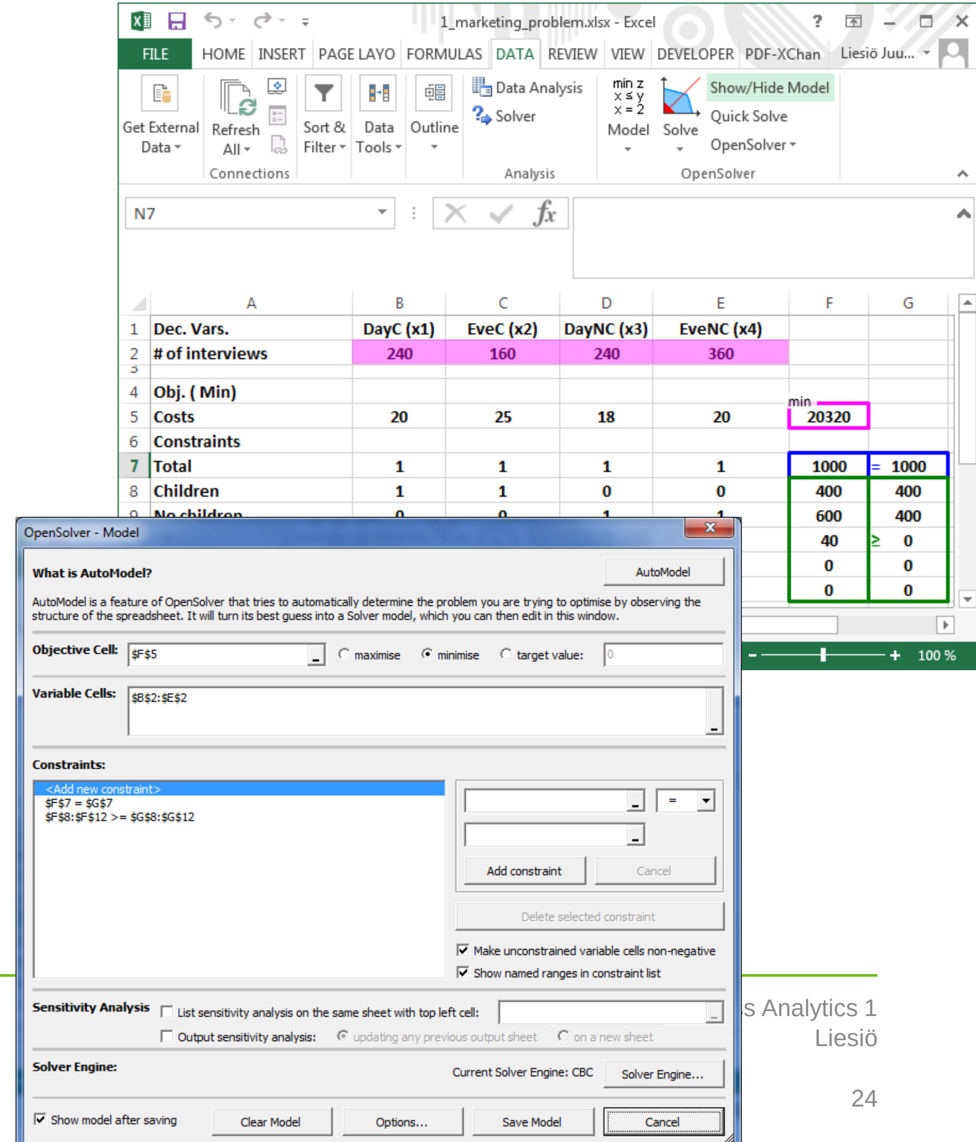
	A	B	C	D	E	F
1		**x1**	**x2**			
2	Decision variable	5	3			
3						
4	Objective f. coefficients	5	7	46		
5						
6						
7	Constraint #1	1	0	5	<=	6
8	Constraint #2	2	3	19	<=	19
9	Constraint #3	1	1	8	<=	8
10						



- Details of building and solving the model can be found at the end of this slide set

Open solver

- Excel Solver is not particularly powerful
  - Difficult to solve
    - large problems (i.e., any constrains/ decision variables)
    - integer LP problems (covered later)
  - Mac versions have bugs
- Open solver offers a free alternative for Win and Mac
  - Download: http://opensolver.org/
  - More powerful solution algorithm
  - Better user interface
  - This also has some bugs…



Guidelines for Building Spreadsheet Models

- All data should be visible and labeled
  - Organize and clearly identify the data
    - Use Borders, shading, and/or colors to distinguish cell types (data/parameters, decision variables, formulas, the objective function)
  - Enter each piece of data into one cell only
    - This makes the model much easier to modify later

- Show entire model on the spreadsheet
  - Avoid putting numbers directly in formulas
  - Break out complicated formulas into subtotals
  - All constraints should be on the spreadsheet (not buried in Solver)

- Try to enter the formula just once, and then use Excel's 'fill' capability.
  - This makes the model easer to build and reduces typos

Formulas in Excel that result in a LP model

Problem parameters (data) located in cells D1:D6

Decision variables located in cells C1:C6.

SUMPRODUCT(D4:D6, C4:C6)
[(D1 + D2) / D3] * C4
SUM(D4:D6)
2*C1 + 3*C4 + C6
C1 + C2 + C3

➡ **LP model**

SUMPRODUCT(C4:C6, C1:C3)
[(C1 + C2) / C3] * D4
ABS(C1)
SQRT(C1)
C1 * C2
C1 / C2
C1 ^2

➡ **NOT a LP model**

A function that is linear in the decision variables x_i: $f(x) = \sum_{i=1}^{n} a_i x_i + b$

Example: Marketing Research

- Market Survey Inc. has been asked to conduct 1000 interviews to find out how consumers react to a new household product
- The client has also given the following guidelines:
  - Interview at least 400 households with children
  - Interview at least 400 households with no children
  - At least as many interviews in the evening as during the day
  - Conduct at least 40% of the children household interviews in the evening
  - Conduct at least 60% of the no children household interviews in the evening

Household	Interview Cost	
	Day	*Evening*
Children	\$20	\$25
No Children	\$18	\$20

Example: Marketing Research (Cont'd)

Household	Interview Cost	
	Day	*Evening*
Children	$20	$25
No Children	$18	$20

- Interview at least 400 households with children
- Interview at least 400 households with no children
- At least as many interviews in the evening as during the day
- Conduct at least 40% of the children household interviews in the evening
- Conduct at least 60% of the no children household interviews in the evening

$$\min 20x_1 + 25x_2 + 18x_3 + 20x_4$$

$$x_1 + x_2 + x_3 + x_4 = 1000$$
$$x_1 + x_2 \geq 400$$
$$x_3 + x_4 \geq 400$$
$$x_2 + x_4 \geq x_1 + x_3$$
$$x_2 \geq 0.4(x_1 + x_2)$$
$$x_4 \geq 0.6(x_3 + x_4)$$

Question:

- A summer trainee developed an LP model to optimize the interview plan. Help the management to understand the model:
  - What do the decision variables represent?
  - What does the objective function measure?
  - What requirements do the constraint capture?

Example: Marketing Research (Cont'd)

- Simplified LP model and its spreadsheet implementation

$$\min 20x_1 + 25x_2 + 18x_3 + 20x_4$$

$$x_1 + x_2 + x_3 + x_4 = 1000$$

$$x_1 + x_2 \geq 400$$
$$x_3 + x_4 \geq 400$$

$$x_2 + x_4 - x_1 - x_3 \geq 0$$

$$-0.4x_1 + 0.6x_2 \geq 0$$

$$-0.6x_3 + 0.4x_4 \geq 0$$

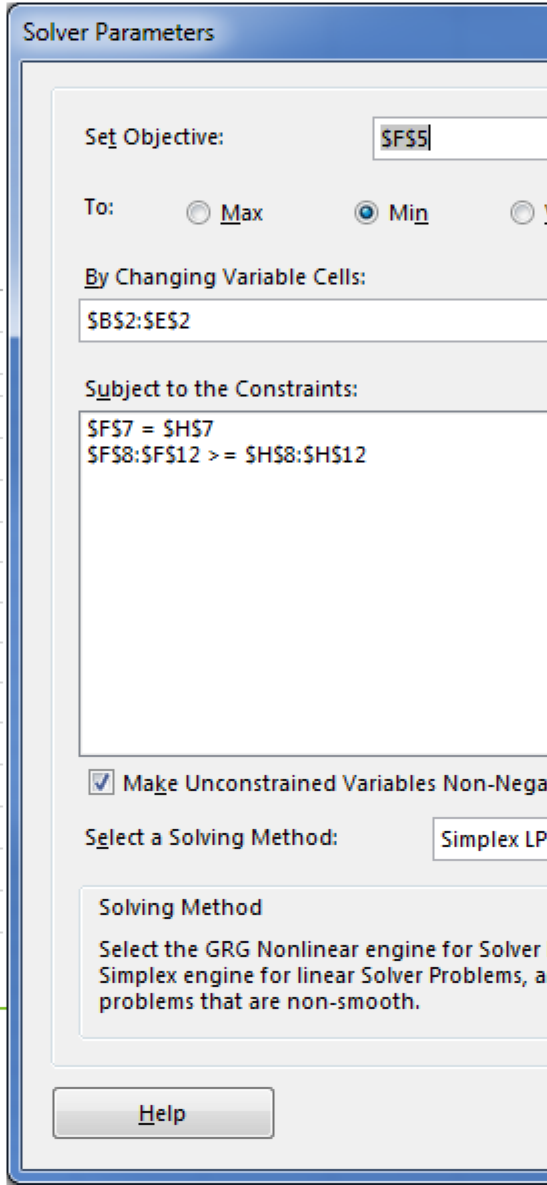
F7 =SUMPRODUCT(B7:E7;$B$2:$E$2)

	A	B	C	D	E	F	G	H
1	Dec. Vars.	DayC (x1)	EveC (x2)	DayNC (x3)	EveNC (x4)			
2	# of interviews	700	0	0	300			
3								
4	Obj. (Min)							
5	Costs	20	25	18	20	20000		
6	Constraints							
7	Total	1	1	1	1	1000	=	1000
8	Children	1	1	0	0	700	>=	400
9	No children	0	0	1	1	300	>=	400
10	Evening	-1	1	-1	1	-400	>=	0
11	Children in evening	-0.4	0.6	0	0	-280	>=	0
12	No chidren in evening	0	0	-0.6	0.4	120	>=	0
13								
14	Decision variables							
15	Parameters							
16	Formulas							

Example: Marketing Research (Cont'd)

- Solver settings and optimal solution

	A	B	C	D	E	F	G	H
1	Dec. Vars.	DayC (x1)	EveC (x2)	DayNC (x3)	EveNC (x4)			
2	# of interviews	240	160	240	360			
3								
4	Obj. (Min)							
5	Costs	20	25	18	20	20320		
6	Constraints							
7	Total	1	1	1	1	1000	=	1000
8	Children	1	1	0	0	400	>=	400
9	No children	0	0	1	1	600	>=	400
10	Evening	-1	1	-1	1	40	>=	0
11	Children in evening	-0.4	0.6	0	0	0	>=	0
12	No chidren in evening	0	0	-0.6	0.4	0	>=	0
13								
14	Decision variables							
15	Parameters							
16	Formulas							
17								



Other properties of LP problems

- Optimal solution will not change if
  - A constant is added to the objective function
  - The objective function is multiplied with a positive constant
  - The objective function (i) is multiplied with a negative constant **and** (ii) min (max) is changed to max (min)
- Any problem can be formulated such that it only has one of the constraint types =, ≥ and ≤
- Adding a constraint can never improve the objective function value

$$\begin{aligned} \min \quad & z = -5x_1 - 7x_2 \\ & 2x_1 + 3x_2 = 19 \\ & x_1 + x_2 \le 8 \end{aligned}$$

same optimal solution (x_1, x_2)

$$\begin{aligned} \max \quad & y = 50x_1 + 70x_2 - 500 \\ & -2x_1 - 3x_2 \le -19 \\ & 2x_1 + 3x_2 \le 19 \\ & x_1 + x_2 \le 8 \end{aligned}$$

optimal w ≥ optimal y

$$\begin{aligned} \max \quad & w = 50x_1 + 70x_2 - 500 \\ & 2x_1 + 3x_2 \le 19 \\ & x_1 + x_2 \le 8 \end{aligned}$$

Slack and Surplus Variables

- A linear program in which all the variables are non-negative and all the constraints are equalities is said to be in <u>standard form</u>
  - Standard form can be obtained by adding <u>slack variables</u> to ≤-constraints, and by subtracting <u>surplus variables</u> from ≥-constraints
  - Slack/surplus variables
    - Represent the difference between the left and right sides of the constraints.
    - Have objective function coefficients equal to zero

- An LP Formulation

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 7x_2 \\
\text{s.t.} \quad & x_1 \le 6 \\
& 2x_1 + 3x_2 \le 19 \\
& x_1 + x_2 \le 8 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Formulation in Standard Form

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 7x_2 + 0s_1 + 0s_2 + 0s_3 \\
\text{s.t.} \quad & x_1 + s_1 = 6 \\
& 2x_1 + 3x_2 + s_2 = 19 \\
& x_1 + x_2 + s_3 = 8 \\
& x_1, x_2, s_1, s_2, s_3 \ge 0
\end{aligned}
$$

Uses of additional decision variables

- Decision variables often have an explicit connection to the real-life decisions, e.g.,
  - x_1: number of houses with children interviewed during day
  - x_2: kilotons of steel to produce

- Sometimes the link is implicit, e.g.,
  - Value of decision variable s_1 depends on how much we decide to produce grade 1 steal, i.e., the value of x_1

- Use of such auxiliary decision variables can make it easier to apply LP in more complex problems (cf. example on next slide)

- An LP Formulation

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 7x_2 \\
\text{s.t.} \quad & x_1 \leq 6 \\
& 2x_1 + 3x_2 \leq 19 \\
& x_1 + x_2 \leq 8 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Formulation in Standard Form

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 7x_2 + 0s_1 + 0s_2 + 0s_3 \\
\text{s.t.} \quad & x_1 + s_1 = 6 \\
& 2x_1 + 3x_2 + s_2 = 19 \\
& x_1 + x_2 + s_3 = 8 \\
& x_1, x_2, s_1, s_2, s_3 \geq 0
\end{aligned}
$$

Application in production scheduling

- A company produces computers
  - **R**egular production capacity is 160 computers per week
    - Production costs 190€ per computer
  - Additional 50 computers per week can be produced with **o**vertime (260€ / cmp.)
  - Cost of holding a computer in **i**nventory to satisfy future demand is 10€ / cmp.
  - Demand for the upcoming four weeks (105, 170, 230, 180 computers) needs to be satisfied

Question:

- Help the management understand this LP model for production planning:
  - What does decision variable i_2 represent?
  - What is the interpretation of constraint
    $$r_2 + o_2 + i_1 - i_2 = 170\ ?$$

$$\min 190r_1 + 260o_1 + 10i_1 + 190r_2 + 260o_2 + 10i_2 + 190r_3 + 260o_3 + 10i_3 + 190r_4 + 260o_4$$

$$r_1 + o_1 - i_1 = 105$$
$$r_2 + o_2 + i_1 - i_2 = 170$$
$$r_3 + o_3 + i_2 - i_3 = 230$$
$$r_4 + o_4 + i_3 = 180$$

$$0 \le r_j \le 160, j = 1, \dots, 4$$
$$0 \le o_j \le 50, j = 1, \dots, 4$$
$$i_j \ge 0, j = 1, \dots, 3$$

Auxiliary decision variables in production scheduling

N9 =SUMPRODUCT(B$5:L$5;B9:L9)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1																
2		Week 1			Week 2			Week 3			Week 4					
3		Regular prod.	Overtime prod.	Inventory carried over to next	Regular prod.	Overtime prod.	inventory to nect week	Regular prod.	Overtime prod.	inventory to nect week	Regular prod.	Overtime prod.				
4		r_1	o_1	i_1	r_2	o_2	i_2	r_3	o_3	i_3	r_4	o_4				
5	Units	160	0	55	160	0	45	160	25	0	160	20		*Tot. cost*		
6	Unit costs	190	260	10	190	260	10	190	260	10	190	260		134300		
7																
8		1	1	-1										105	=	105
9				1	1	1	-1							170	=	170
10							1	1	1	-1				230	=	230
11										1	1	1		180	=	180
12		1												160	<=	160
13			1											0	<=	50
14					1									160	<=	160
						1								0	<=	50
								1						160	<=	160
									1					25	<=	50
											1			160	<=	160
												1		20	<=	50

$$\min 190r_1 + 260o_1 + 10i_1 + 190r_2 + 260o_2 + 10i_2 + 190r_3 + 260o_3 + 10i_3 + 190r_4 + 260o_4$$

$$r_1 + o_1 - i_1 = 105$$
$$r_2 + o_2 + i_1 - i_2 = 170$$
$$r_3 + o_3 + i_2 - i_3 = 230$$
$$r_4 + o_4 + i_3 = 180$$

$$r_j \le 160, j = 1, \ldots, 4$$
$$o_j \le 50, j = 1, \ldots, 4$$

Solver Parameters

Set Objective: $N$6

To: ○ Max ◉ Min ○ Value Of:

By Changing Variable Cells:

$B$5:$L$5

Subject to the Constraints:

$N$8:$N$11 = $P$8:$P$11
$N$12:$N$19 <= $P$12:$P$19

☑ Make Unconstrained Variables Non-Negative

Select a Solving Method: Simplex LP

General matrix formulation of an LP problem

- Since each constraint can be presented with ≤-constraints and the objective function can be multiplied by -1 to change "min" to "max," any LP problem can be written in the form

$$\max \sum_{i=1}^{n} c_i x_i$$

$$\sum_{i=1}^{n} a_{ji} x_i \leq b_j, j = 1, \dots, m$$

$$\Leftrightarrow$$

$$\max \boldsymbol{c}^T \boldsymbol{x}$$

$$\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$$

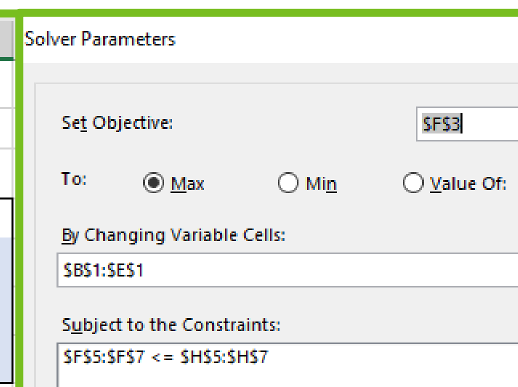
$$\Leftrightarrow$$

$$\max \begin{bmatrix} c_1 & \dots & c_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

$$\begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \leq \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

$$\Updownarrow$$

	A	B	C	D	E	F	G	H
1	$\boldsymbol{x}$	x_1	x_2	x_3	x_4			
2								
3	$\boldsymbol{c}^T$	c_1	c_2	c_3	c_4	=SUMPRODUCT($B$1:$E$1;B3:E3)		
4								$\boldsymbol{b}$
5	$\boldsymbol{A}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	=SUMPRODUCT($B$1:$E$1;B5:E5)	≤	b_1
6		$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	=SUMPRODUCT($B$1:$E$1;B6:E6)	≤	b_2
7		$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	=SUMPRODUCT($B$1:$E$1;B7:E7)	≤	b_3



Additional examples

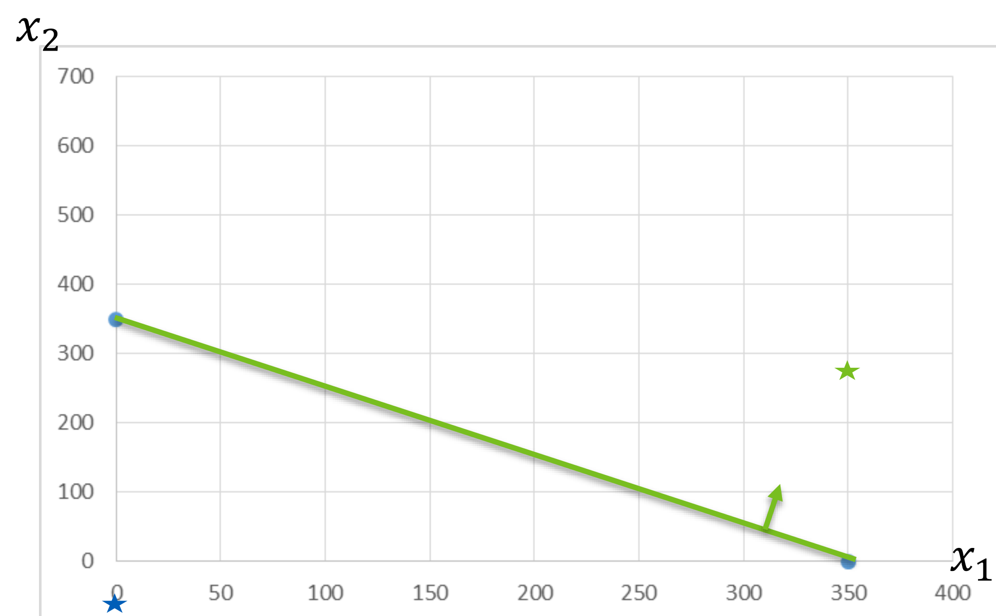
Example 1:

Graphical solution approach

Example 1: Solving LPs graphically

$$\min 2x_1 + 3x_2$$
$$x_1 + x_2 \geq 350 \quad (1)$$
$$2x_1 + x_2 \leq 600 \quad (2)$$
$$x_1, x_2 \geq 0$$

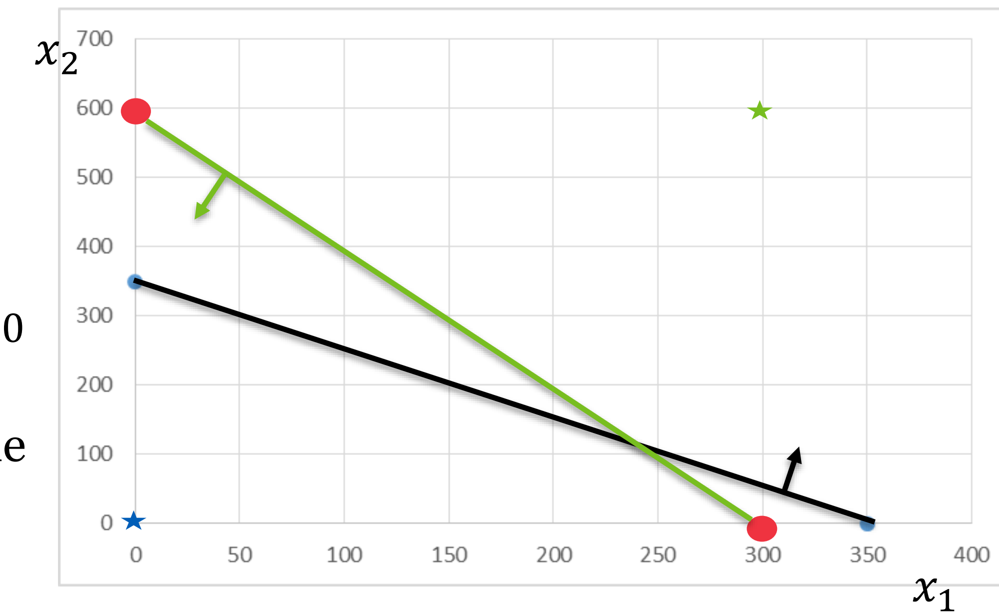
- Constraint (1)
  - Draw line $x_1 + x_2 = 350$
    - If x_1=0 then x_2=350
    - If x_2=0 then x_1=350
  - Hence the line goes through points (0,350) and (350,0)
  - Which side of the line is feasible?
    - ★(0,0) gives $0 + 0 \geq 350 \rightarrow$ constraint not satisfied
    - ★(350,350) gives $350 + 350 \geq 350 \rightarrow$ constraint satisfied
  - Thus, points located up and right of the line are feasible



Example 1: Solving LPs graphically

$$\min\ 2x_1 + 3x_2$$
$$x_1 + x_2 \geq 350 \quad (1)$$
$$2x_1 + x_2 \leq 600 \quad (2)$$
$$x_1, x_2 \geq 0$$

- Constraint (2)
  - Draw line $2x_1 + x_2 = 600$
    - If x_1=0 then x_2=600
    - If x_2=0 then x_1=300
  - Hence the line goes through points (0,600) and (300,0) ●
  - Which side of the line is feasible?
    - ★ (0,0) gives 0 ≤600 → constraint satisfied
    - ★ (300,600) gives $2 * 300 + 600 \leq 600$ → constraint not satisfied
  - Points down left and right of the line are feasible

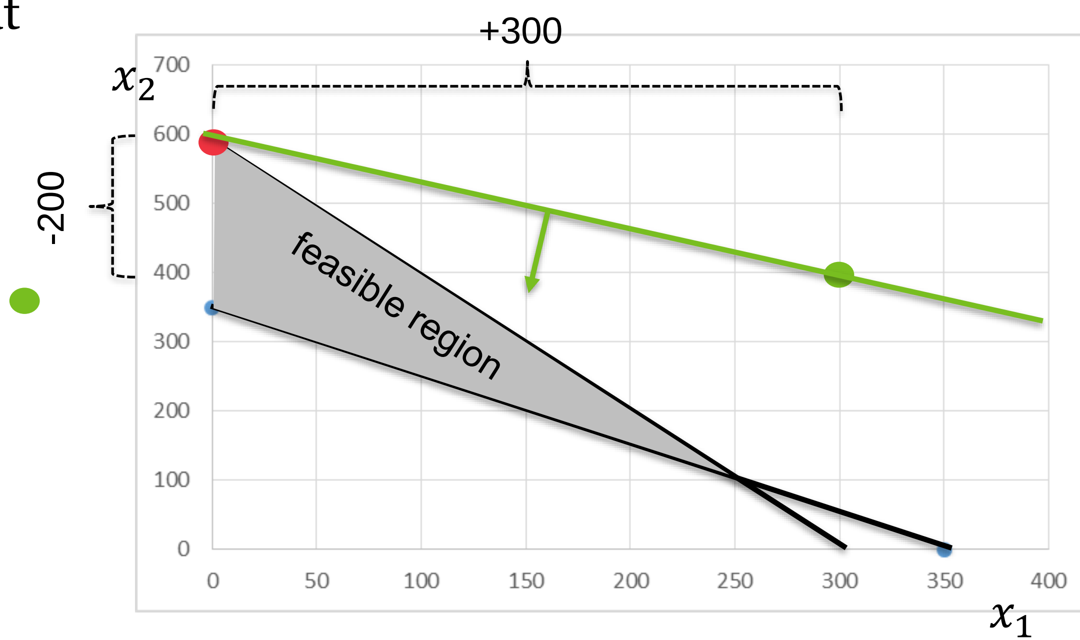


Example 1: Solving LPs graphically

$$\min z = 2x_1 + 3x_2$$
$$x_1 + x_2 \geq 350 \quad (1)$$
$$2x_1 + x_2 \leq 600 \quad (2)$$
$$x_1, x_2 \geq 0$$

- Objective function line
  - Evaluate objective function at point (0,600)
    $z = 2 * 0 + 3 * 600 = 1800$
  - Find point $(300, x_2)$ such that z=1800
    $z = 2 * 300 + 3 * x_2 = 1800$
    $\Rightarrow x_2 = 400$
  - Objective function value is 1800 also at point (300,400)

- The contours of a linear function are straight lines



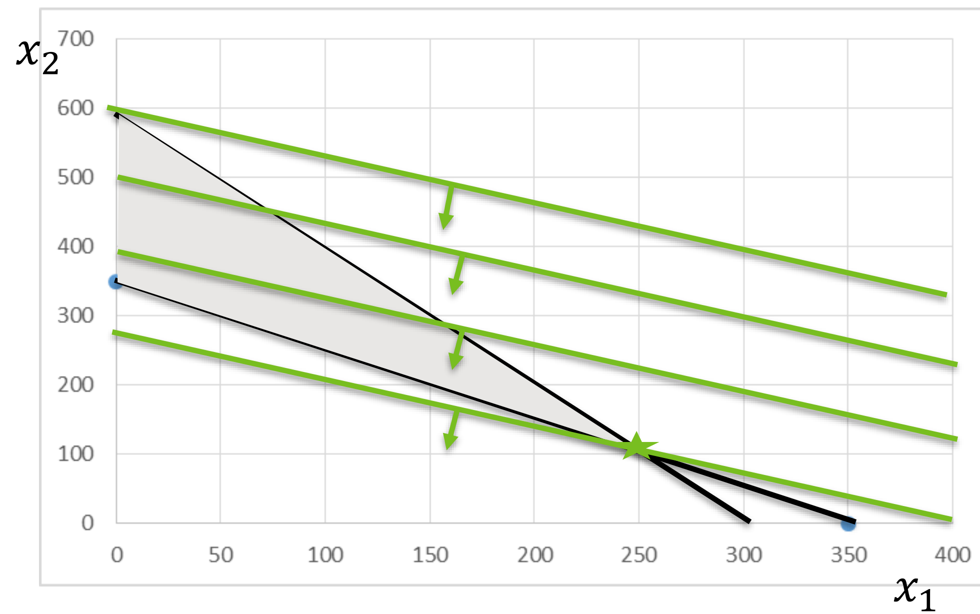
$$2 * (+300) + 3 * (-200) = 600 - 600 = 0$$

Example 1: Solving LPs graphically

$$\min z = 2x_1 + 3x_2$$
$$x_1 + x_2 \geq 350 \quad (1)$$
$$2x_1 + x_2 \leq 600 \quad (2)$$
$$x_1, x_2 \geq 0$$

- Move objective function line
  - Clearly at (0,0) z=0 → better solutions are found by when both decision variable decrease
- Optimum is at the extreme point defined by constraints (1) and (2)
  - Solve (x_1, x_2) such that $x_1 + x_2 = 350 \text{ and } 2x_1 + x_2 = 600$

$$x_2 = 350 - x_1$$
$$2x_1 + (350 - x_1) = 600$$
$$\Rightarrow x_1 = 250$$
$$\Rightarrow x_2 = 350 - 250 = 100$$
$$\Rightarrow (x_1, x_2) = (250{,}100) \bigstar$$



- Optimal objective function value:
  z=2*250+3*100=800

Example 1: Binding vs. Redundant constraints

- Constraints (1) and (2) are **binding** since at optimum we have

$$x_1 + x_2 = 350$$
$$2x_1 + x_2 = 600$$

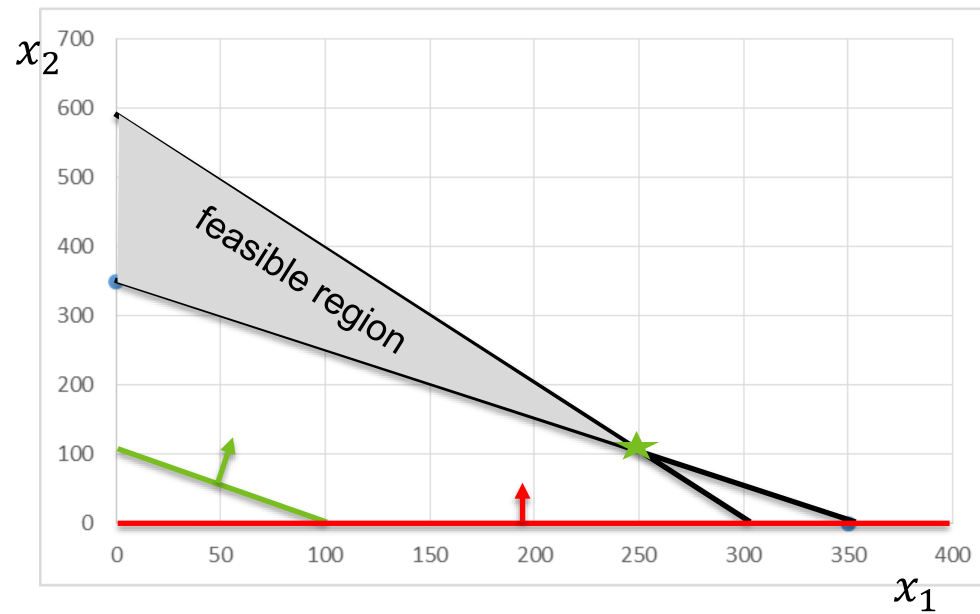
  - $x_1 \geq 0, x_2 \geq 0$ are not binding

- A constraint is **redundant** if removing it would not change the feasible region

  $\rightarrow$ The constraint $x_2 \geq 0$ is redundant

  - Also the constraint $x_1 + x_2 \geq 100$ would be redundant

$$\min z = 2x_1 + 3x_2$$
$$x_1 + x_2 \geq 350 \quad (1)$$
$$2x_1 + x_2 \leq 600 \quad (2)$$
$$x_1, x_2 \geq 0$$



Example 2:

Optimization with Excel Solver

Example 2: Spreadsheet formulation

LP Formulation

$$\text{Max} \quad z = 5x_1 + 7x_2$$

$$\text{s.t.} \quad x_1 \le 6$$

$$2x_1 + 3x_2 \le 19$$

$$x_1 + x_2 \le 8$$

$$x_1, x_2 \ge 0$$

1. Enter the problem data.

- Objective coefficients are in cells B4 and C4.
- Constraint coefficients are in cells B7, C7, B8, C8, B9, C9.

2. Specify cell locations for all decision variables.

- Cell B2 is reserved for x1, cell C2 is reserved for x2.

3. Select a cell and enter a formula for computing the value of the objective function.

- Cell D4 contains formula for computing value of the obj. function.

	A	B	C	D	E	F
1		**x1**	**x2**			
2	**Decision Variables**					
3						
4	**Objective Coefficients**	5	7	=SUMPRODUCT(B$2:C$2,B4:C4)		
5						
6						
7	**Constraint #1**	1	0		<=	6
8	**Constraint #2**	2	3		<=	19
9	**Constraint #3**	1	1		<=	8

Example 2: Spreadsheet formulation

LP Formulation

$$\text{Max} \quad z = 5x_1 + 7x_2$$

$$\text{s.t.} \quad x_1 \leq 6$$

$$2x_1 + 3x_2 \leq 19$$

$$x_1 + x_2 \leq 8$$

$$x_1, x_2 \geq 0$$

4. Select a cell and enter a formula for computing the left-hand-side of each constraint.

- Cells D7, D8, D9 contain formulas for computing the LHSs of the constraints.

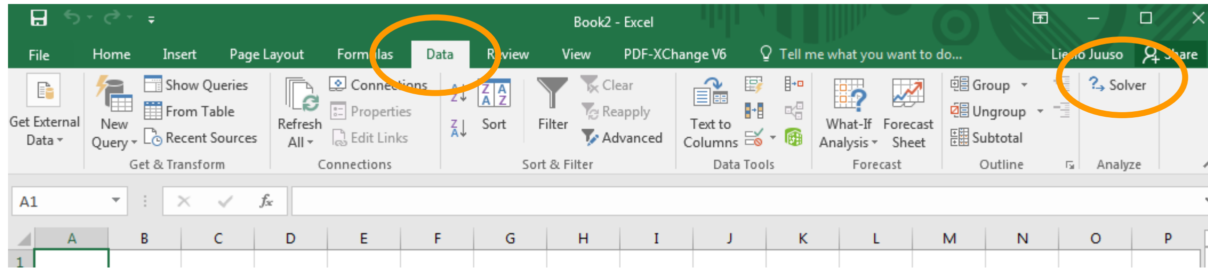
5. Select a cell and enter the value of the RHS of each constraint

- Cells F7, F8, F9 contain the RHS values.

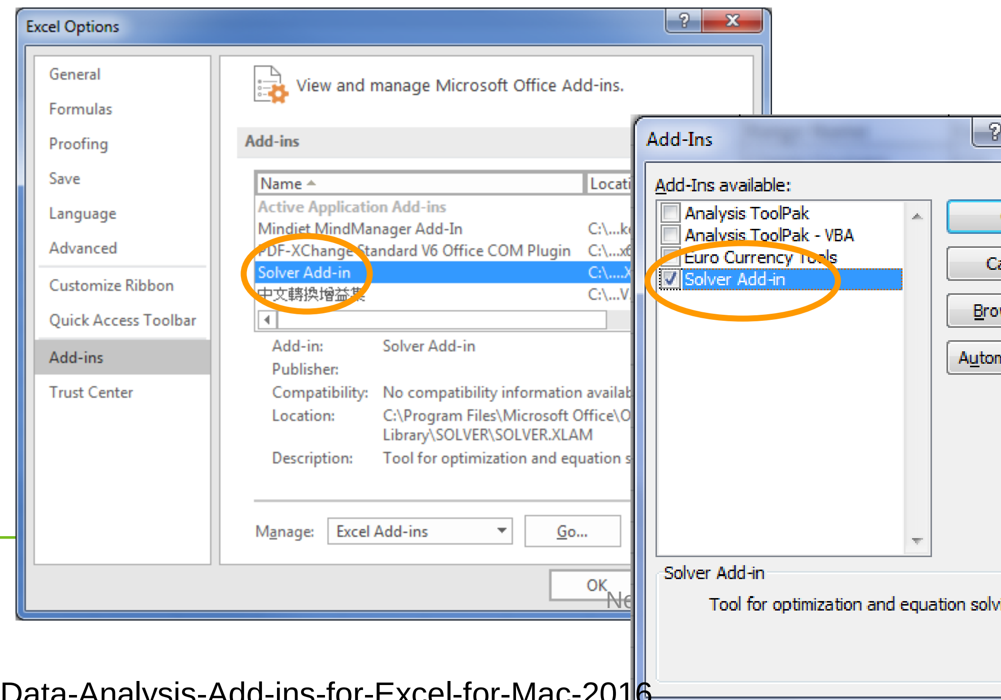
	A	B	C	D	E	F
1		**x1**	**x2**			
2	**Decision Variables**					
3						
4	**Objective Coefficients**	5	7	=SUMPRODUCT(B$2:C$2,B4:C4)		
5						
6						
7	**Constraint #1**	1	0	=SUMPRODUCT(B$2:C$2,B7:C7)	<=	6
8	**Constraint #2**	2	3	=SUMPRODUCT(B$2:C$2,B8:C8)	<=	19
9	**Constraint #3**	1	1	=SUMPRODUCT(B$2:C$2,B9:C9)	<=	8

Example 2: Installing and opening the solver

- Select the **Solver** option from **Data** tab



- If the solver is not visible you have to install it first:
  - File -> Options -> Add-ins
  - In the Add-ins box, select Solver Add-In and click Go.
  - In the Add-Ins available box, check the Solver Add-in and then OK.

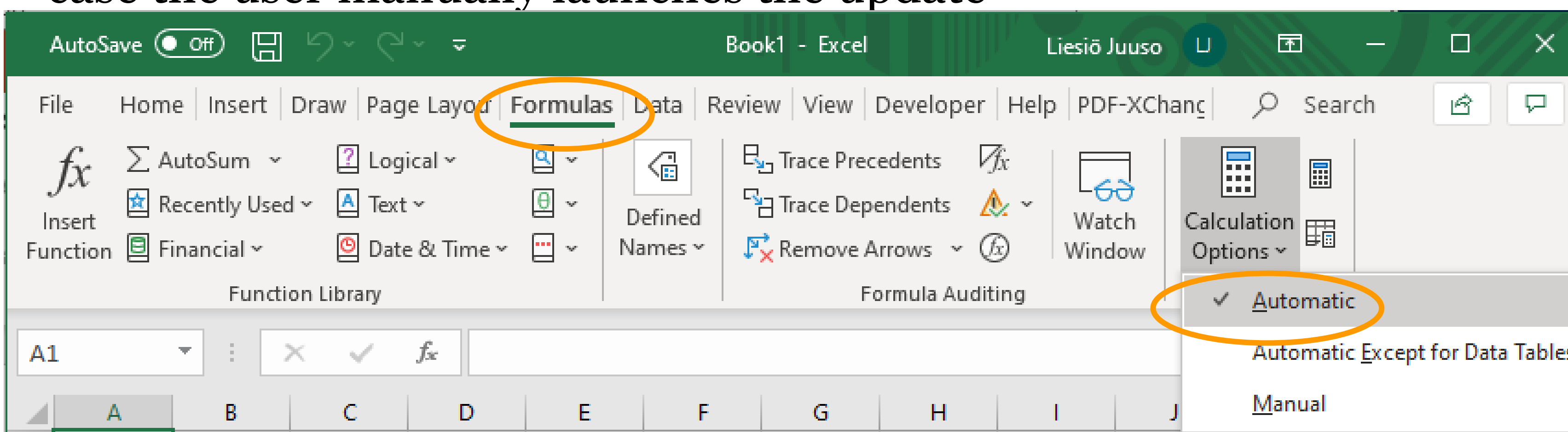


Aalto University
School of Business

MacOS:
https://support.ivey.ca/hc/en-us/articles/209172446-Solver-and-Data-Analysis-Add-ins-for-Excel-for-Mac-2016

Example 2: Set formula calculation to "Automatic" – otherwise solver will not work!

- Solver needs to know how the changes it makes to the decision variable values affects the values of the objective function and the constraints
- This is not possible if the values of these cells are calculated only in the case the user manually launches the update

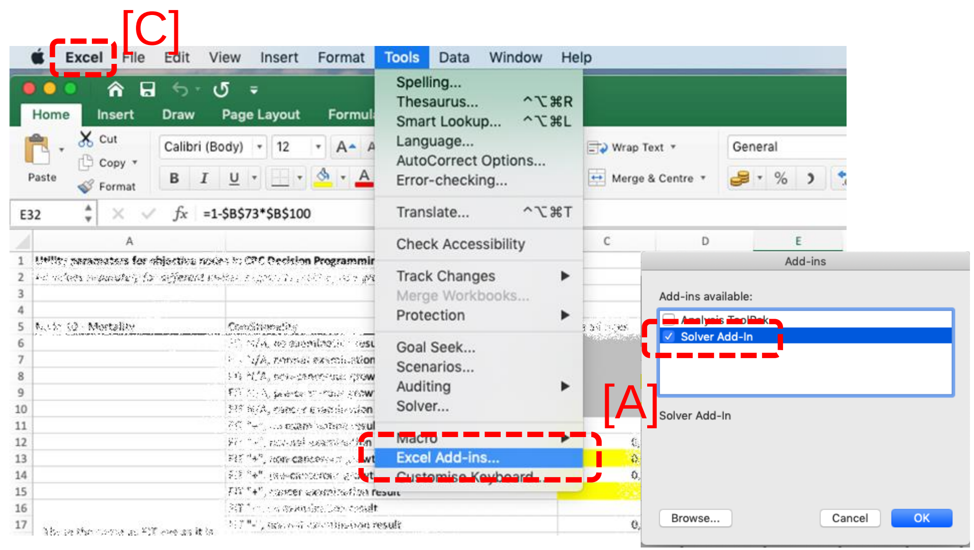


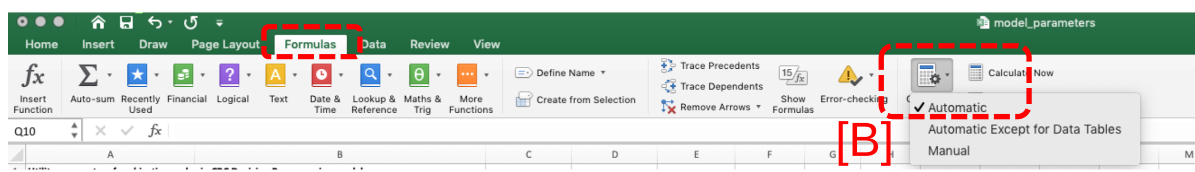
Example 2: Using Excel solver on Mac

- Solver is on the '**Data**'-tab if it is installed

- To install Solver [A]
  - **Tools → Excel Add-ins → Solver**

- Enable automatic update of call values from `**Formulas**'-tab [B]
  - Verify that Calculation is set to Automatic in **Excel-**menu → **Preferences → Calculation** [C]





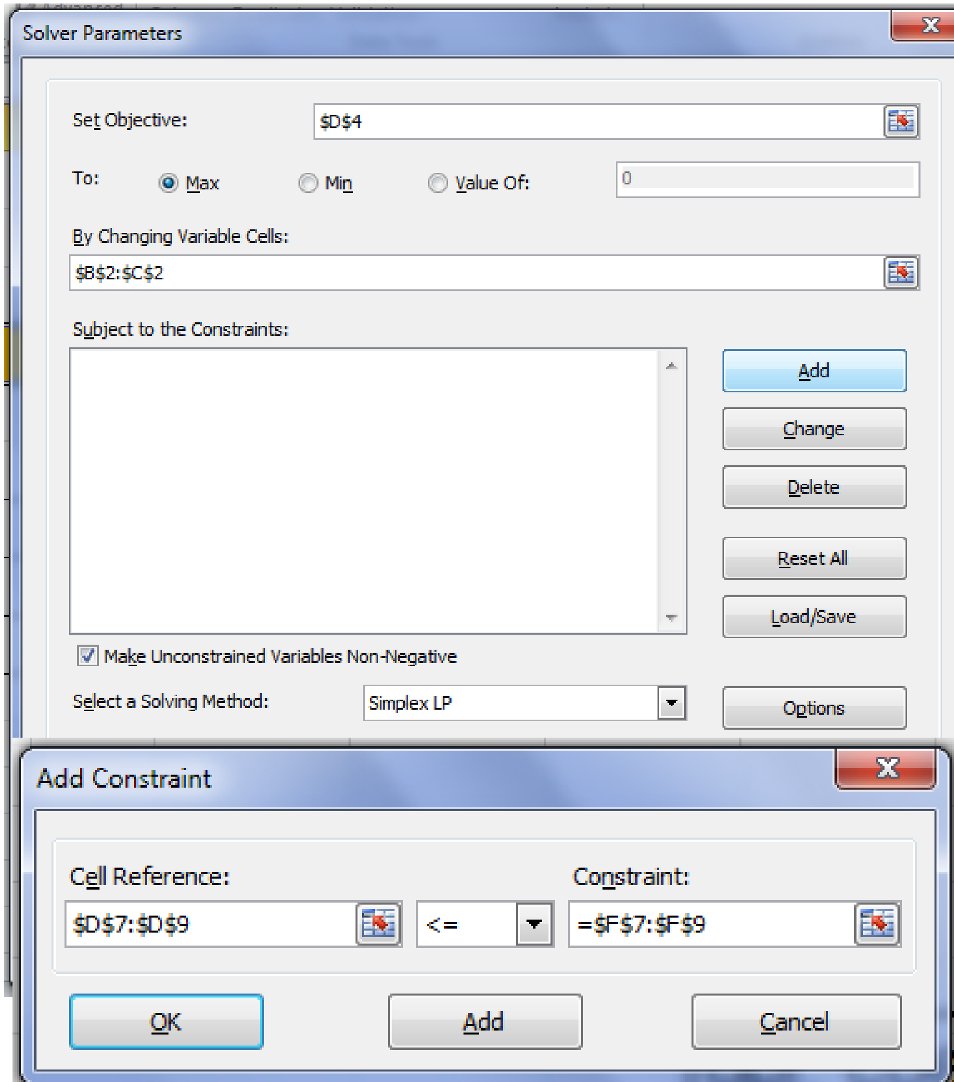
Example 2: Using the Excel Solver

In **solver parameters** dialog box:

- Enter D4 into the **Set Target Cell** box
- Select the **Max** option
- Enter B2:C2 into the **By Changing Cells** box
- Choose **Add**

In **Add Constraint** dialog box:

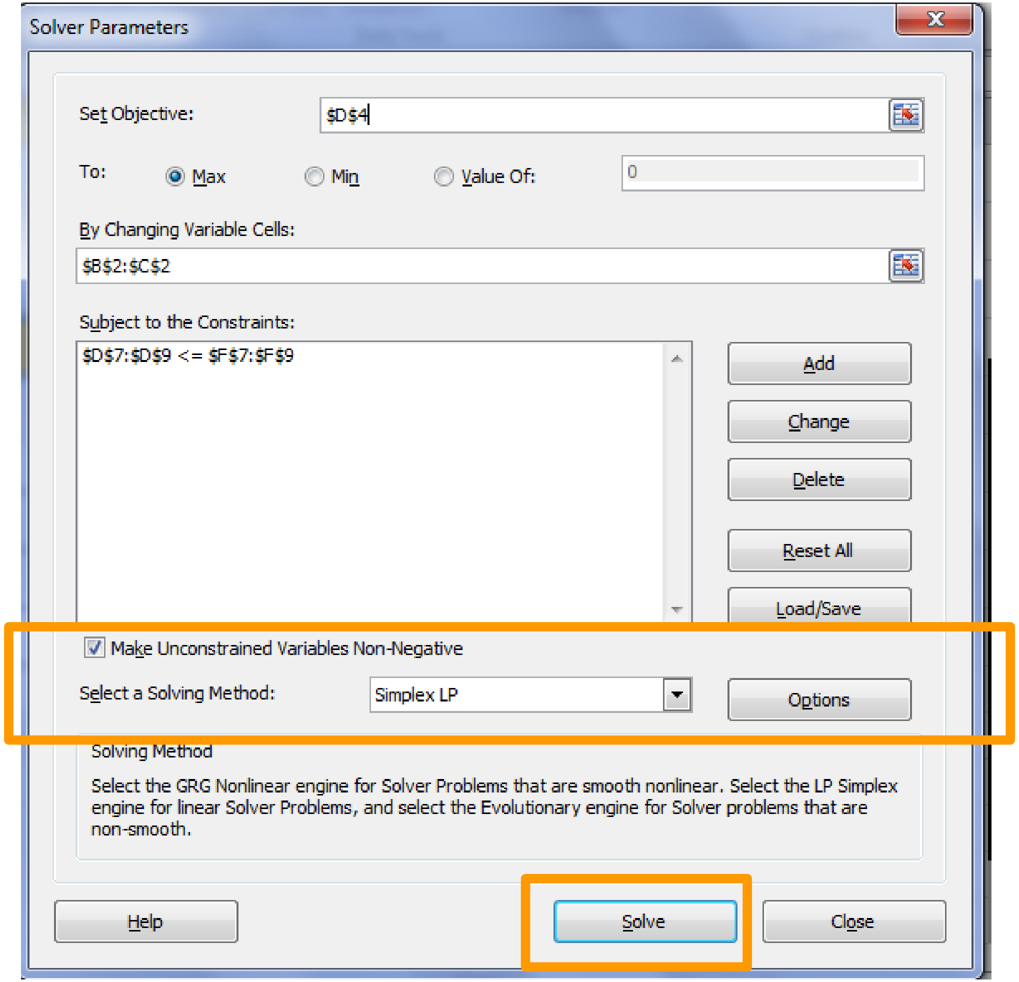
- Enter D7:D9 in the **Cell Reference** box
- Select <=
- Enter F7:F9 into the **Constraint** box
- Choose **OK**



Example 2: Using the Excel Solver

In the **solver parameters** dialog box:

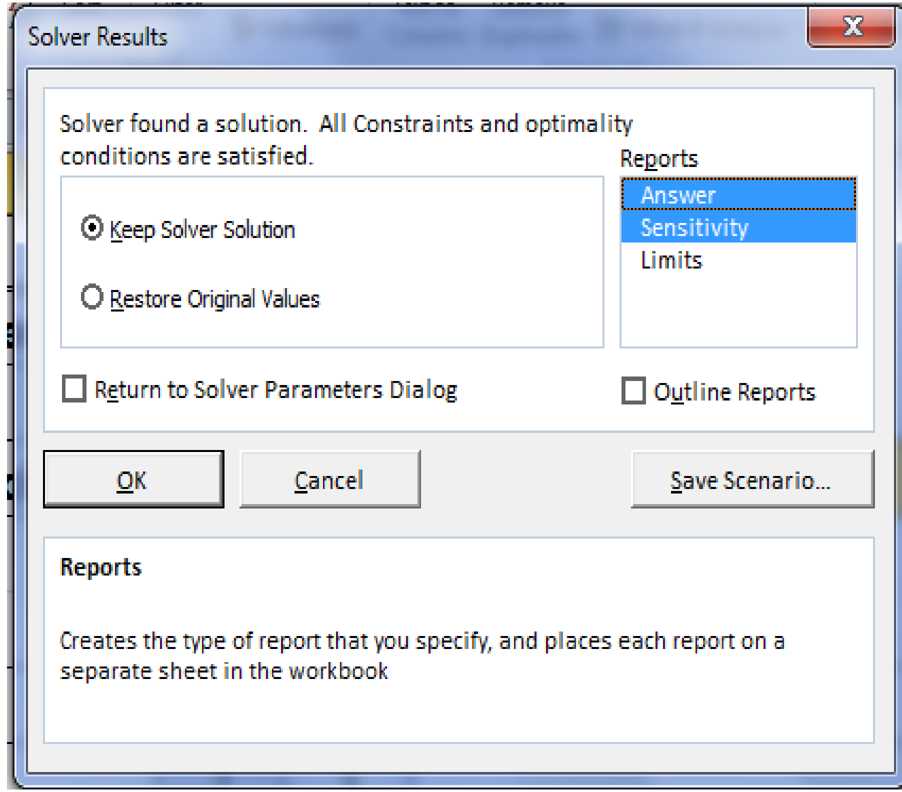
- Select **Simplex LP** as a Solving Method
- Make unconstrained Variables **Non-negative**
- Choose **Solve**



Example 2: Using the Excel Solver

When the **Solver Results** dialog box appears:

- Select **Keep Solver Solution**
- Select **Answer** and **Sensitivity** reports
- Choose **OK** to produce the optimal solution output.



Example 2: Solver solution

	A	B	C	D	E	F
1		x1	x2			
2	Decision Variables	5	3			
3						
4	Objective Coefficients	5	7	46		
5						
6						
7	Constraint #1	1	0	5	<=	6
8	Constraint #2	2	3	19	<=	19
9	Constraint #3	1	1	8	<=	8

Example 2: Answer report

Objective Cell (Max)

Cell	Name	Original Value	Final Value
$D$4	Objective Coefficients	0	46

Variable Cells

Cell	Name	Original Value	Final Value	Integer
$B$2	Decision Variables x1	0	5	Contin
$C$2	Decision Variables x2	0	3	Contin

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$D$7	Constraint #1	5	$D$7<=$F$7	Not Binding	1
$D$8	Constraint #2	19	$D$8<=$F$8	Binding	0
$D$9	Constraint #3	8	$D$9<=$F$9	Binding	0

Example 2: Sensitivity report

Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$2	Decision Variables x1	5	0	5	2	0.333333333
$C$2	Decision Variables x2	3	0	7	0.5	2

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$D$7	Constraint #1	5	0	6	1E+30	1
$D$8	Constraint #2	19	2	19	5	1
$D$9	Constraint #3	8	1	8	0.333333333	1.666666667

Aalto University
School of Business

Linear programming – Sensitivity Analysis

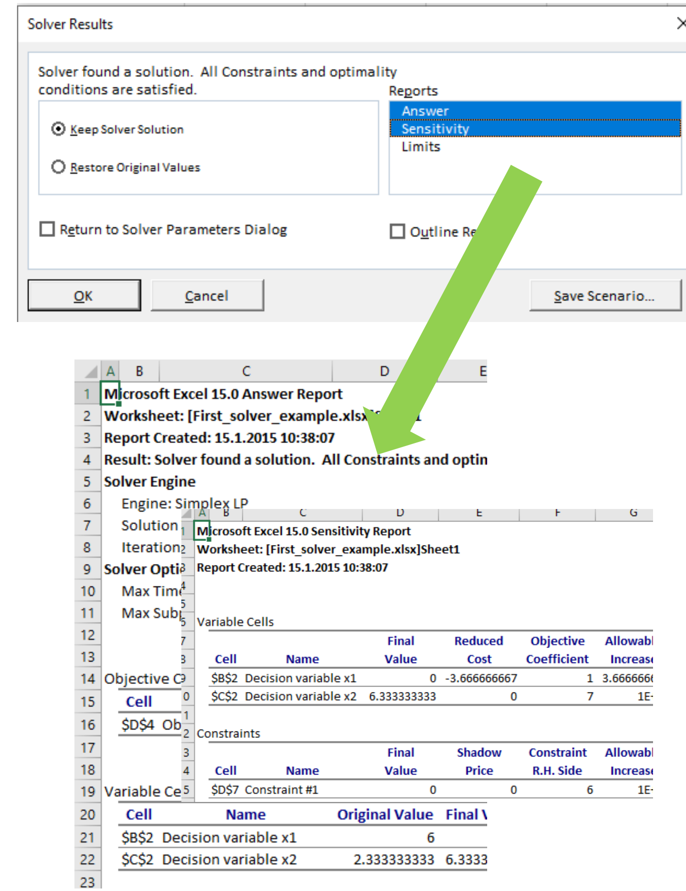
- *Sensitivity Analysis*
- *The dual problem*

LP - Sensitivity Analysis

- Solving a LP problem with the Simplex algorithm produces other information on the problem besides the optimal solution

- This information gives insight how the optimal solution is affected by changes in
  - the objective function coefficients
  - the constraints' right-hand side (RHS) values

- Allows the manager to ask certain <u>what-if questions</u> about the problem.



Information about the Objective function

- Range of optimality is the range of objective function coefficient values over which the optimal solution does not change
  - Optimal solution = optimal values for decision variables

- Reduced cost is the amount the decision variable's objective function coefficient would have to be improved before the variable's optimal value would be positive
  - Improved = increased for max. and decreased for min. problems

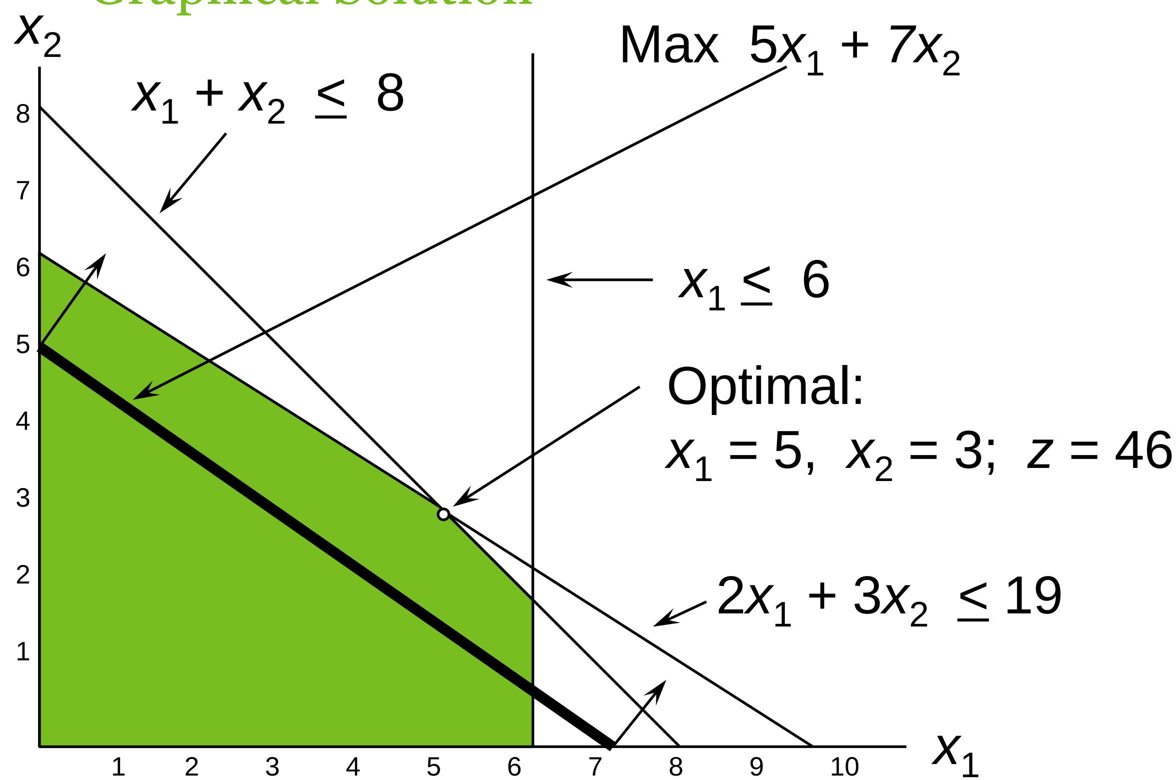
  →The reduced cost for a decision variable with a positive optimal value is zero.

Iron Works - Example Revisited

LP Formulation

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 7x_2 \\
& x_1 \leq 6 \\
& 2x_1 + 3x_2 \leq 19 \\
& x_1 + x_2 \leq 8 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Graphical Solution



Iron works: Reduced costs and Optimality ranges

Reduced cost for c_1 and c_2

Range of Optimality for c_1 and c_2

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	X1	5.0	0.0	5	2	0.333333333
$C$8	X2	3.0	0.0	7	0.5	2

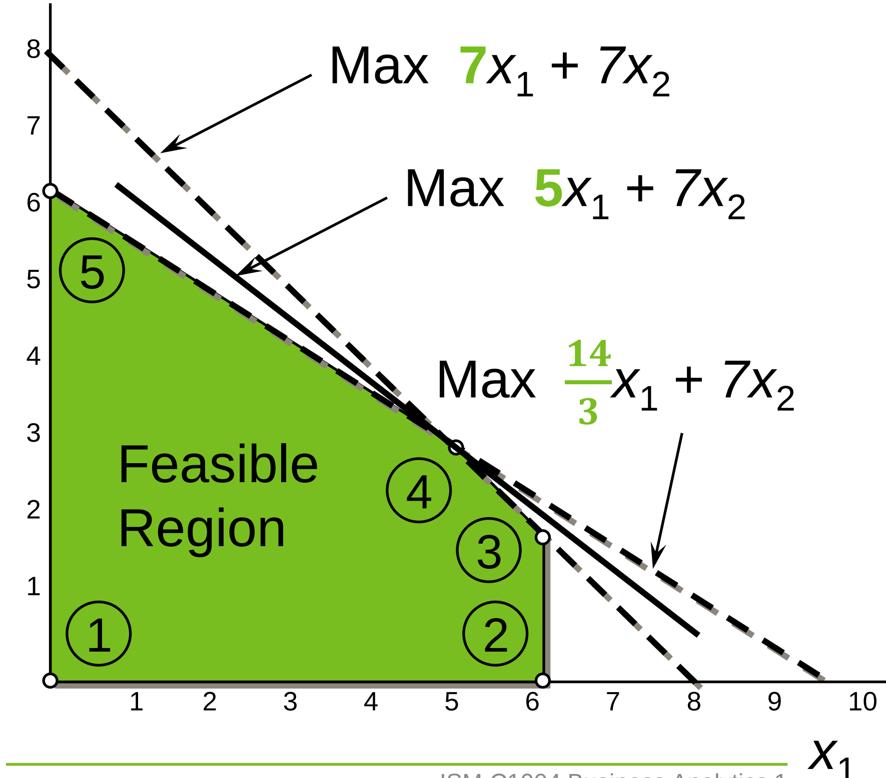
Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	#1	5	0	6	1E+30	1
$B$14	#2	19	2	19	5	1
$B$15	#3	8	1	8	0.333333333	1.666666667

Iron Works: Range of optimality – Visual interpretation

- Range of Optimality for c_1
  - The slope of the objective function line $\boldsymbol{c_1x_1 + 7x_2}$ is $\boldsymbol{-c_1/7}$.
  - The slope of the first binding constraint $\boldsymbol{x_1 + x_2 = 8}$, is **-1**
  - The slope of the second binding constraint $\boldsymbol{2x_1 + 3x_2 = 19}$, is **-2/3.**
  - Find values for c_1 such that the objective function slope lies between those of the two binding constraints:

    $\boldsymbol{-1 \leq -c_1/7 \leq -2/3}$
  - Multiplying by -7 and reversing the inequalities gives: $\boldsymbol{14/3 \leq c_1 \leq 7}$
  - This is consistent with Solver results:
    - 5-14/3=0.33333 (allowed decrease)
    - 7-5=2 (allowed increase)



Slope

$$ax_1 + bx_2 = d \Leftrightarrow x_2 = \boxed{-\frac{a}{b}}x_1 + \frac{d}{b}$$

Information about the constraints

- The change in the optimal value of the objective function per unit increase in the constraint right-hand side is called the **shadow price**.
  - E.g., added revenue (10k€) per unit (kton) increase in the available amount of iron ore

→If a constraint is not binding then its shadow price is zero.

- The **range of feasibility** is the range over which the shadow price is applicable
  - E.g., at some point additional iron ore will no longer increase revenues as other constraints limit production

Iron Works: Shadow Prices and Ranges of Feasibility

Shadow prices

Ranges of feasibility

Adjustable Cells

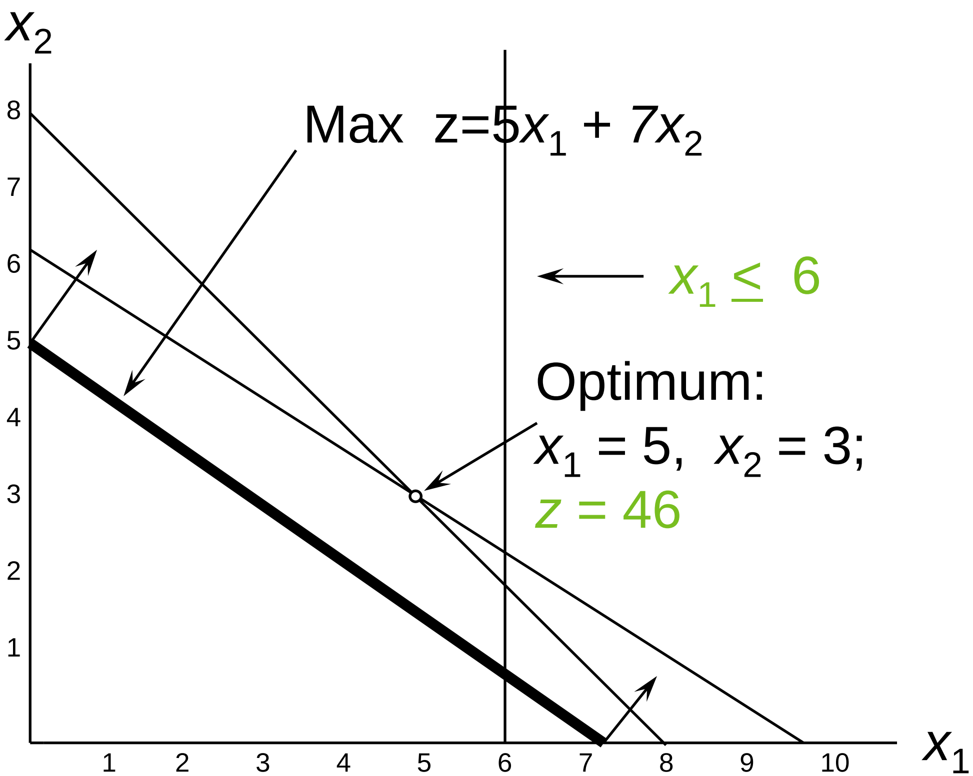
Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	X1	5.0	0.0	5	2	0.333333333
$C$8	X2	3.0	0.0	7	0.5	2

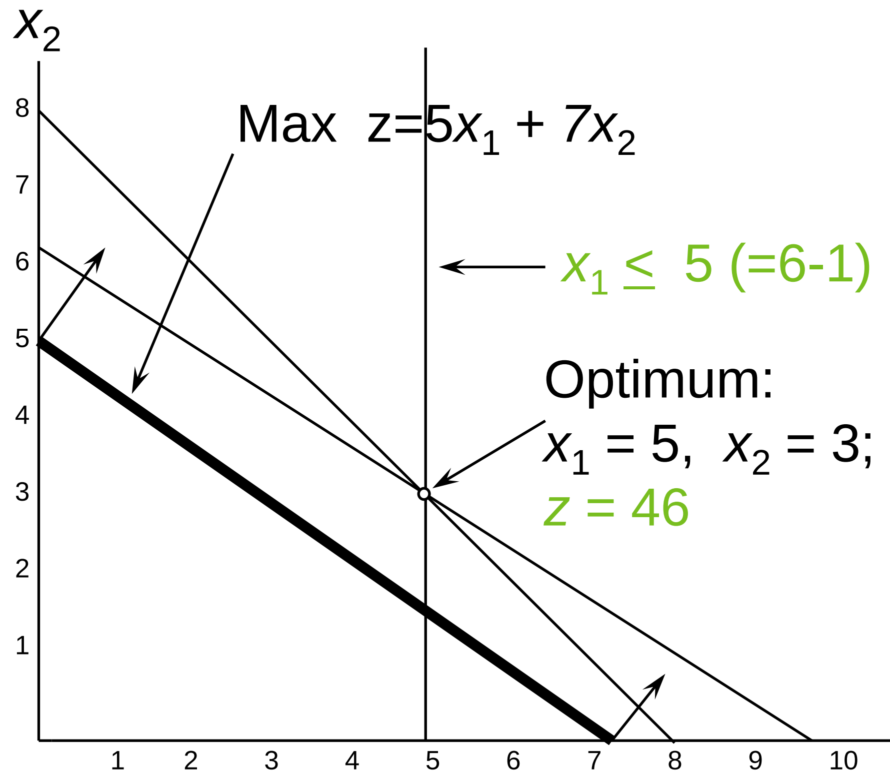
Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	#1	5	0	6	1E+30	1
$B$14	#2	19	2	19	5	1
$B$15	#3	8	1	8	0.333333333	1.666666667

$1E+30=1\times 10^{30}=1000000000000000000000000000000$

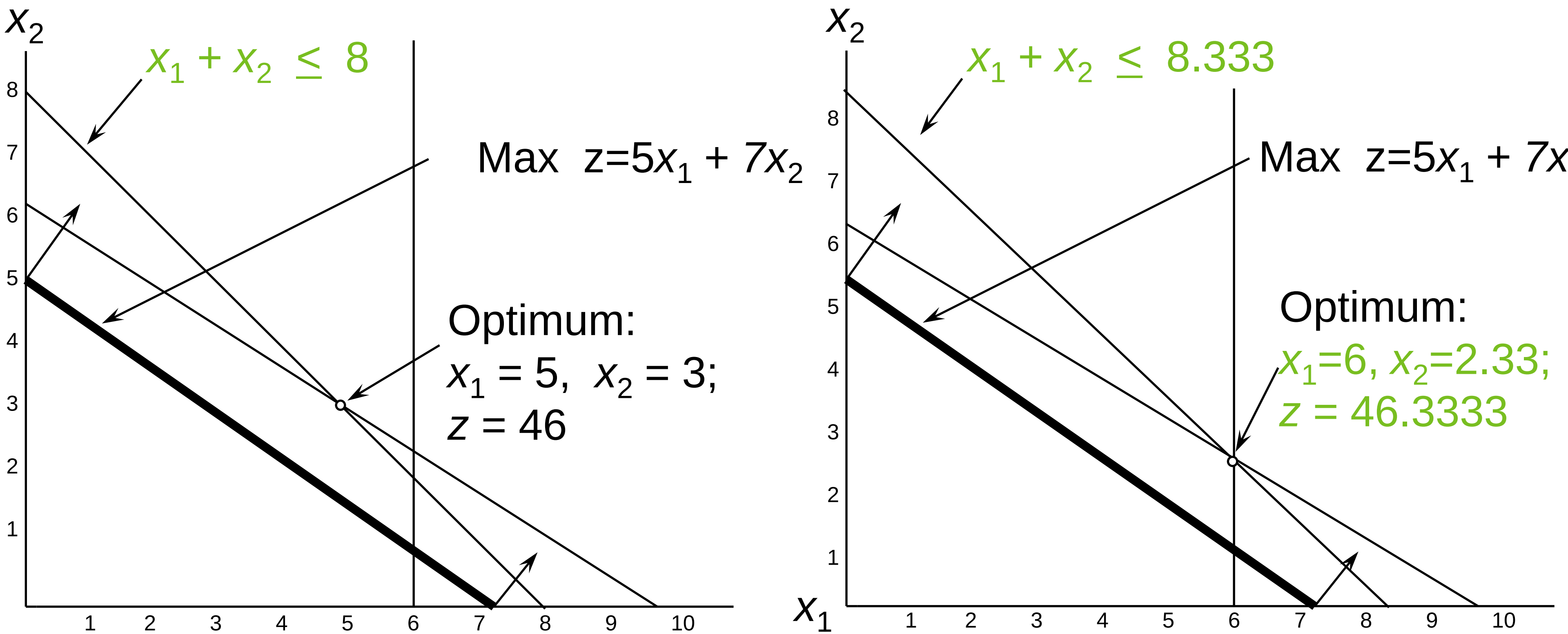
Iron Works: Shadow Prices and Ranges of Feasibility Constraint 1





Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
\$B\$13	#1	5	0	6	1E+30	1
\$B\$14	#2	19	2	19	5	1
\$B\$15	#3	8	1	8	0.333333333	1.666666667

Iron Works: Shadow Prices and Ranges of Feasibility Constraint 3



Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	#1	5	0	6	1E+30	1
$B$14	#2	19	2	19	5	1
$B$15	#3	8	1	8	0.333333333	1.666666667

Example: Olympic Bike Co.

- Olympic Bike is introducing two new lightweight bicycle frames, the Deluxe and the Professional, to be made from special aluminum and steel alloys.
  - The anticipated unit profits are $10 for the Deluxe and $15 for the Professional.
  - The number of pounds of each alloy needed per frame is

	Aluminum Alloy	Steel Alloy
Deluxe	2	3
Professional	4	2

  - A supplier delivers 100 pounds of the aluminum alloy and 80 pounds of the steel alloy weekly.
- How many Deluxe and Professional frames should Olympic produce each week?

Olympic Bike Co. – Model Formulation

Verbal:

- Objective Function: “Maximize total weekly profit”.
- Constraints:
  - “Total weekly usage of aluminum alloy no more than 100 pounds“
  - “Total weekly usage of steel alloy no more than than 80 pounds”
- Decision Variables:
  - “x_1 is the number of Deluxe frames produced weekly”
  - “x_2 is the number of Professional frames produced weekly”

Mathematical:

$$\begin{array}{lrl} \max & 10x_1 + 15x_2 & \text{(Total Weekly Profit)} \\ \text{s.t.} & 2x_1 + 4x_2 \leq 100 & \text{(Aluminum Available)} \\ & 3x_1 + 2x_2 \leq 80 & \text{(Steel Available)} \\ & x_1, x_2 \geq 0 & \end{array}$$

Olympic Bike Co. - Solution

Objective Cell (Max)

Cell	Name	Original Value	Final Value
$E$5	Obj. Coef	412.5	412.5

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	Deluxe	15	0	10	12.5	2.5
$C$8	Profess.	17.500	0.000	15	5	8.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	Aluminum	100	3.125	100	60	46.66666667
$B$14	Steel	80	1.25	80	70	30

Question (Optimal solution)

What is the optimal production plan, i.e., how many deluxe and professional bikes should be produced weekly?

Olympic Bike Co. - Solution

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	Deluxe	15	0	10	12.5	2.5
$C$8	Profess.	17.500	0.000	15	5	8.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	Aluminum	100	3.125	100	60	46.66666667
$B$14	Steel	80	1.25	80	70	30

Question (Ranges of optimality)

Suppose the unit profit on deluxe frames is increased to $20. Is the above solution still optimal? What is the total profit?

Yes, solution is still optimal, since increase of $10 is below $12.5.
Profit increase= 15 bikes x 10 $/bike= $150. Total profit $412.5+$150 = $562.5

Olympic Bike Co. - Solution

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	Deluxe	15	0	10	12.5	2.5
$C$8	Profess.	17.500	0.000	15	5	8.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	Aluminum	100	3.125	100	60	46.66666667
$B$14	Steel	80	1.25	80	70	30

Question (Ranges of optimality)

If the unit profit on deluxe frames were $6 instead of $10, would the optimal solution change?

The solution would change since decrease of $4 is greater than $2.5

Olympic Bike Co. - Solution

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	Deluxe	15	0	10	12.5	2.5
$C$8	Profess.	17.500	0.000	15	5	8.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	Aluminum	100	3.125	100	60	46.66666667
$B$14	Steel	80	1.25	80	70	30

Question (Ranges of feasibility)

What is the maximum amount the company should pay for 50 extra pounds of aluminum?

50lb < 60lb → shadow price holds → increase in the objective function value is 50lb x 3.125 $/lb=$156.25. This is the maximum price.

Olympic Bike Co. - Solution

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$8	Deluxe	15	0	10	12.5	2.5
$C$8	Profess.	17.500	0.000	15	5	8.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$13	Aluminum	100	3.125	100	60	46.66666667
$B$14	Steel	80	1.25	80	70	30

Question (Ranges of feasibility)

What would the optimal profit be if the company had 200 pounds of steel?

Increase of 120 (=200-80) is greater than the allowable increase of 70, so the sensitivity report does not provide the answer.

Final comments on sensitivity analysis

- Sensitivity analysis gives information on what would happen if only one coefficient would be changed
  - I.e., it is assumed that all other parameters of the problem remain unchanged
- One can always solve (optimize) the problem for a new set of parameter values and then analyze how the optimal solution and the objective function value have changed
  - If the problem is very fast to solve this can be done for hundreds of different parameter value combinations (=Global Sensitivity Analysis)
- Two more extra examples on Sensitivity Analysis can be found at the end of this slide set

Dual problem

- Every LP problem has dual problem, which is also an LP problem
  - Knowledge of the dual provides interesting economic and sensitivity analysis insights
  - Dual might be faster to solve
- **Property 1:** If the dual has an optimal solution, then the original problem has an optimal solution. Furthermore, the optimal objective function values of the problems are identical
- **Property 2**: Dual of the dual problem is equal to the primal problem

Primal (=original) LP

Decision variables: $x_1,\ldots, x_n$

Number of constrains: m

$$\begin{aligned} \max z &= c_1x_1 + c_2x_2 + \ldots + c_nx_n \\ \text{s.t.}\quad & a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \le b_1 \\ & a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n \le b_2 \\ & \ldots \quad \ldots \quad \ldots \quad \ldots \\ & a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \le b_m \\ & x_j \ge 0 \ (j = 1, 2, \ldots, n) \end{aligned}$$

Dual of the original LP

Decision variables: $y_1,\ldots, y_m$

Number of constrains: n

$$\begin{aligned} \min w &= b_1y_1 + b_2y_2 + \ldots + b_my_m \\ \text{s.t.}\quad & a_{11}y_1 + a_{21}y_2 + \ldots + a_{m1}y_m \ge c_1 \\ & a_{12}y_1 + a_{22}y_2 + \ldots + a_{m2}y_m \ge c_2 \\ & \ldots \quad \ldots \quad \ldots \quad \ldots \\ & a_{1n}y_1 + a_{2n}y_2 + \ldots + a_{mn}y_m \ge c_n \\ & y_i \ge 0 \ (i = 1, 2, \ldots, m) \end{aligned}$$

Example: The Dakota Furniture Co

- The Dakota Furniture Company manufactures desks, tables, and chairs using resources with the selling prices as follows:
  - Currently, 48 board feet of lumber, 20 finishing hours, and 8 carpentry hours are available.
  - Because the available resources have already been purchased, Dakota wants to maximize total revenue.

	Desk	Table	Chair
Lumber (board ft)	8	6	1
Finishing hours	4	2	1.5
Carpentry hours	2	1.5	0.5
Selling price	\$60	\$30	\$20

The primal is: $\max z = 60x_1 + 30x_2 + 20x_3$

$$
\begin{aligned}
\text{s.t.}\quad 8x_1 + 6x_2 + x_3 &\le 48 \quad &&\text{(Lumber constraint)}\\
4x_1 + 2x_2 + 1.5x_3 &\le 20 &&\text{(Finishing constraint)}\\
2x_1 + 1.5x_2 + 0.5x_3 &\le 8 &&\text{(Carpentry constraint)}\\
x_1, x_2, x_3 &\ge 0
\end{aligned}
$$

Dakota Example: The Dual

The primal is: $\max z = 60x_1 + 30x_2 + 20x_3$

$$
\begin{array}{llll}
\text{s.t.} & 8x_1 + 6x_2 + x_3 \le 48 & & \text{(Lumber constraint)} \\
& 4x_1 + 2x_2 + 1.5x_3 \le 20 & & \text{(Finishing constraint)} \\
& 2x_1 + 1.5x_2 + 0.5x_3 \le 8 & & \text{(Carpentry constraint)} \\
& x_1, x_2, x_3 \ge 0 & &
\end{array}
$$

The dual is: $\min w = 48y_1 + 20y_2 + 8y_3$

$$
\begin{array}{llll}
\text{s.t.} & 8y_1 + 4y_2 + 2y_3 \ge 60 & & \text{(Desk constraint)} \\
& 6y_1 + 2y_2 + 1.5y_3 \ge 30 & & \text{(Table constraint)} \\
& y_1 + 1.5y_2 + 0.5y_3 \ge 20 & & \text{(Chair constraint)} \\
& y_1, y_2, y_3 \ge 0 & &
\end{array}
$$

Dakota Example: Interpretation of the Dual

Resource	Desk	Table	Chair	Availability
Lumber (board feet)	8	6	1	48
Finishing hours	4	2	1.5	20
Carpentry hours	2	1.5	0.5	8
Selling Price	$60	$30	$20	

- Suppose an entrepreneur wants to purchase all of Dakota's resources
  - I.e., 48 board feet of lumber, 20 finishing hours, and 8 carpentry hours
- The entrepreneur minimizes purchase price:

$$\min w = 48y_1 + 20y_2 + 8y_3$$

  - y_1: price paid for 1 board feet of lumber
  - y_2: price paid for 1 finishing hour
  - y_3: price paid for 1 carpentry hour
- But Dakota will not sell if it can make more revenue by making furniture from these resource!

Dakota Example: Interpretation of the Dual

Resource	Desk	Table	Chair	Availability
Lumber (board feet)	8	6	1	48
Finishing hours	4	2	1.5	20
Carpentry hours	2	1.5	0.5	8
Selling Price	$60	$30	$20	

$\min w = 48y_1 + 20y_2 + 8y_3$

y_1 = price for 1 board feet of lumber

y_2 = price for 1 finishing hour

y_3 = price for 1 carpentry hour

- From 8 board feet of lumber, 4 finishing hours, and 2 carpentry hours Dakota can produce a $60 desk. Hence, the total price for this resource combination has to exceed 60:

$$8y_1 + 4y_2 + 2y_3 \geq 60$$

- A 30$ table can be produced from resources 6 bf , 2 fh, and 1.5 ch:

$$6y_1 + 2y_2 + 1.5y_3 \geq 30$$

- A $20 chair can be produced from resources 1 bf ,1.5 fh and 0.5 ch:

$$y_1 + 1.5y_2 + 0.5y_3 \geq 20$$

- The optimal purchase prices can be solved from the dual problem!

Dakota Example – Comparison of the optimal solutions of the primal and dual problem

PRIMAL

2	0	8			
60	30	20	280		
8	6	1	24	<=	48
4	2	1.5	20	<=	20
2	1.5	0.5	8	<=	8

DUAL

0	10	10			
48	20	8	280		
8	4	2	60	>=	60
6	2	1.5	35	>=	30
1	1.5	0.5	20	>=	20

PRIMAL SENSITIVITY

ıriable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$G$4	PRIMAL	2	0	60	20	4
$H$4		0	-5	30	5	1E+30
$I$4		8	0	20	2.5	5

ınstraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$J$7		24	0	48	1E+30	24
$J$8		20	10	20	4	4
$J$9		8	10	8	2	1.333333333

DUAL SENSITIVITY

Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$N$4	DUAL	0	24	48	1E+30	24
$O$4		10	0	20	4	4
$P$4		10	0	8	2	1.333333333

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$Q$7	<=	60	2	60	20	4
$Q$8	<=	35	0	30	5	1E+30
$Q$9	<=	20	8	20	2.5	5

Extra: sensitivity report summaries

Aalto University
School of Business

Summary: Sensitivity Report for the Objective Function

Final Value

- The value of the decision variables (changing cells) in the optimal solution.

Reduced Cost

- Improvement needed in the objective function coefficient of a zero-valued variable

Objective Coefficient

- The current value of the objective coefficient.

Allowable Increase/Decrease

- Defines the range of the coefficients in the objective function for which the current solution (value of the decision variables or changing cells in the optimal solution) will not change.

Summary: Sensitivity Report for Constraints

Final Value

- The usage of the resource in the optimal solution—the left-hand side of the constraint.

Shadow Price

- The change in the value of the objective function per unit increase in the right-hand-side of the constraint (RHS):

  $$\Delta Z = (\text{Shadow Price})(\Delta \text{RHS})$$

  (Note: only valid if change is within the allowable range)

Constraint R.H. Side

- The current value of the right-hand-side of the constraint.

Allowable Increase/Decrease

- Defines the range of values for the RHS for which the shadow price is valid and hence for which the new objective function value can be calculated.

Extra example on sensitivity analysis

Extra Example

$$\begin{aligned} \text{Min} \quad & 6x_1 + 9x_2 \quad (\$\ \text{cost}) \\ \text{s.t.} \quad & x_1 + 2x_2 \le 8 \\ & 10x_1 + 7.5x_2 \ge 30 \\ & x_2 \ge 2 \\ & x_1, x_2 \ge 0 \end{aligned}$$

Target Cell (Min)

Cell	Name	Original Value	Final Value
$D$3	Obj. (min)	0	27

Adjustable Cells

Cell	Name	Original Value	Final Value
$B$2	x1	0	1.5
$C$2	x2	0	2

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$D$4	Const. 1	5.5	$D$4<=$F$4	Not Binding	2.5
$D$5	Const. 2	30	$D$5>=$F$5	Binding	0
$D$6	Const. 3	2	$D$6>=$F$6	Binding	0

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$2	x1	1.5	0	6	6	6
$C$2	x2	2	0	9	1E+30	4.5

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$D$4	Const. 1	5.5	0	8	1E+30	2.5
$D$5	Const. 2	30	0.6	30	25	15
$D$6	Const. 3	2	4.5	2	2	2

Extra Example

$$\begin{array}{lrl} \text{Min} & 6x_1 + 9x_2 & (\$ \text{ cost}) \\ \text{s.t.} & x_1 + 2x_2 \le 8 & \\ & 10x_1 + 7.5x_2 \ge 30 & \\ & x_2 \ge 2 & \\ & x_1, x_2 \ge 0 & \end{array}$$

Target Cell (Min)

Cell	Name	Original Value	Final Value
$D$3	Obj. (min)	0	27

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$2	x1	1.5	0	6	6	6
$C$2	x2	2	0	9	1E+30	4.5

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$D$4	Const. 1	5.5	0	8	1E+30	2.5
$D$5	Const. 2	30	0.6	30	25	15
$D$6	Const. 3	2	4.5	2	2	2

Question (Ranges of optimality)

Suppose the unit cost of x_1 is decreased to $4. Is the current solution still optimal?

What is the value of the objective function when this unit cost is decreased to $4?

Extra Example

$$
\begin{aligned}
\text{Min} \quad & 6x_1 + 9x_2 \quad (\$ \text{ cost}) \\
\text{s.t.} \quad & x_1 + 2x_2 \le 8 \\
& 10x_1 + 7.5x_2 \ge 30 \\
& x_2 \ge 2 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$2	x1	1.5	0	6	6	6
$C$2	x2	2	0	9	1E+30	$\boxed{4.5}$

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$D$4	Const. 1	5.5	0	8	1E+30	2.5
$D$5	Const. 2	30	0.6	30	25	15
$D$6	Const. 3	2	4.5	2	2	2

Question (Ranges of optimality)

How much can the unit cost of x_2 be decreased without concern for the optimal solution changing?

Extra Example

$$
\begin{aligned}
\text{Min} \quad & 6x_1 + 9x_2 \quad (\$\ \text{cost}) \\
\text{s.t.} \quad & x_1 + 2x_2 \le 8 \\
& 10x_1 + 7.5x_2 \ge 30 \\
& x_2 \ge 2 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$2	x1	1.5	0	6	6	6
$C$2	x2	2	0	9	1E+30	4.5

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$D$4	Const. 1	5.5	0	8	1E+30	2.5
$D$5	Const. 2	30	0.6	30	25	15
$D$6	Const. 3	2	4.5	2	2	2

Question (Ranges of feasibility)

If the right-hand side of constraint 3 is increased by 2, what will be the effect on the optimal objective function value?

Extra Example: Answers

- Ranges of optimality #1
  - Yes, the current solution is still optimal since decrease of $2 (= from $6 to $4) is less than $6
  - New objective function value $4*1.5 + $2*9 = $24

- Ranges of optimality #2
  - Max allowed decrease is $4.5

- Ranges of feasibility
  - New objective function value will be $27+ 2*$4.5 =36

Extra example on sensitivity analysis

Example: Steelco

- Steelco uses coal, iron, and labor to produce three types of steel.
- 200 tons of coal, 60 tons of iron and 100 labor hours are available.
- How many tons of each steel type should be produced to maximize total profit?

	Coal required	Iron required	Labor required	Profit/ton
Steel 1	3 tons	1 ton	1 hour	$80
Steel 2	2 tons	0 ton	1 hour	$50
Steel 3	1 ton	1 ton	1 hour	$20

Steelco – Formulation and Answer Report

Let x_i = tons of steel i produced, i=1,2,3.

$$
\begin{array}{llr}
\text{Max} & 80x_1 + 50x_2 + 20x_3 & \text{(Profit)} \\
& 3x_1 + 2x_2 + x_3 \le 200 & \text{(Coal)} \\
& x_1 + x_3 \le 60 & \text{(Iron)} \\
& x_1 + x_2 + x_3 \le 100 & \text{(Labor)} \\
& x_1, x_2, x_3 \ge 0 &
\end{array}
$$

Target Cell (Max)

Cell	Name	Original Value	Final Value
$B$5	Obj. Fcn x1	0	5300

Adjustable Cells

Cell	Name	Original Value	Final Value
$B$3	Dec. Var. x1	0	60
$C$3	Dec. Var. x2	0	10
$D$3	Dec. Var. x3	0	0

Question (Ranges of feasibility)

What are the optimal decision variables values and objective function value?

Steelco– Sensitivity Report

Target Cell (Max)

Cell	Name	Original Value	Final Value
$B$5	Obj. Fcn x1	0	5300

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$3	Dec. Var. x1	60	0	80	1E+30	5
$C$3	Dec. Var. x2	10	0	50	3.333333333	50
$D$3	Dec. Var. x3	0	-10	20	10	1E+30

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$8	Coal LHS	200	25	200	60	20
$B$9	Iron LHS	60	5	60	6.666666667	60
$B$10	Labor LHS	70	0	100	1E+30	30

Question (Ranges of feasibility)

What would profit be if only 40 tons of iron were available?

20t < 60t → shadow price holds → decrease in the profit 20t x 5$/t=$100. New profit 5300-100=5200

Aalto University
School of Business

Steelco– Sensitivity Report

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$3	Dec. Var. x1	60	0	80	1E+30	5
$C$3	Dec. Var. x2	10	0	50	3.333333333	50
$D$3	Dec. Var. x3	0	-10	20	10	1E+30

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$8	Coal LHS	200	25	200	60	20
$B$9	Iron LHS	60	5	60	6.666666667	60
$B$10	Labor LHS	70	0	100	1E+30	30

Question (Reduced cost)

What is the smallest profit per ton of steel 3 that would make it desirable to produce?

Aalto University
School of Business

Use the reduced cost: 20$/ton-(-10$/ton)=30$/ton

Aalto University
School of Business

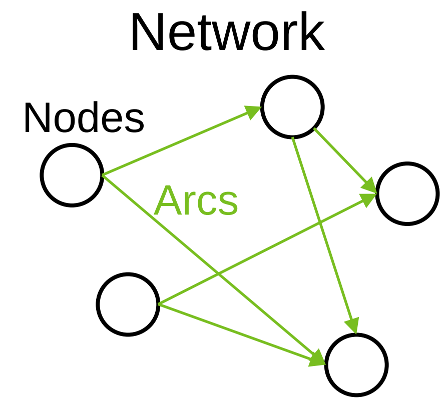
Linear programming – Distribution and Network models

- Transportation problem
- Transshipment problem
- Assignment problem

Linear Programming: Network models

- "How to distribute products from manufacturing to end-customers?"
- "How to assign workers with different skillsets to specific tasks?"

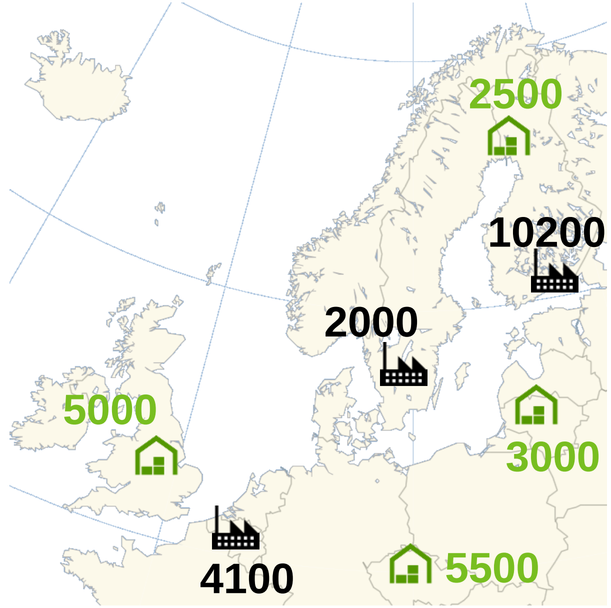
- These decision can be supported by network models
  = Linear Programming (LP) models with a special network structure



- General relationship between LP formulation and network structure
  - Decision variables = Arcs
  - Constrains = Nodes

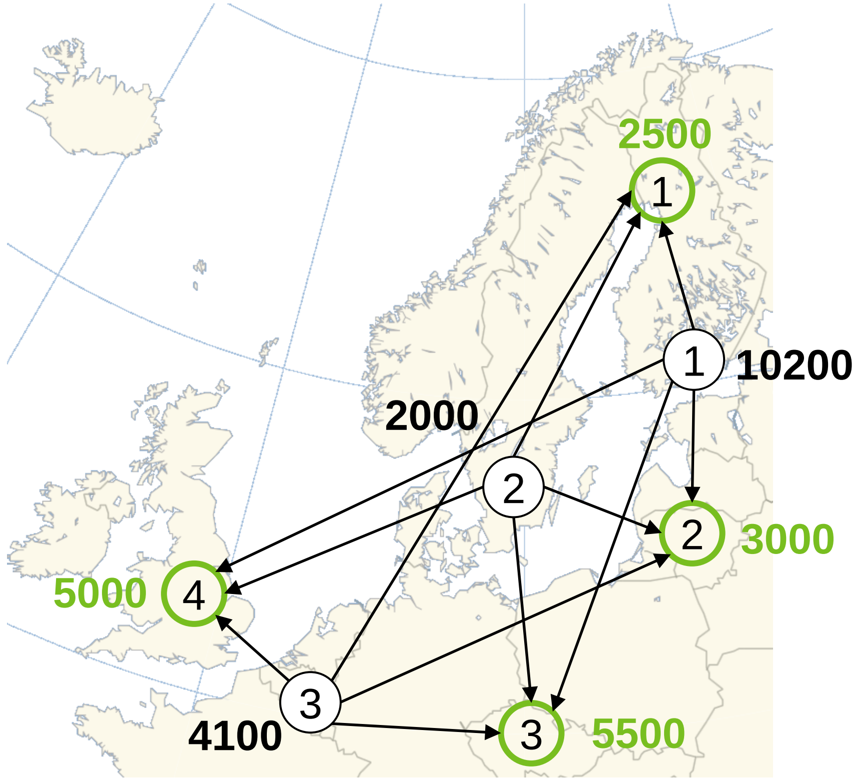
Network LP example: P&P transportation problem

- Pulp&Paper Ltd. produces cardboard at 3 mills
  - Monthly production capacities shown on the map
- From the mills the cardboard is transported to 4 warehouses that supply the customers
  - Monthly customer demand is shown on the map
- P&P wants you to build a network model to select distribution routes so that demand is satisfied with minimal transportation costs



Capacities and **demands** in tons

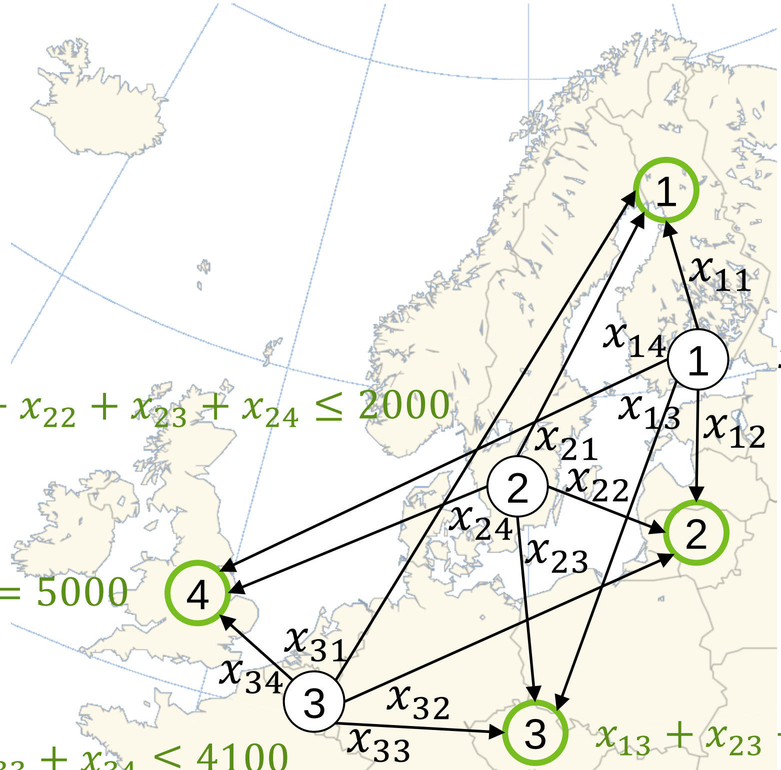
P&P transportation problem: Graphical representation



Capacities and **demands** in tons

P&P transportation problem: LP formulation

Decision variables x_{ij}:
Tons of cardboard transported form mill i to warehouse j



$x_{11} + x_{21} + x_{31} = 2500$

$x_{11} + x_{12} + x_{13} + x_{14} \leq 10200$

$x_{21} + x_{22} + x_{23} + x_{24} \leq 2000$

$x_{12} + x_{22} + x_{32} = 3000$

$x_{14} + x_{24} + x_{34} = 5000$

$x_{13} + x_{23} + x_{33} = 5500$

$x_{31} + x_{32} + x_{33} + x_{34} \leq 4100$

- Question: Interpret the constraints in **green**: Why are they needed?

P&P transportation problem: LP formulation (cont'd)

- Formulating the objective function requires information on unit transportation **costs** on each route (arc)

⇒LP-formulation:

$$\min 53x_{11} + 95x_{12} + 136x_{13} + 160x_{14} + 60x_{21} + 120x_{22} + 132x_{23} + 140x_{24} + 210x_{31} + 190x_{32} + 89x_{33} + 71x_{34}$$

Minimize total transportation costs

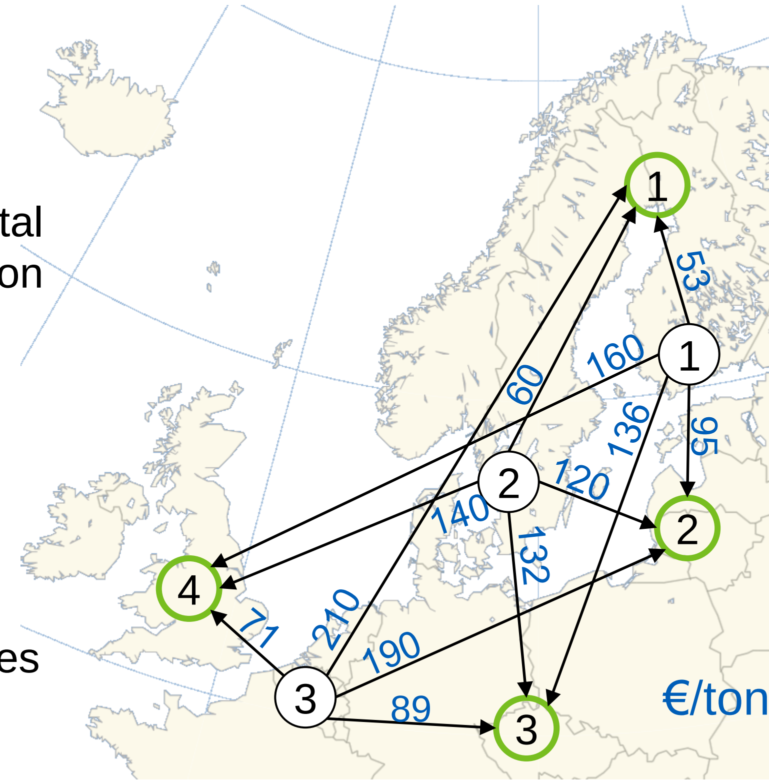
$$\begin{aligned} x_{11} + x_{21} + x_{31} &= 2500 \\ x_{12} + x_{22} + x_{32} &= 3000 \\ x_{13} + x_{23} + x_{33} &= 5500 \\ x_{14} + x_{24} + x_{34} &= 5000 \end{aligned}$$

Satisfy demand

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &\le 10200 \\ x_{21} + x_{22} + x_{23} + x_{24} &\le 2000 \\ x_{31} + x_{32} + x_{33} + x_{34} &\le 4100 \end{aligned}$$

Do not exceed production capacities

$$x_{ij} \ge 0, i = 1, \dots, 3, j = 1, \dots 4$$



P&P transportation problem: Spreadsheet impl.

G9 =SUMPRODUCT($C$5:$F$7;$C$12:$F$14)

	A	B	C	D	E	F	G	H	I
1	**Pulp&Paper Company**								
2									
3	**Transportation cost per ton**		Warehouse						
4			1. Finland	2. Lithuania	3. Czech	4. UK			
5	Mill	1. Finland	53 €	95 €	136 €	160 €			
6		2. Sweden	60 €	120 €	132 €	140 €			
7		3. Belgium	210 €	190 €	89 €	71 €			
8							Total Cost		
9							180 €		
10	**Transported quantity x_{ij} (tons)**		Warehouse						
11			1. Finland	2. Lithuania	3. Czech	4. UK	Total Shipped		Capacity
12	Mill	1. Finland					0	≤	10200
13		2. Sweden	1	1			2	≤	2000
14		3. Belgium					0	≤	4100
15		Total Received	1	1	0	0			
16			=	=	=	=			
17		Demand	2500	3000	5500	5000			
18									
19									
20	Parameters								
21	Decision variables								

Solver Parameters

Set Objective: $G$9

To: Max Min Value Of

By Changing Variable Cells:
$C$12:$F$14

Subject to the Constraints:
$C$15:$F$15 = $C$17:$F$17
$G$12:$G$14 <= $I$12:$I$14

Make Unconstrained Variables Non-Negative

Select a Solving Method: Simplex LP

Solving Method
Select the GRG Nonlinear engine for Solver Problem
Simplex engine for linear Solver Problems, and selec
problems that are non-smooth.

Help

Spreadsheet model available in MyCourses

P&P transportation problem: Optimal solution

G9 =SUMPRODUCT($C$5:$F$7;$C$12:$F$14)

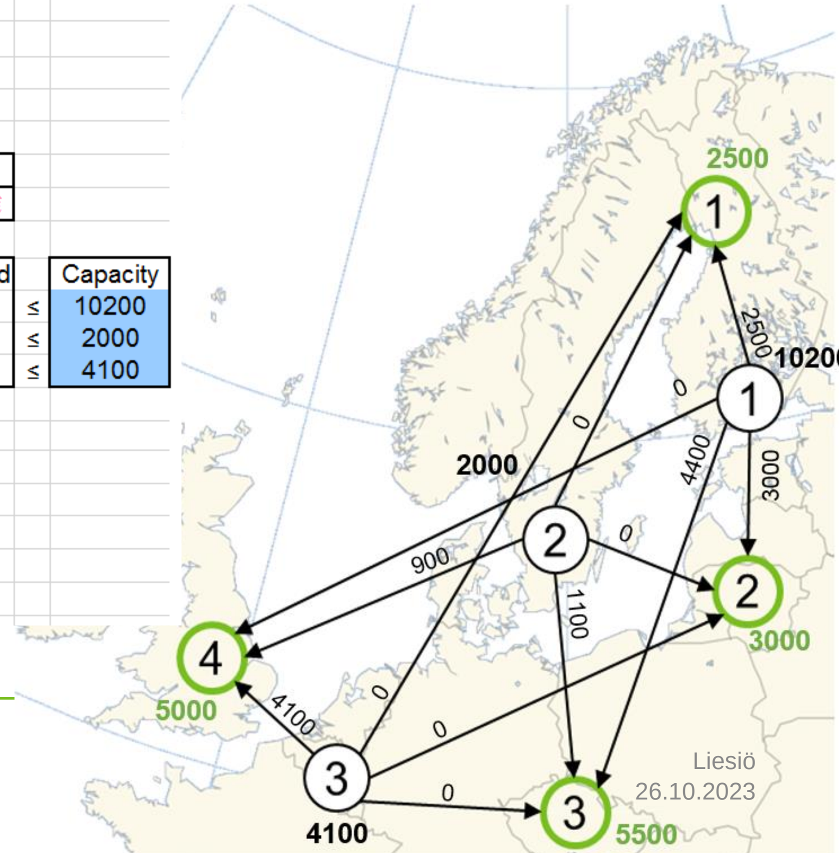
Pulp&Paper Company

Transportation cost per ton		Warehouse			
		1. Finland	2. Lithuania	3. Czech	4. UK
Mill	1. Finland	53 €	95 €	136 €	160 €
	2. Sweden	60 €	120 €	132 €	140 €
	3. Belgium	210 €	190 €	89 €	71 €

Total Cost
1 578 200 €

Transported quantity x_{ij} (tons)		Warehouse						
		1. Finland	2. Lithuania	3. Czech	4. UK	Total Shipped		Capacity
Mill	1. Finland	2500	3000	4400	0	9900	≤	10200
	2. Sweden	0	0	1100	900	2000	≤	2000
	3. Belgium	0	0	0	4100	4100	≤	4100
	Total Received	2500	3000	5500	5000			
		=	=	=	=			
	Demand	2500	3000	5500	5000			

Parameters
Decision variables



Transportation Problem: General Characteristics

- A common problem in logistics is how to transport goods from a set of sources (e.g., plants, warehouses, etc.) to a set of destinations (e.g., warehouses, customers, etc.) with minimum possible cost
- Nodes (Constraints)
  - a set of sources, each with a given supply
  - a set of destinations, each with a given demand
- Arcs (Decision variables)
  - Possible transport routes between sources and destinations, each with a shipping cost
- Objective
  - To determine how much should be shipped from each source node to each destination node so that the total transportation costs are minimized

Transportation Problem: General LP-formulation

x_{ij}: the amount shipped from supply point i to the demand point j.

$$\text{Min} \quad \Sigma_i \Sigma_j c_{ij} x_{ij}$$

$$\begin{aligned} \text{s.t.} \quad & \Sigma_j x_{ij} \le s_i && \text{for each source } i \\ & \Sigma_i x_{ij} = d_j && \text{for each destination } j \\ & x_{ij} \ge 0 && \text{for all } i \text{ and } j \end{aligned}$$

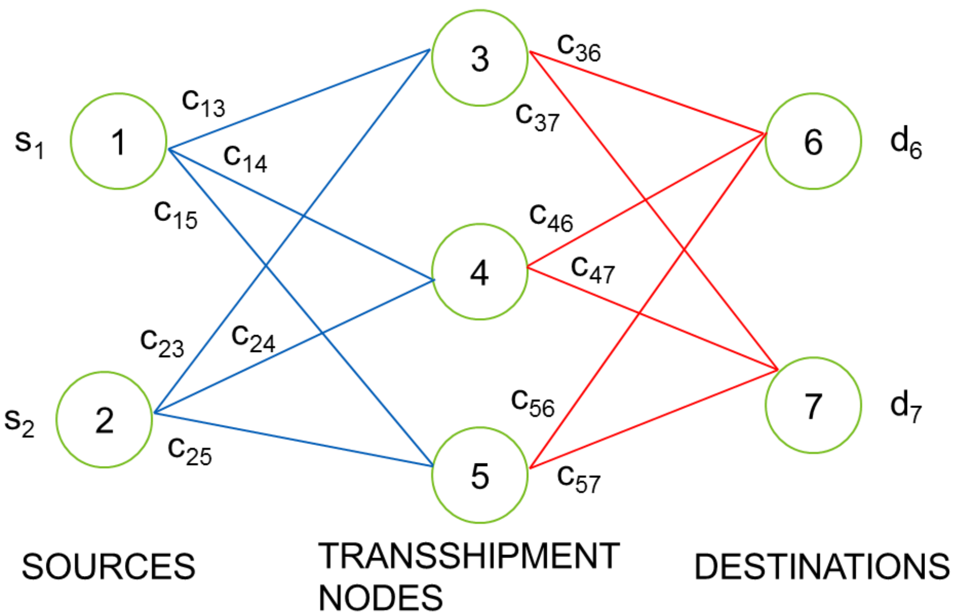
- If total supply ≥ Total demand, then not all supply will be used
- If total supply < Total demand, then the problem is infeasible
  - Redefine the problem: Satisfy the demand as much as possible at minimum cost
    - Every supply node must send its supply.
    - Every demand node receives up to its demand.

Transportation problem: Variations

- Maximize the objective function
  - c_{ij} is then the unit profit obtained by supplying j from i
- Limited route capacities
  - Capacity limitations on arcs can be handled by additional constraints, e.g.,
    - $x_{ij} \leq U_{ij}$ (max. that can be transported)
    - $x_{ij} \geq L_{ij}$ (min. that has to be transported)
- Unacceptable routes
  - It may not always be possible to use all the routes, e.g., no railroad transportation between two cities.
  - Drop the variable, corresponding to an unacceptable route, from the objective function and all constraints
    - Or limit route capacity to zero by setting $U_{ij}=0$

Transshipment Problem

- A transportation problem but shipment may move through intermediate nodes (transshipment nodes) before reaching a particular destination node.



$$
\min \quad c_{13}x_{13} + c_{14}x_{14} + c_{15}x_{15} + c_{23}x_{23} + c_{24}x_{24} + c_{25}x_{25} + c_{36}x_{36} + c_{37}x_{37} + c_{46}x_{46} + c_{47}x_{47} + c_{56}x_{56} + c_{57}x_{57}
$$

$$
\left.\begin{aligned}
x_{13} + x_{14} + x_{15} &\le s_1 \\
x_{23} + x_{24} + x_{25} &\le s_2
\end{aligned}\right\} \text{Sources}
$$

$$
\left.\begin{aligned}
x_{13} + x_{23} &= x_{36} + x_{37} \\
x_{14} + x_{24} &= x_{46} + x_{47} \\
x_{15} + x_{25} &= x_{56} + x_{57}
\end{aligned}\right\} \text{Trans-shipment nodes}
$$

$$
\left.\begin{aligned}
x_{36} + x_{46} + x_{56} &= d_6 \\
x_{37} + x_{47} + x_{57} &= d_7
\end{aligned}\right\} \text{Destinations}
$$

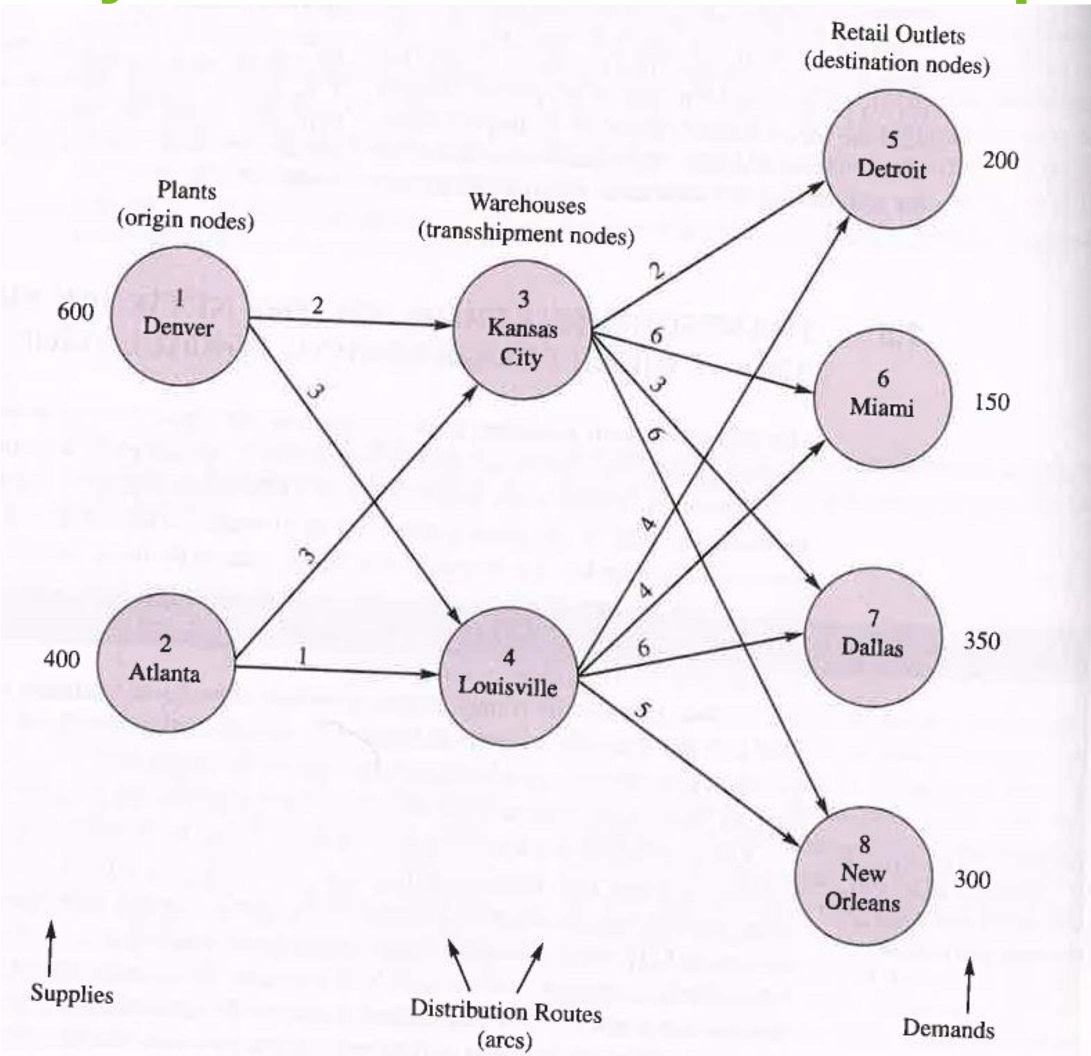
Transhipment example: Ryan Electronics

- Ryan Electronics wants to optimize its component distribution network
  - The firm holds its annual strategy weekend in Aspen
  - Production plants in Denver and Atlanta.
  - After production components are shipped to regional warehouses in Kansas City and Louisville
  - The firm's HQ is located in Denver
  - From regional warehouses the firm supplies its retail outlets in Detroit, Miami, Dallas and New Orleans

Question: Network representation

- Sources?
- Transshipment nodes?
- Destinations?

Ryan Electronics: Network representation



Question: LP formulation

- How many decision variables are needed?
- How many constraints?

Ryan Electronics: LP formulation

$$\text{Min } \overbrace{2x_{13} + 3x_{14}}^{\text{From Denver}} + \overbrace{3x_{23} + 1x_{24}}^{\text{From Atlanta}} + \overbrace{2x_{35} + 6x_{36} + 3x_{37} + 6x_{38}}^{\text{From Kansas City}} + \overbrace{4x_{45} + 4x_{46} + 6x_{47} + 5x_{48}}^{\text{From Louisville}}$$

s.t.

$$\begin{array}{lll}
x_{13} + x_{14} & \leq 600 & \text{Denver (1)} \\
x_{23} + x_{24} & \leq 400 & \text{Atlanta (2)} \\
-x_{13} - x_{23} + x_{35} + x_{36} + x_{37} + x_{38} & = 0 & \text{Kansas City (3)} \\
-x_{14} - x_{24} + x_{45} + x_{46} + x_{47} + x_{48} & = 0 & \text{Louisville (4)} \\
x_{35} + x_{45} & = 200 & \text{Detroit (5)} \\
x_{36} + x_{46} & = 150 & \text{Miami (6)} \\
x_{37} + x_{47} & = 350 & \text{Dallas (7)} \\
x_{38} + x_{48} & = 300 & \text{New Orleans (8)}
\end{array}$$

$x_{ij} \geq 0$ for all i and j

Ryan Electronics: Spreadsheet implementation

C21 =SUMPRODUCT(D6:E7;D12:E13)+SUMPRODUCT(L6:O7;L12:O13)

Unit Cost

		Warehouse (transshipment node)			Retail outlet (destination node)			
		Kansas City	Louisville		Detroit	Miami	Dallas	New Orleans
Plant (source node)	Denver	2	3	Warehouse (transshipment node) Kansas city	2	6	3	6
	Atlanta	3	1	Louisville	4	4	6	5

Shipment Quantity

		Warehouse (transshipment node)				
		Kansas City	Louisville	Total Shipped		Supply
Plant (source node)	Denver	550	50	600	<=	600
	Atlanta	0	400	400	<=	400
	Total Received	550	450			

		Retail outlet (destination node)				
		Detroit	Miami	Dallas	New Orleans	Total Shipped
Warehouse (transshipment node)	Kansas city	200	0	350	0	550
	Louisville	0	150	0	300	450
	Total Received	200	150	350	300	
		=	=	=	=	
	Demand	200	150	350	300	

Total costs

5 200

Solver Parameters

Set Objective: $C$21

To: Max Min Value Of

By Changing Variable Cells:
$D$12:$E$13;$L$12:$O$13

Subject to the Constraints:
$D$14 = $P$12
$E$14 = $P$13
$F$12:$F$13 <= $H$12:$H$13
$L$14:$O$14 = $L$16:$O$16

Make Unconstrained Variables Non-Negative

Select a Solving Method: Simplex LP

Transshipment Problem: General LP Formulation

x_{ik}, x_{kj} represents the shipment from node i to node k and from node k to node j, respectively ($i \in N_{source}$, $k \in N_{tran}$, $j \in N_{dest.}$)

Min $\Sigma_i \Sigma_k c_{ik} x_{ik} + \Sigma_k \Sigma_j c_{kj} x_{kj}$

$\Sigma_k x_{ik} \leq s_i$	for each source $i \in N_{source}$
$\Sigma_i x_{ik} - \Sigma_j x_{kj} = 0$	for each transshipment node $k \in N_{tran}$
$\Sigma_k x_{kj} = d_j$	for each destination $j \in N_{dest.}$
$x_{ik}, x_{kj} \geq 0$	for all i,j,k

Example:

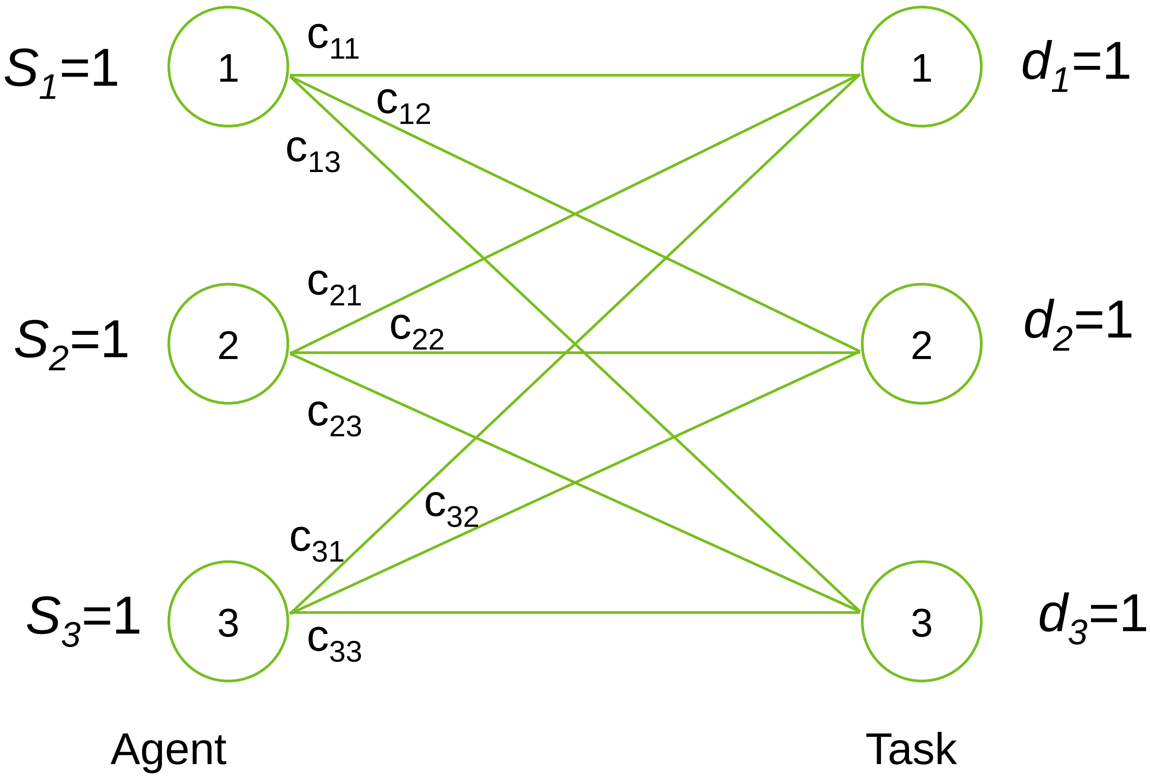
- Source nodes: $N_{source} = \{1,2\}$
- Transshipment nodes: $N_{tran} = \{3,4\}$

$$\sum_i \sum_k c_{ik} x_{ik} = \sum_{i=1}^{2} \sum_{k=3}^{4} c_{ik} x_{ik} = \sum_{i=1}^{2} (c_{i3} x_{i3} + c_{i4} x_{i4})$$
$$= (c_{13} x_{13} + c_{14} x_{14}) + (c_{23} x_{23} + c_{24} x_{24})$$

Assignment Problem

- The problem of assigning agents (people, machines) to a set of tasks is called an assignment problem
  - Problem components
    - a set of agents
    - a set of tasks
    - a cost table (cost associated with each agent performing each task)
  - Objective: Allocate agents to the tasks so that all tasks are performed at the minimum possible cost
  - A special case of a transportation problem

Assignment Problem: Network representation and general LP formulation



$$\text{Min } \sum_{i}\sum_{j} c_{ij} x_{ij}$$

$$\text{s.t. } \sum_{j} x_{ij} \le 1, \text{ for each agent } i$$

$$\sum_{i} x_{ij} = 1, \text{ for each task } j$$

$$x_{ij} \ge 0, \text{ for all } i \text{ and } j.$$

Assignment Problem Example: Swim team

- Help the coach of a swim team to assign swimmers to a 200-yard medley relay team
  - Four swimmers swimming 50 yards using one of the four strokes
- Data on the swimmers (time in seconds for 50 yeards):

	Backstroke	Breaststroke	Butterfly	Freestyle
Carl	37.7	43.4	33.3	29.2
Chris	32.9	33.1	28.5	26.4
David	33.8	42.2	38.9	29.6
Tony	37.0	34.7	30.4	28.5
Ken	35.4	41.8	33.6	31.1

Question: LP formulation

- How many constraints and decision variables?
- What would be the objective function?

Swim team: LP Formulation

Let $x_{ij} = 1$ if swimmer i swims stroke j; 0 otherwise
t_{ij} = time of swimmer i in stroke j

Minimize total time = $\sum_i \sum_j t_{ij} x_{ij}$

subject to

(each stroke swum) $\sum_i x_{ij} = 1$ for each stroke j

(each swimmer swims 1) $\sum_j x_{ij} \leq 1$ for each swimmer i

$x_{ij} \geq 0$ for all i and j.

Swim team: Spreadsheet Formulation

Best Times	Backstroke	Breastroke	Butterfly	Freestyle			
Carl	37.7	43.4	33.3	29.2			
Chris	32.9	33.1	28.5	26.4			
David	33.8	42.2	38.9	29.6			
Tony	37.0	34.7	30.4	28.5			
Ken	35.4	41.8	33.6	31.1			
Assignment	Backstroke	Breastroke	Butterfly	Freestyle			
Carl	0	0	0	1	1	<=	1
Chris	0	0	1	0	1	<=	1
David	1	0	0	0	1	<=	1
Tony	0	1	0	0	1	<=	1
Ken	0	0	0	0	0	<=	1
	1	1	1	1	Time	=	126.2
	=	=	=	=			
	1	1	1	1			

Optimal assignment

- Backstroke – David
- Breaststroke – Tony
- Butterfly – Chris
- Freestyle – Carl

(Ken is the towel boy)

Assignment Problem: Variations

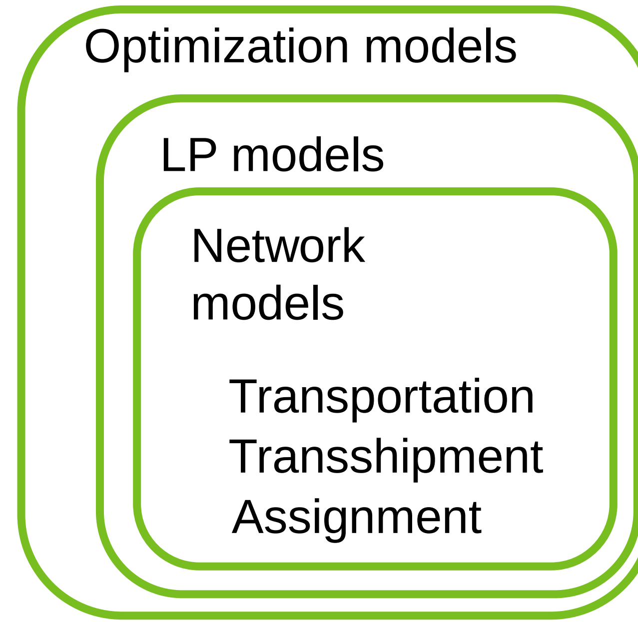
$$\begin{aligned}
\text{Min} \quad & \sum_i \sum_j c_{ij} x_{ij} & \\
\text{s.t.} \quad & \sum_j x_{ij} \le 1 & \text{for each agent } i \\
& \sum_i x_{ij} = 1 & \text{for each task } j \\
& x_{ij} \ge 0 & \text{for all } i \text{ and } j.
\end{aligned}$$

- Certain agents are unable to perform certain tasks
- There are more tasks than agents (some tasks will not be done)
- There are more agents than tasks (some agents will not work)
- An agent can be assigned to perform more than one task
- A task can be performed jointly by more than one agents

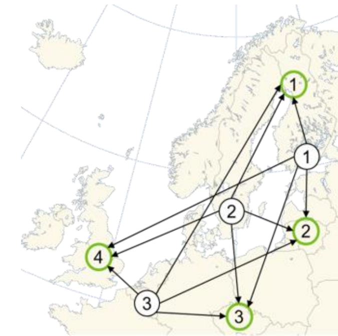
Summary: Network LP Models

- Network Components and their LP equivalents
  - Arc = decision variable
  - Node = constraint:

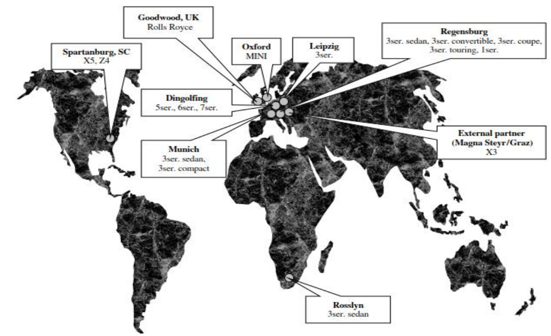
    (flow in) – (flow out) {=, ≥,≤,} constant
  - Objective function = linear function of arc flows
- Properties:
  - Fast to solve (good LP solvers detect and exploit network structure)
  - Illustrative graphical representations
  - Provide integer solutions without the need for explicit *integrality constraints* used in integer linear programming



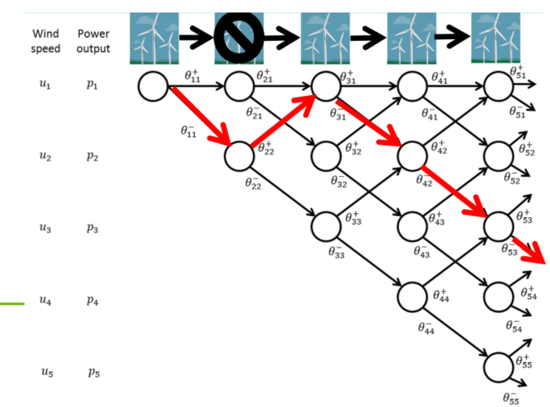
Network models: Application examples



- Sheeting network optimization at “Nordic Wood Processing Company”
  - Customized student business project



- Strategic production planning at BMW[1]
  - Optimize production allocation, supply of materials, distribution with a 12-year planning horizon



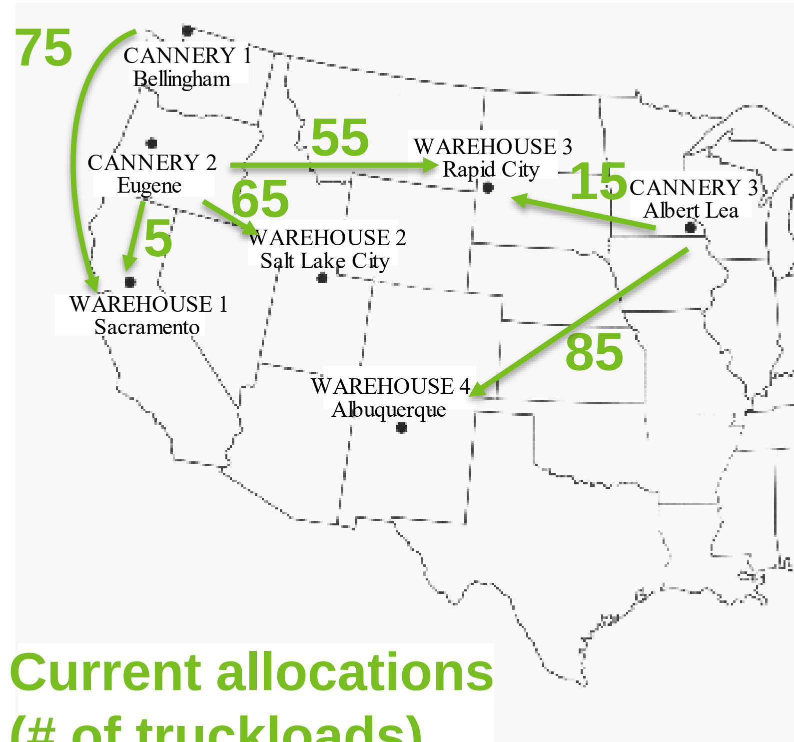
- Siting of off-shore wind farms[2]
  - Economic and environmental objectives
  - Network models used to capture wake interactions

Articles available on MyCourses:
1) Fleischmann, B., Ferber, S., Henrich, P., 2006, *Interfaces*.
2) Cranmer, A., Baker, E., Liesiö, J., Salo, A., 2018, *European Journal of Operation Research*.

Extra network LP examples: Transportation problem

P&T Company Distribution Problem

- P&T company (producer of canned peas) is unhappy with their total shipping costs
  - Peas are prepared at three canneries and shipped by truck to four distributing warehouses
  - Up to now some intuitive guidelines have been used to determine the shipment amounts from canneries to warehouses

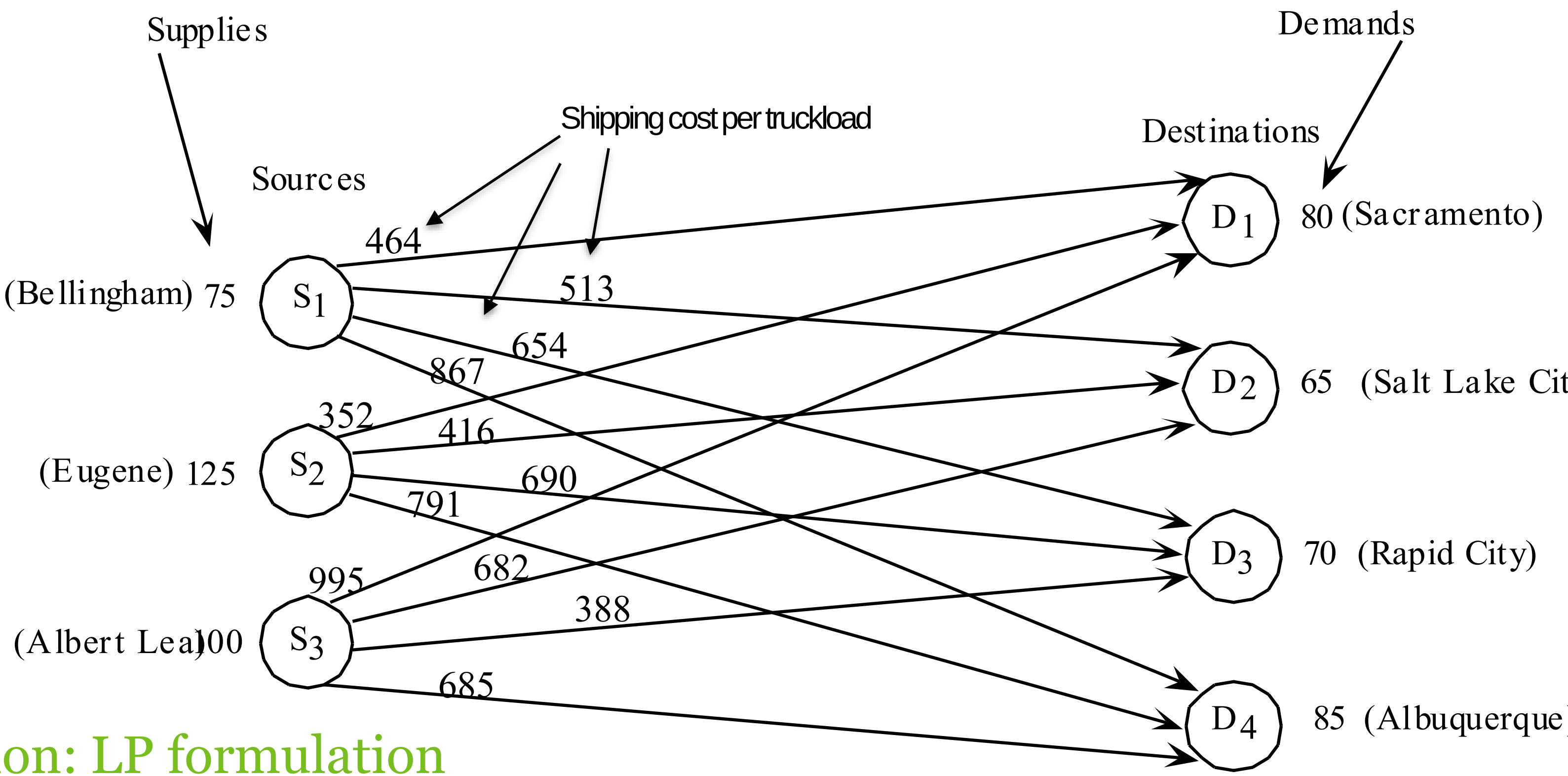


Current allocations (# of truckloads)

Question: Network representation

- What would be the
  - sources and their supplies?
  - destinations and their demands?

P&T Company: Network Representation



Question: LP formulation

- What would be the decision variables, objective function, constraints?

P&T Company: Linear Programming formulation

x_{ij}: the number of truckloads to ship from cannery i to warehouse j ($i = 1, 2, 3; j = 1, 2, 3, 4$)

$$\text{Minimize Cost} = \$464x_{11} + \$513x_{12} + \$654x_{13} + \$867x_{14} + \$352x_{21} + \$416x_{22} + \$690x_{23} + \$791x_{24} + \$995x_{31} + \$682x_{32} + \$388x_{33} + \$685x_{34}$$

subject to

Cannery 1: $x_{11} + x_{12} + x_{13} + x_{14} \leq 75$

Cannery 2: $x_{21} + x_{22} + x_{23} + x_{24} \leq 125$

Cannery 3: $x_{31} + x_{32} + x_{33} + x_{34} \leq 100$

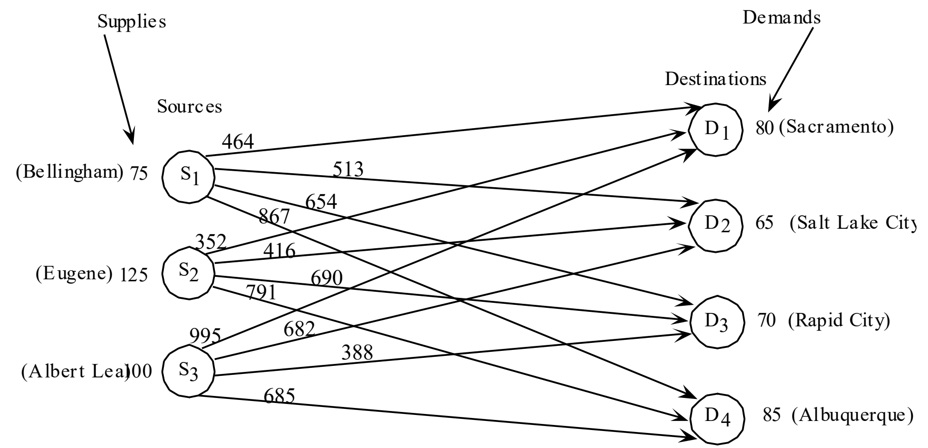
Warehouse 1: $x_{11} + x_{21} + x_{31} = 80$

Warehouse 2: $x_{12} + x_{22} + x_{32} = 65$

Warehouse 3: $x_{13} + x_{23} + x_{33} = 70$

Warehouse 4: $x_{14} + x_{24} + x_{34} = 85$

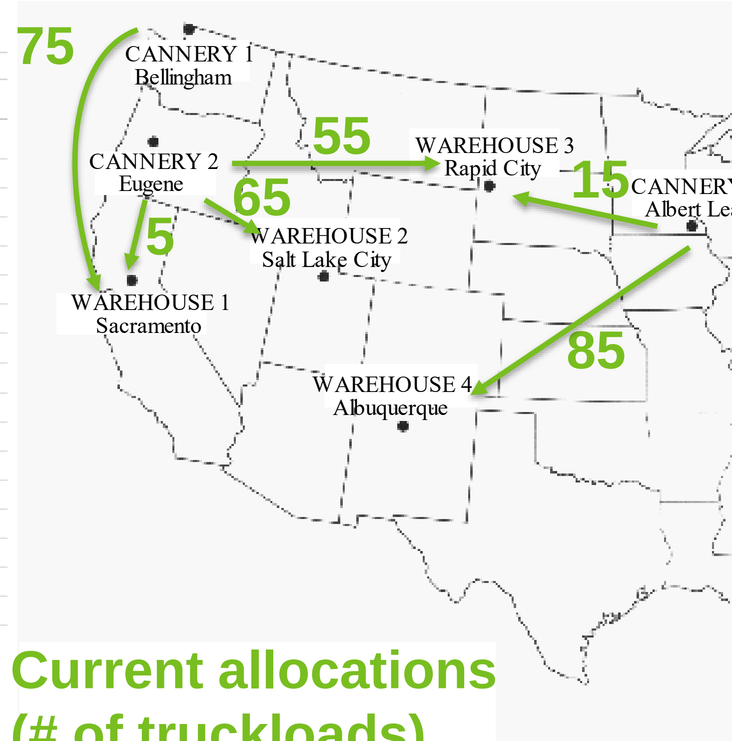
Non-negativity: $x_{ij} \geq 0$ ($i = 1, 2, 3;\ j = 1, 2, 3, 4$)



P&T Company: Spreadsheet Formulation

H9 =SUMPRODUCT($D$5:$G$7;$D$12:$G$14)

	A	B	C	D	E	F	G	H	I	J
1	**P&T Co. Distribution Problem**									
2										
3		**Unit Cost**		Destination (Warehouse)						
4				Sacramento	Salt Lake City	Rapid City	Albuquerque			
5		Source	Bellingham	$464	$513	$654	$867			
6		(Cannery)	Eugene	$352	$416	$690	$791			
7			Albert Lea	$995	$682	$388	$685			
8								Total Cost		
9								$165 595		
10		**Shipment Quantity**		Destination (Warehouse)						
11		**(Truckloads)**		Sacramento	Salt Lake City	Rapid City	Albuquerque	Total Shipped		Supply
12		Source	Bellingham	75	0	0	0	75	<=	75
13		(Cannery)	Eugene	5	65	55	0	125	<=	125
14			Albert Lea	0	0	15	85	100	<=	100
15			Total Received	80	65	70	85			
16				=	=	=	=			
17			Demand	80	65	70	85			



Current allocations (# of truckloads)

P&T Company: Optimal Solution

H9 =SUMPRODUCT($D$5:$G$7;$D$12:$G$14)

P&T Co. Distribution Problem

		Sacramento	Salt Lake City	Rapid City	Albuquerque			
Unit Cost		Destination (Warehouse)						
		Sacramento	Salt Lake City	Rapid City	Albuquerque			
Source (Cannery)	Bellingham	$464	$513	$654	$867			
	Eugene	$352	$416	$690	$791			
	Albert Lea	$995	$682	$388	$685			
						Total Cost		
						$152 535		
Shipment Quantity		Destination (Warehouse)						
(Truckloads)		Sacramento	Salt Lake City	Rapid City	Albuquerque	Total Shipped		Supply
Source (Cannery)	Bellingham	0	20	0	55	75	<=	75
	Eugene	80	45	0	0	125	<=	125
	Albert Lea	0	0	70	30	100	<=	100
	Total Received	80	65	70	85			
		=	=	=	=			
	Demand	80	65	70	85			

Solver Parameters

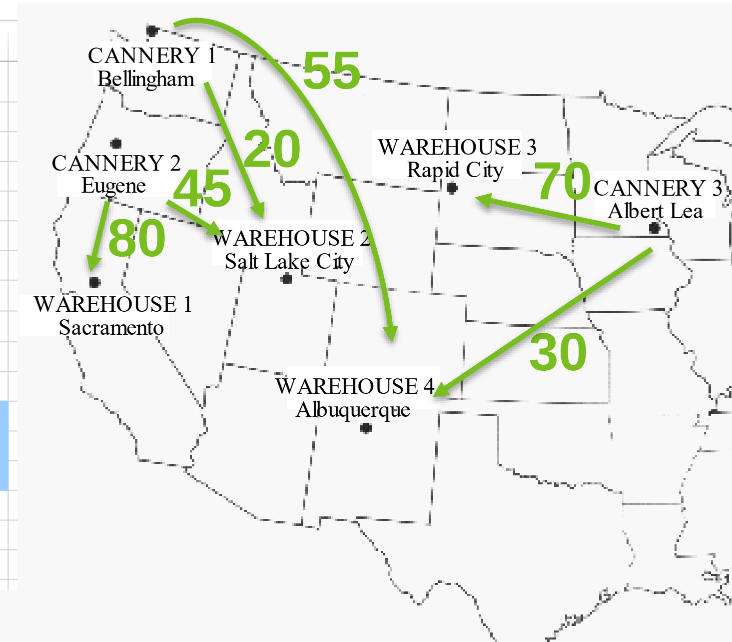
Set Objective: $H$9

To: Max / **Min** / Value Of: 0

By Changing Variable Cells: $D$12:$G$14

Subject to the Constraints:
$D$15:$G$15 = $D$17:$G$17
$H$12:$H$14 <= $J$12:$J$14

Add



Optimal allocation (# of truckloads)

Aalto University
School of Business

Mixed Integer Linear Programming (MILP)

- *Types of Integer Linear Programming Models*
- *Feasible regions and graphical solution*
- *LP relaxation*
- *Special 0-1 constraints*
- *Computer solution*
- *Cautionary notes on sensitivity analysis and rounding*
- *(M)ILP applications and formulations*

Integers

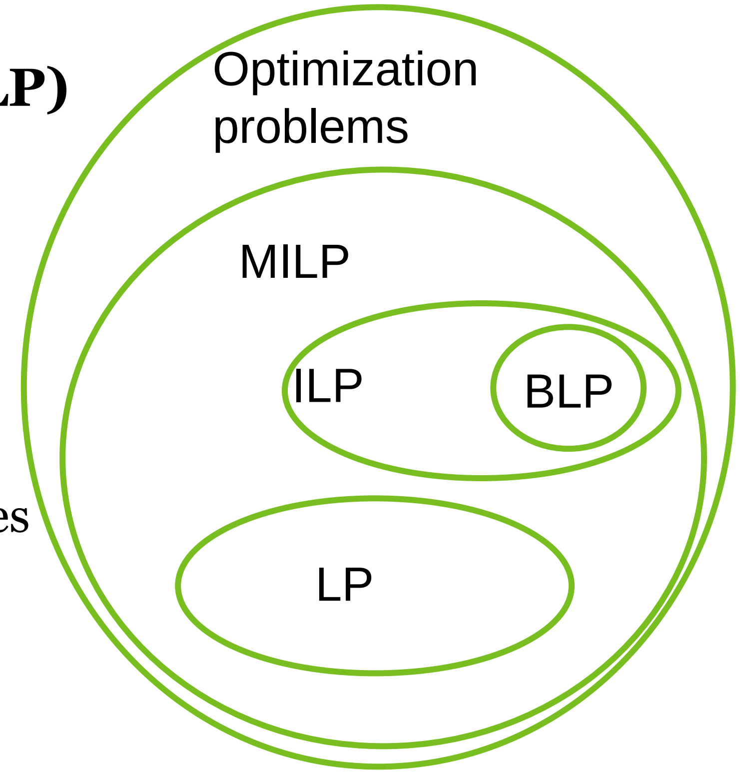
$$\ldots -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5 \ldots$$

group often denoted with $\mathbb{Z}$,
e.g. $x \in \mathbb{Z}$

$\in$ reads “in” or “is element of the set”

Types of Integer Linear Programming Models

- **Pure Integer Linear Programming (ILP)**
  - All the decision variables are integers
- **Mixed Integer Linear Programming (MILP)**
  - Some of the decision variables are integers
- **Binary Linear Programming (BLP)**
  - Decision variables restricted to be binary values (i.e. 0 or 1)
  - Sometimes the term zero-one linear programming (ZOLP) is used
  - Pure BLP: all the decision variables binary
  - Mixed BLP: some decision variables binary



Examples of ILP problems

BLP

$$
\begin{aligned}
\text{Max} \quad & 3x_1 + 2x_2 \\
\text{s.t.} \quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1, x_2 \in \{0,1\}
\end{aligned}
$$

(pure) ILP

$$
\begin{aligned}
\text{Max} \quad & 3x_1 + 2x_2 \\
\text{s.t.} \quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1, x_2 \ge 0 \\
& x_1, x_2 \text{ integer}
\end{aligned}
$$

MILP

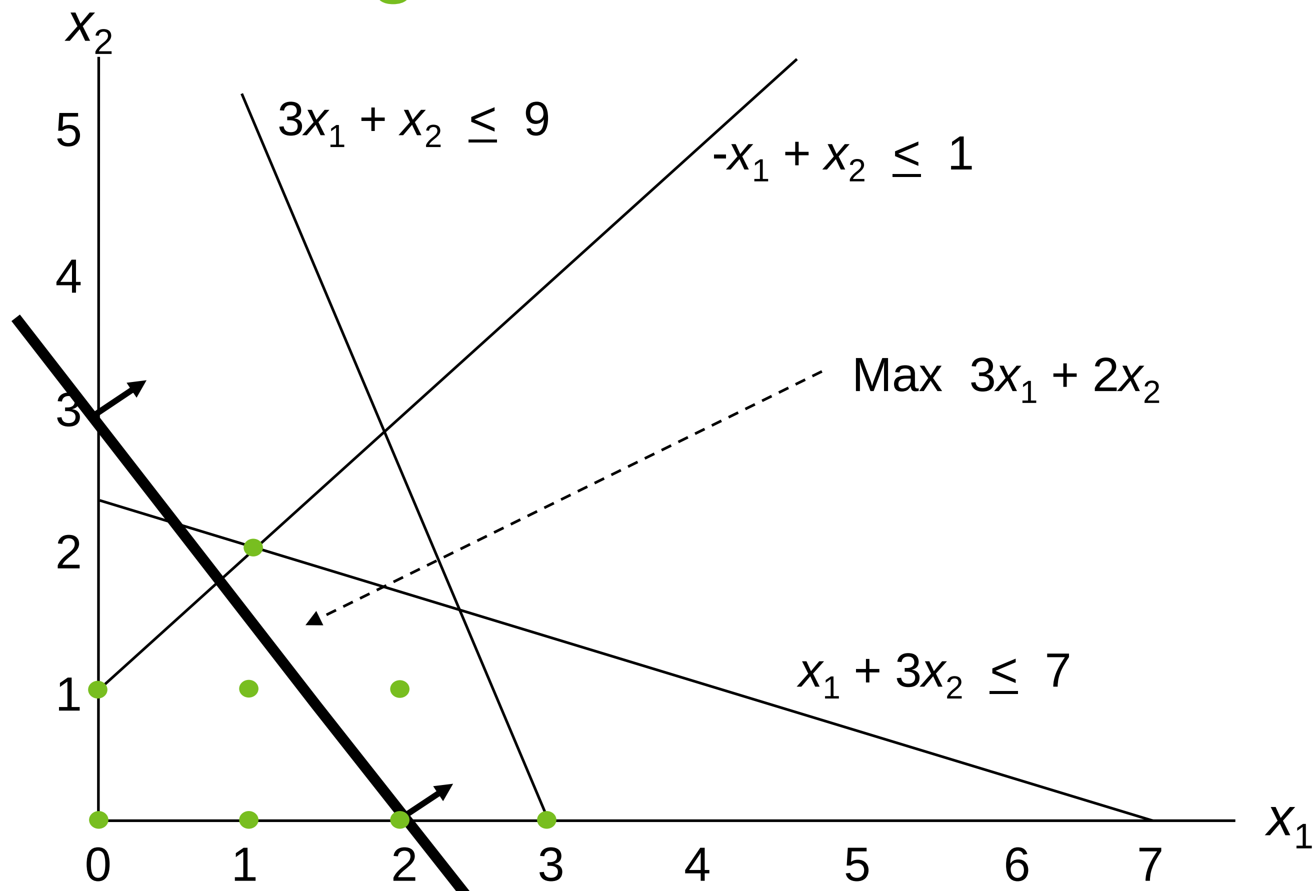
$$
\begin{aligned}
\text{Max} \quad & 3x_1 + 2x_2 \\
\text{s.t.} \quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1, x_2 \ge 0 \\
& x_1 \text{ integer}
\end{aligned}
$$

∈ reads “in” or “element of the set”

Example of the feasible region: Pure ILP

$$\begin{aligned}
\text{Max} \quad & 3x_1 + 2x_2 \\
\text{s.t.} \quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1, x_2 \ge 0 \\
& x_1, x_2 \text{ integer}
\end{aligned}$$

Feasible region



Example of the feasible region: MILP

Max $3x_1 + 2x_2$

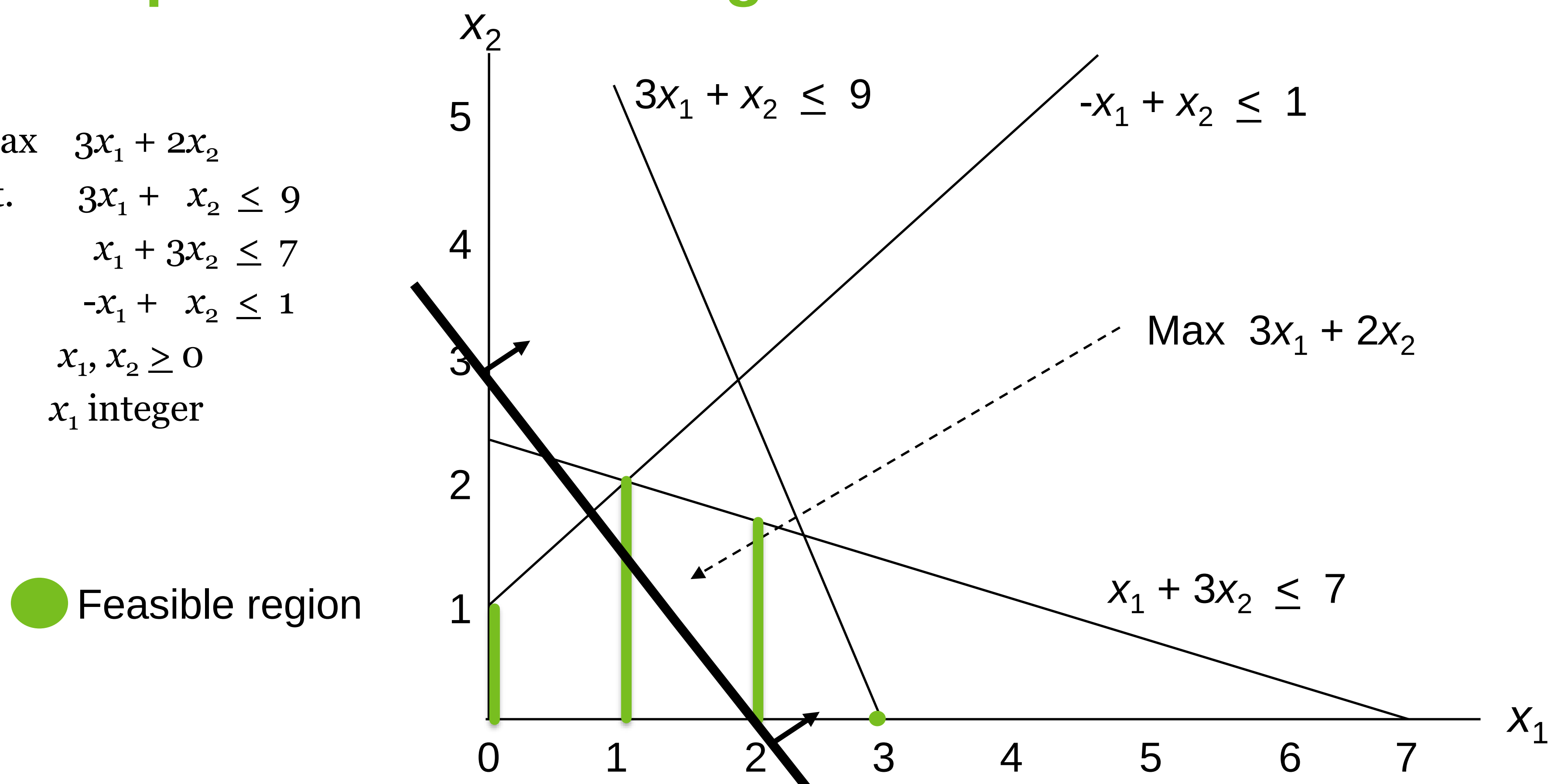
s.t. $3x_1 + x_2 \leq 9$

$x_1 + 3x_2 \leq 7$

$-x_1 + x_2 \leq 1$

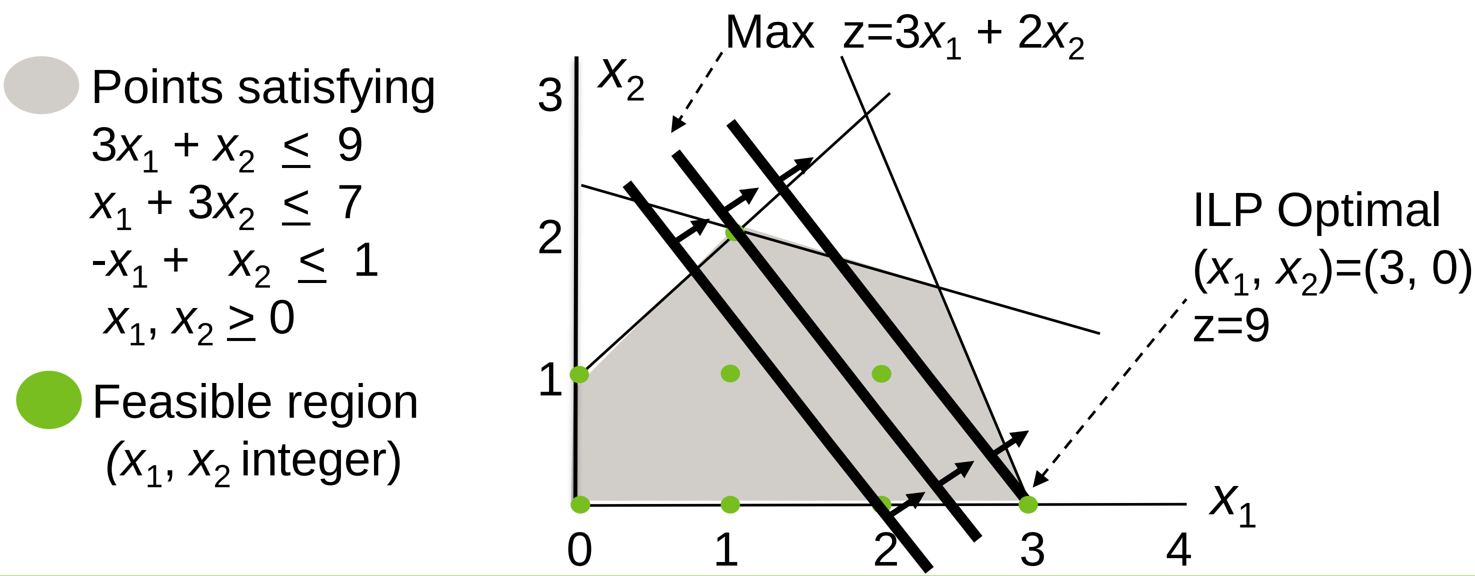
$x_1, x_2 \geq 0$

x_1 integer



Graphical Method for Solving MILP Problems

- Optimal solutions to MILP problems with two decision variables can be found by applying the graphical solution method for LPs
  - Caution: feasible region not equal to the LP feasible region!



Graphical Method for Solving ILP Problems (Cont'd)

MILP example:

Max $3x_1 + 2x_2$

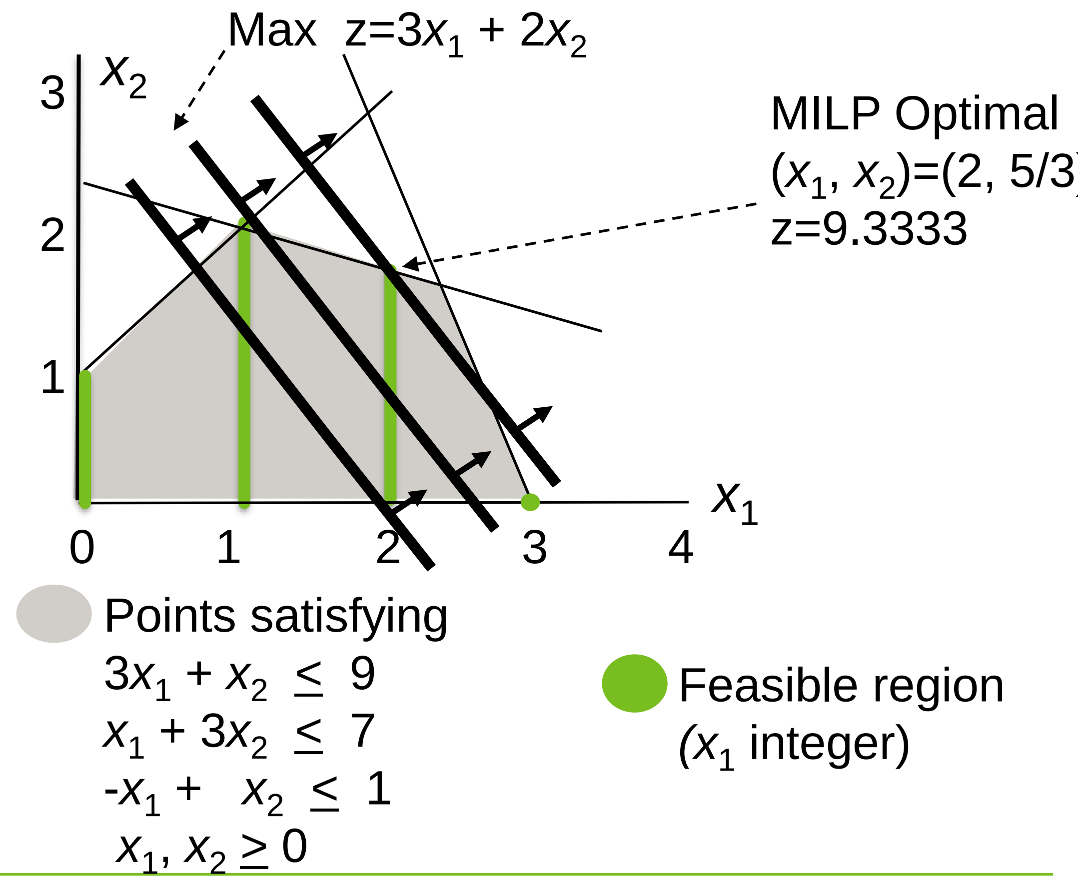
s.t. $3x_1 + x_2 \leq 9$

$x_1 + 3x_2 \leq 7$

$-x_1 + x_2 \leq 1$

$x_1, x_2 \geq 0$

x_1 integer



Linear programming relaxation

- The LP relaxation of a (M)ILP problem is the LP problem obtained when all the integrality constraints are removed

(pure) ILP

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ \text{s.t.} \quad & 3x_1 + x_2 \le 9 \\ & x_1 + 3x_2 \le 7 \\ & -x_1 + x_2 \le 1 \\ & x_1, x_2 \ge 0 \\ & x_1, x_2 \text{ integer} \end{aligned}$$

MILP

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ \text{s.t.} \quad & 3x_1 + x_2 \le 9 \\ & x_1 + 3x_2 \le 7 \\ & -x_1 + x_2 \le 1 \\ & x_1, x_2 \ge 0 \\ & x_2 \text{ integer} \end{aligned}$$

LP relaxation

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ \text{s.t.} \quad & 3x_1 + x_2 \le 9 \\ & x_1 + 3x_2 \le 7 \\ & -x_1 + x_2 \le 1 \\ & x_1, x_2 \ge 0 \end{aligned}$$

Linear programming relaxation (cont'd)

- The LP relaxation of a BLP problem is the LP problem obtained when all the integrality constraints are removed

BLP

$$
\begin{aligned}
\text{Max}\quad & 3x_1 + 2x_2 \\
\text{s.t.}\quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1, x_2 \in \{0,1\}
\end{aligned}
$$

LP relaxation

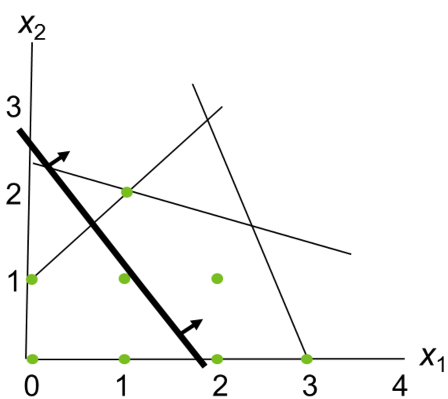
$$
\begin{aligned}
\text{Max}\quad & 3x_1 + 2x_2 \\
\text{s.t.}\quad & 3x_1 + x_2 \le 9 \\
& x_1 + 3x_2 \le 7 \\
& -x_1 + x_2 \le 1 \\
& x_1 \le 1 \\
& x_2 \le 1 \\
& x_1, x_2 \ge 0
\end{aligned}
$$

∈ reads "in" or "element of the set"

Linear programming relaxation (cont'd)

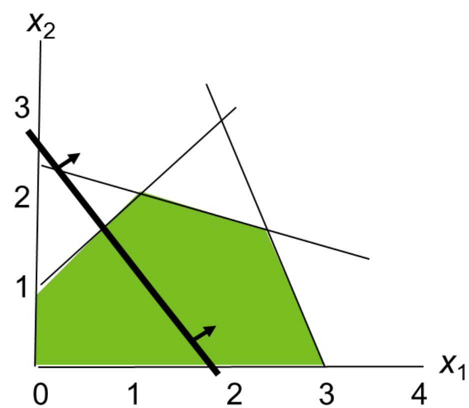
(pure) ILP

$$
\begin{array}{ll}
\text{Max} & 3x_1 + 2x_2 \\
\text{s.t.} & 3x_1 + x_2 \le 9 \\
 & x_1 + 3x_2 \le 7 \\
 & -x_1 + x_2 \le 1 \\
 & x_1, x_2 \ge 0 \\
 & x_1, x_2 \text{ integer}
\end{array}
$$



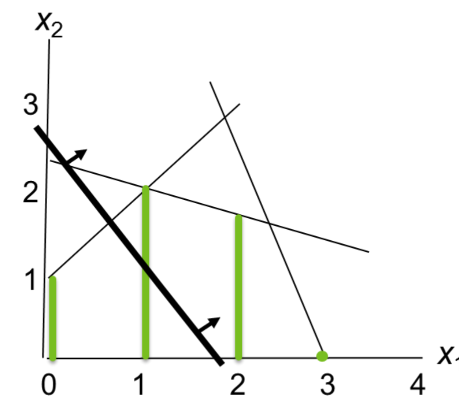
LP relaxation

$$
\begin{array}{ll}
\text{Max} & 3x_1 + 2x_2 \\
\text{s.t.} & 3x_1 + x_2 \le 9 \\
 & x_1 + 3x_2 \le 7 \\
 & -x_1 + x_2 \le 1 \\
 & x_1, x_2 \ge 0
\end{array}
$$



MILP

$$
\begin{array}{ll}
\text{Max} & 3x_1 + 2x_2 \\
\text{s.t.} & 3x_1 + x_2 \le 9 \\
 & x_1 + 3x_2 \le 7 \\
 & -x_1 + x_2 \le 1 \\
 & x_1, x_2 \ge 0 \\
 & x_1 \text{ integer}
\end{array}
$$



Understanding the implications of relaxing (integrality) constraints

I

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ 3x_1 + x_2 &\leq 9 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \\ x_1, x_2 & \text{ integer} \end{aligned}$$

II

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ 3x_1 + x_2 &\leq 9 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

III

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ 3x_1 + x_2 &\leq 9 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\in \{0,1\} \end{aligned}$$

IV

$$\begin{aligned} \text{Max} \quad & 3x_1 + 2x_2 \\ 3x_1 + x_2 &\leq 9 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \\ x_2 & \text{ integer} \end{aligned}$$

Question:

- Rank problems I-IV with regard to their optimal objective function value (i.e., the highest, 2nd highest,…)

BLP example: Product portfolio selection

- Metropolitan Microwaves, an electronic appliances store, is planning to include new product lines to its selection
  - The company has identified seven new possible product lines:

	Product line	Initial investment ($)	Floor space (m^2)	Expexted rate of return (%)
1	Tablets	6,000	125	8.1
2	Laptops	2,000	150	9
3	Workstations	20,000	200	11
4	VR-equipment	14,000	40	10.2
5	Phones	15,000	40	10.5
6	Video Games	2,000	20	14.1
7	Gaming consoles	32,000	100	13.2

  - The company has $45,000 to invest and 420 sq. ft. of floor space available
- A management scientist developed an integer linear programming model to support this decision, but she left for the academia and only her notes about the model remain

BLP example: Product portfolio selection (Cont'd)

"Model Notes"

Max $(6*1.081)x_1 + (2*1.09)x_2 + \ldots$

s.t. A. $125x_1 + 150\,x_2 + \ldots + 100\,x_7 \leq 420$

B. $6x_1 + \ldots + 32x_7 \leq 45$

C. $x_4 + x_5 \leq 1$

D. $x_6 \leq x_7$

E. $2x_3 \leq x_1 + x_2$

F. $x_1 + x_2 + \ldots + x_7 \geq 3$

$x_1, \ldots, x_7 \in \{0,1\}$

Data

	Product line	**Initial investment ($)**	**Floor space (m^2)**	**Expected rate of return (%)**
1	Tablets	6,000	125	8.1
2	Laptops	2,000	150	9
3	Workstations	20,000	200	11
4	VR-equipment	14,000	40	10.2
5	Phones	15,000	40	10.5
6	Video Games	2,000	20	14.1
7	Gaming consoles	32,000	100	13.2

Question

- Interpret the meaning of the decision variables and the constraints

BLP: Special 0-1 Constraints

- Since binary variables only provide two choices, they are ideal for modelling yes-or-no (go/no-go, continue/discard, etc.) decisions
- Constraints can then be used to capture logical dependencies between these decisions
- Examples:
  - $x_i = 1$, $i=1,\ldots,n$ if and only if project i is started, otherwise $x_i = 0$
  - At most k out of n projects can be started:

    $$\Sigma_i x_i \leq k$$

  - Project j is conditional on project i:

    $$x_j - x_i \leq 0$$

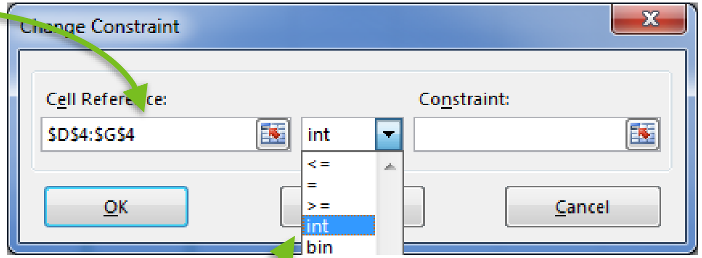
  - Projects i and j are mutually exclusive:

    $$x_i + x_j \leq 1$$

  - ...

Computer Solution to (M)ILP problems

- In Excel Solver (2010-16)
  - Add an additional constraint that tells solver which decision variables are required to take integer ('int') or binary ('bin') values
  - Choose algorithm "Simplex LP" as before
    - This tells Solver that problem is linear, i.e. ILP, MILP, or BLP (not some non-linear optimization problem with integer variables)



- The Simplex algorithm is only for solving LP problems despite the misleading naming of algorithms used in Excel Solver
- Algorithms for (M)ILP problems: Branch-and-bound, Cutting plane,..
  - These algorithms often use the Simplex for solving sub-problems

Branch-and-bound (B&B) algorithm: Example*

ILP problem:

$$\text{Max} \quad z = 5x_1 + 4x_2$$

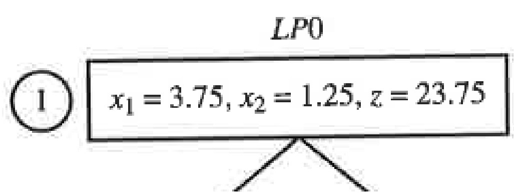
$$\text{s.t.} \quad x_1 + x_2 \leq 5$$

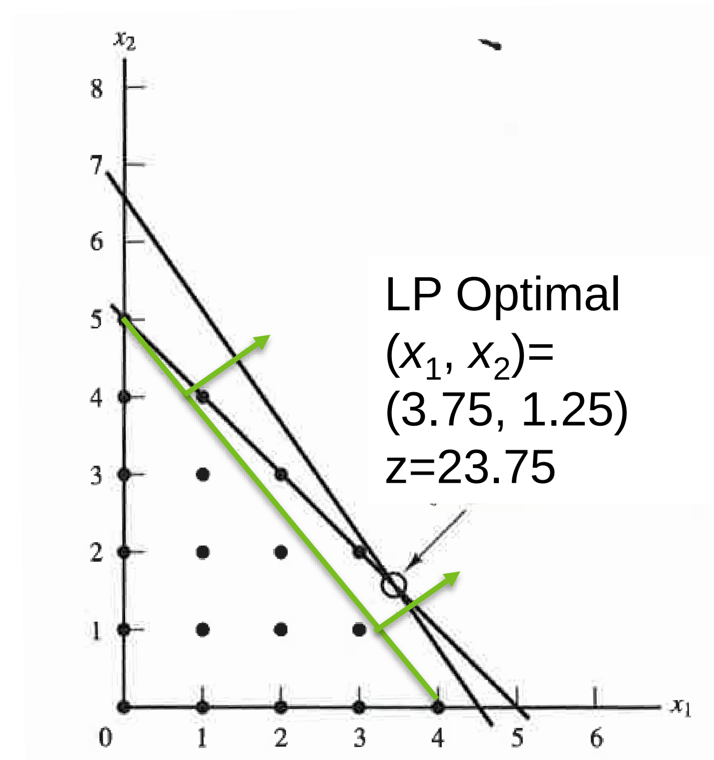
$$10x_1 + 6x_2 \leq 45$$

$$x_1, x_2 \geq 0 \text{ and integer}$$

Step 1: Bounding

- Solve the LP relaxation with Simplex
  - Gives an upper bound to the optimal value (cf. implications of relaxation)



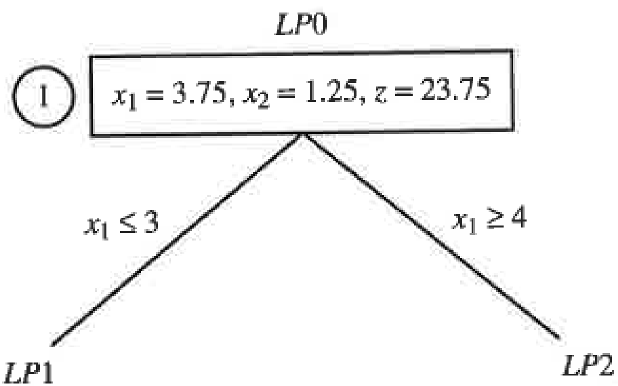


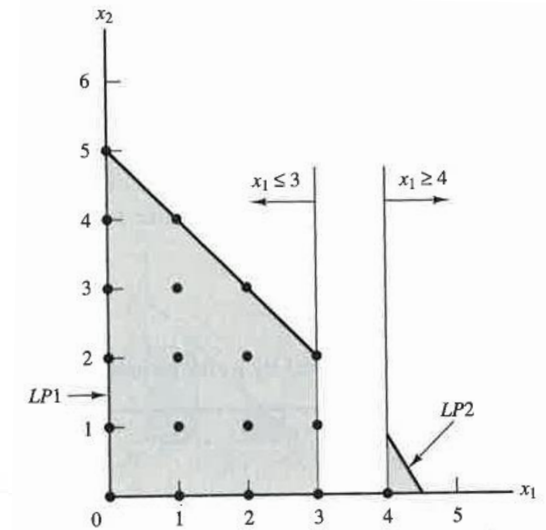
*) Example from Taha 1997: Operations research – An introduction, Prentice-Hall

Branch-and-bound (B&B) algorithm: Example

Step 2: Branching

- Add constraints on one variable that did not have an integer value in the optimal solution





LP1

$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 4x_2 \\
\text{s.t.} \quad & x_1 + x_2 \le 5 \\
& 10x_1 + 6x_2 \le 45 \\
& x_1 \le 3 \\
& x_1, x_2 \ge 0 \\
& \text{and integer}
\end{aligned}
$$

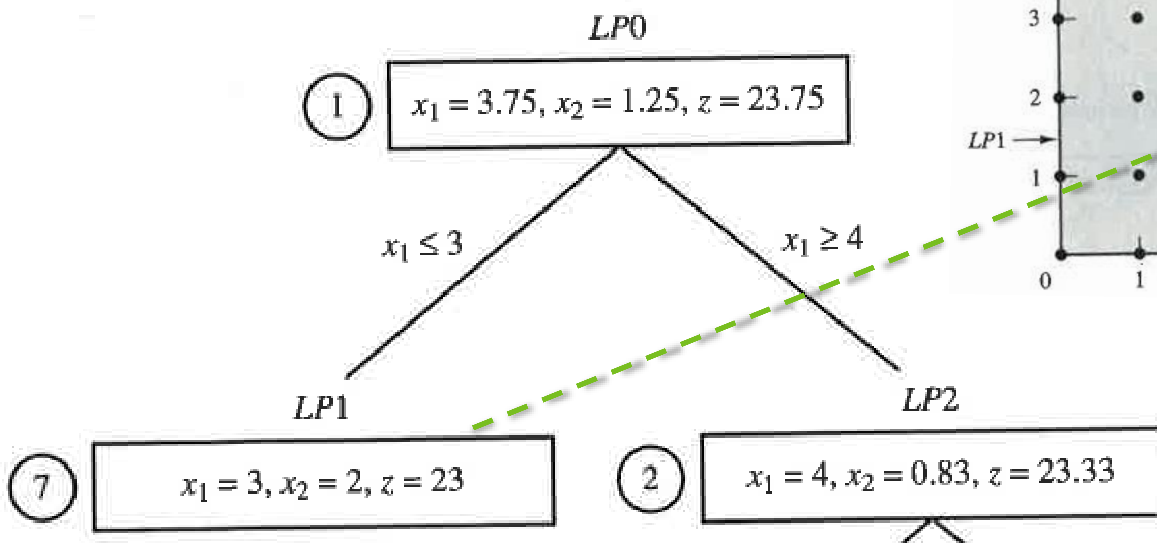
LP2

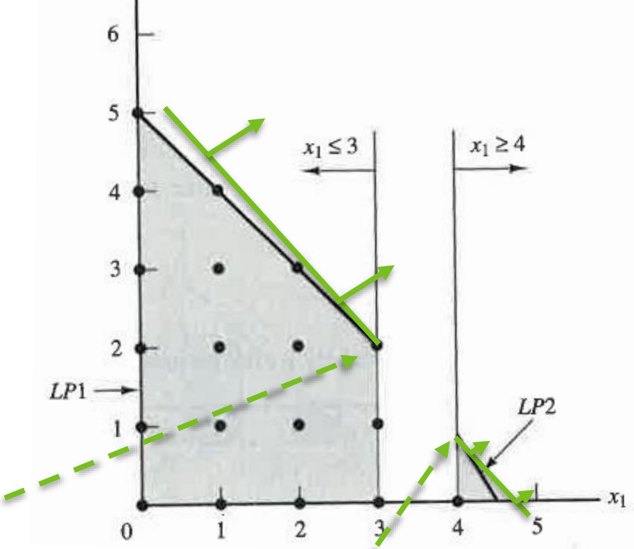
$$
\begin{aligned}
\text{Max} \quad & z = 5x_1 + 4x_2 \\
\text{s.t.} \quad & x_1 + x_2 \le 5 \\
& 10x_1 + 6x_2 \le 45 \\
& x_1 \ge 4, x_2 \ge 0 \\
& \text{and integer}
\end{aligned}
$$

Branch-and-bound (B&B) algorithm: Example

Step 3: Bounding

- Solve the LP relaxations in each branch

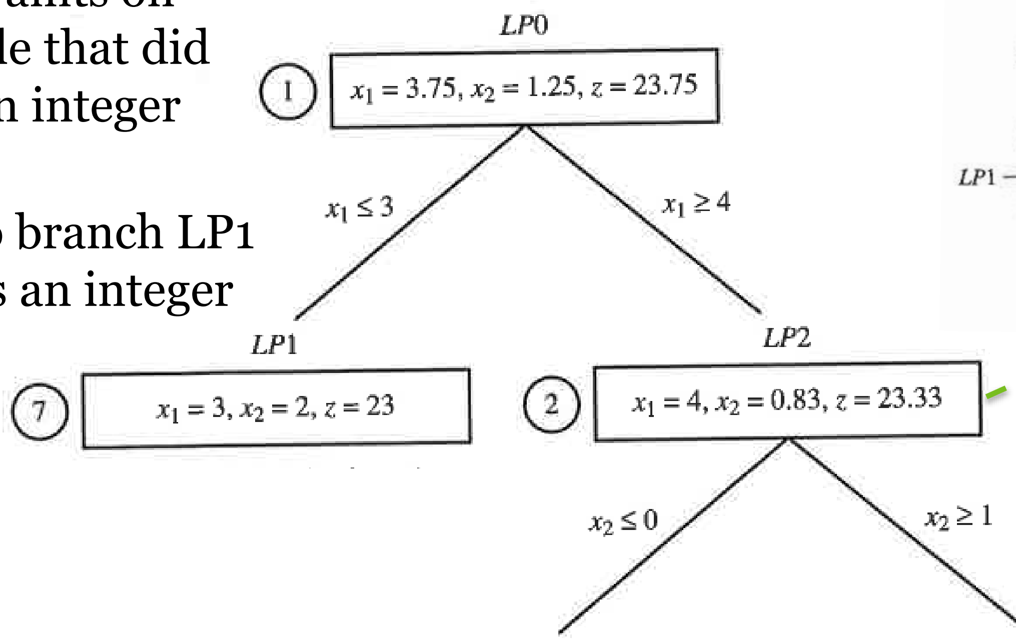


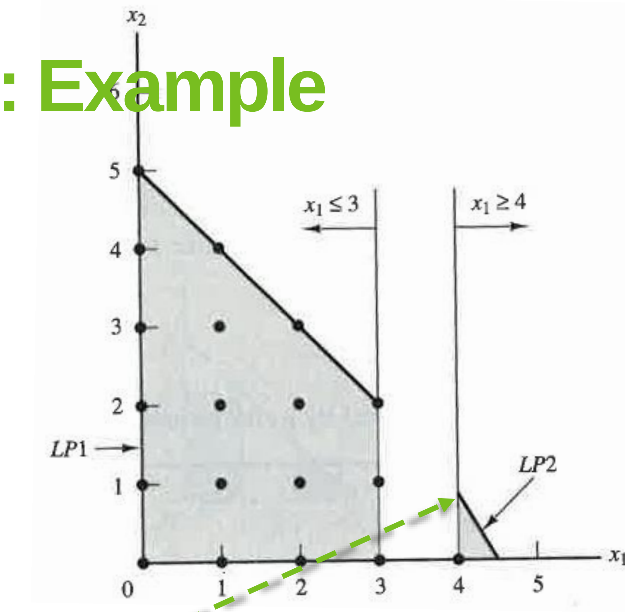


Branch-and-bound (B&B) algorithm: Example

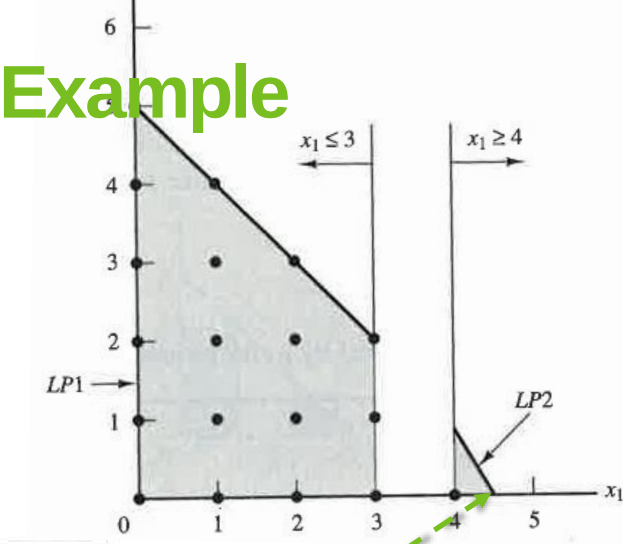
Step 4: Branching

- Add constraints on one variable that did not have an integer value
- No need to branch LP1 since it has an integer solution!



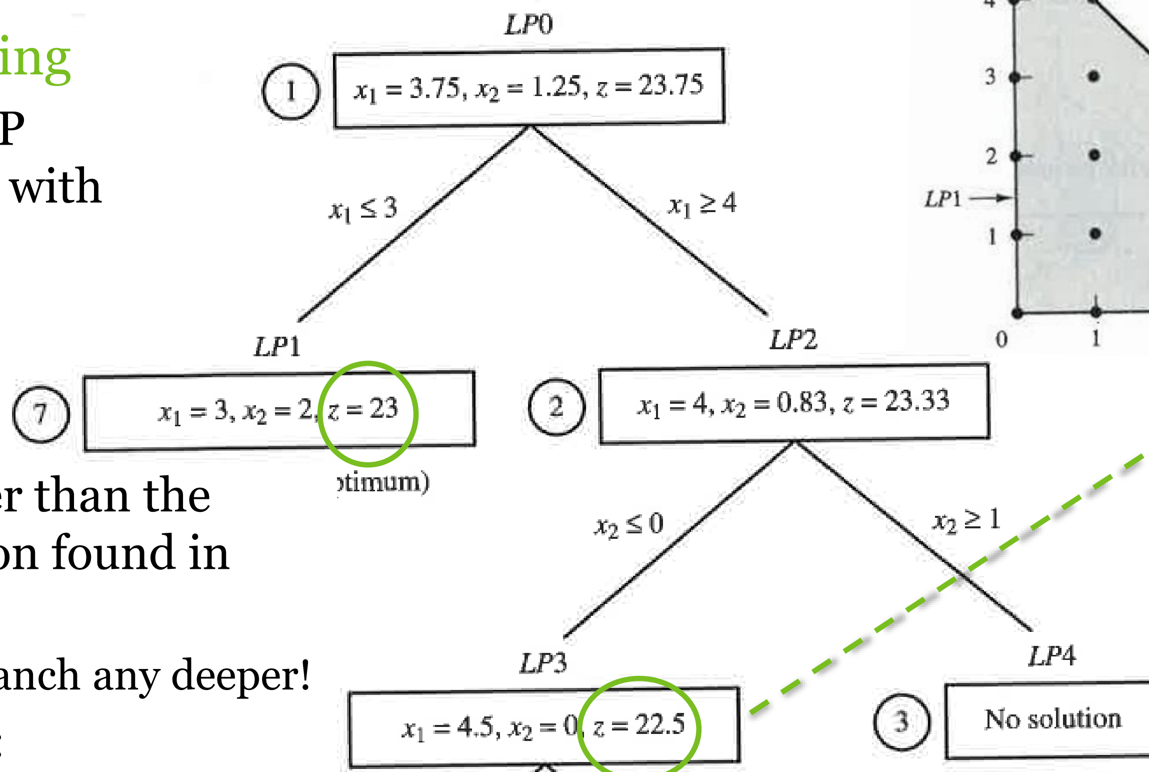


Branch-and-bound (B&B) algorithm: Example



Step 5: Bounding

- Solve the LP relaxations with Simple



(the problem is infeasible)

- Bound is lower than the integer solution found in Step 3:

  No need to branch any deeper!

- ILP optimum:

  $(x_1, x_2) = (3,2)$, $z=23$

Computational complexity: LP vs. MILP

- Solving (M)ILP problems is computationally much more demanding then solving LP problems
  - It is possible that in the B&B algorithm the number of sub-problems doubles with each branching-step
  - Hence, running time of the algorithm can grow exponentially as a function of the number of integer valued decision variables

- Adding constraints to an LP problem usually makes solving it computationally more demanding

- Adding constraints to a MILP problem can make it easier to solve!
  - Think about the B&B example problem with the additional constraint $x_1 \leq 3$

BLP Example: Capital Budgeting

- Perry Construction is faced with the problem of determining which projects it should undertake over the next three years:

$$\max \quad 180x_A + \ldots + 80x_E$$

s.t.
1. $30x_A + \ldots + 20x_E \leq 70$
2. $40x_A + 8x_B \ldots + 40x_E \leq 90$
3. $40x_A + 20x_C \ldots + 40x_E \leq 100$
4. $x_A, \ldots, x_E \in \{0,1\}$

Project	Estimated Present Value	Capital Requirements Year1	Year2	Year3
A	180 000	30 000	40 000	40 000
B	20 000	12 000	8 000	0
C	72 000	30 000	20 000	20 000
D	25 000	15 000	10 000	24 000
E	80 000	20 000	40 000	40 000
Funds Available		70 000	90 000	100 000

Question

- Interpret the meaning of the decision variables and the constraints

BLP Example: Capital Budgeting Revisited

- Question: Help the mgmt to formulate the additional restrictions:

1. At most three projects can be selected

$$x_A + x_B + x_C + x_D + x_E \leq 3$$

2. Projects A and D cannot be both selected

$$x_A + x_D \leq 1$$

3. If project C is selected, then project E must also be selected

$$x_C \leq x_E \quad (or\ x_C - x_E \leq 0)$$

4. If project B or E is selected, then project A cannot be selected

$$x_B + x_E + 2x_A \leq 2$$

5. If project E is selected, then projects C and D must also be selected

$$x_E \leq x_D$$

$$x_E \leq x_C$$

BLP: Special 0-1 Constraints (Cont'd)

- To make sure that a logical constraint works check that:
  1. solutions that are not allowed by the problem description are infeasible
  2. solutions allowed by the problem description are feasible
- Example: 1st year capital requirement is reduced by \$10,000 if at least 2 of projects C, D and E are selected (cf. a synergy)

  →New "dummy" variable x_s added to the model and 1. constraint modified:

$$30x_A + \ldots + 20x_E - 10x_S \le 70$$

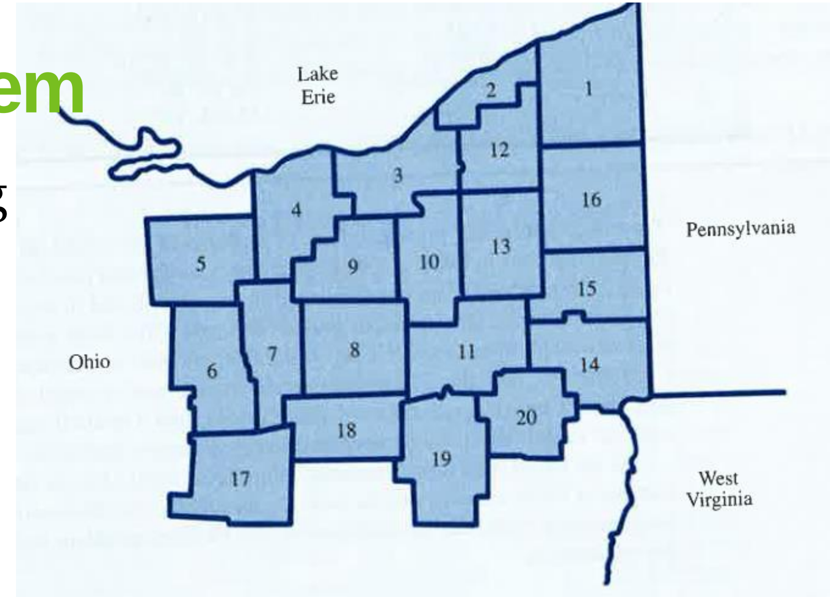
- New constraints ensure that "project S" is selected if and only if at least 2 of projects C, D and E are selected:

$$x_c + x_D + x_E - 2x_s \ge 0$$
$$x_c + x_D + x_E - 2x_s \le 1$$

Allowed?	$x_C + x_D + x_E$	x_S	$x_C + x_D + x_E - 2x_S$	Feasible?
yes	0	0	0	yes
no	0	1	-2	no
yes	1	0	1	yes
no	1	1	-1	no
no	2	0	2	no
yes	2	1	0	yes
no	3	0	3	no
yes	3	1	1	yes

BLP example: Covering problem

- Ohio Trust Company (OTC) is expanding into 20 new counties
  - Ohio Banking law: “A branch bank can be established in a county only if an adjacent county has a Principle Place of Business (PPB)”
  - Establishing a PPB requires state’s approval so OTC seeks to establish as few as possible new PPBs



$$\min x_1 + x_2 + x_3 + \cdots + x_{20}$$

(1) $x_1 + x_2 + x_{12} + x_{16} \geq 1$

(2) $x_1 + x_2 + x_3 + x_{12} \geq 1$

(3) $x_2 + x_3 + x_4 + x_9 + x_{10} + x_{12} + x_{13} \geq 1$

…

(20) $x_{11} + x_{14} + x_{19} + x_{20} \geq 1$

$$x_i \in \{\,0,1\}\,, i = 1, \ldots, 20 \geq 1$$

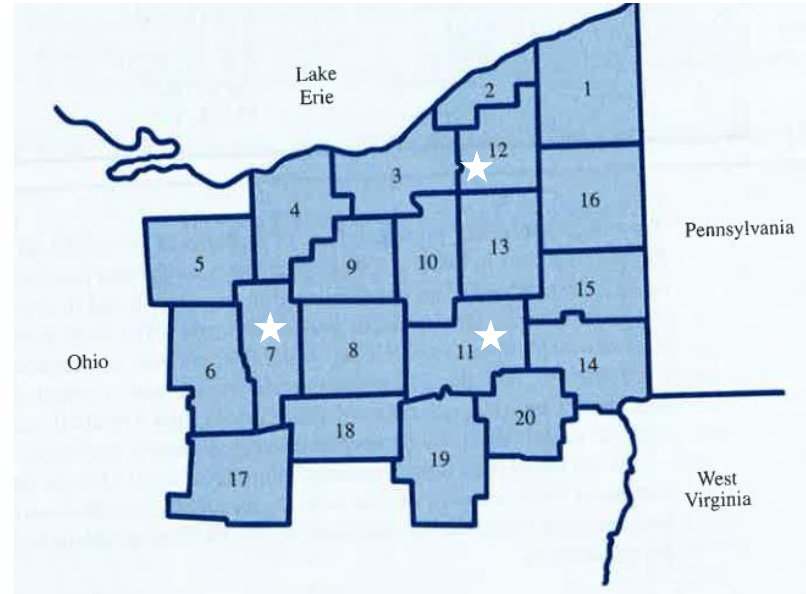
Question:

- Interpret the BLP problem
  - HINT: Decision variable $x_i = 1$ iff PPB is established in county i

BLP example: Covering problem (Cont'd)

- Optimal solution:

$$x_i = 1, \qquad i = 7, 11, 12$$
$$x_i = 0, \qquad i \neq 7, 11, 12$$



Question:

- Assume that the cost of establishing a PPB varies across counties
  - Denote cost in county *i* by c_i
- How would you modify the BLP model to minimize total costs?

$$\min \; c_1x_1 + c_2x_2 + c_3x_3 + \cdots + c_{20}x_{20}$$

(1) $x_1 + x_2 + x_{12} + x_{16} \geq 1$

(2) $x_1 + x_2 + x_3 + x_{12} \geq 1$

(3) $x_2 + x_3 + x_4 + x_9 + x_{10} + x_{12} + x_{13} \geq 1$

…

(20) $x_{11} + x_{14} + x_{19} + x_{20} \geq 1$

$x_i \in \{0,1\}, i = 1, \ldots, 20 \geq 1$

BLP Example: Marketing Plan

- The Supersuds Corporation is developing its next year's marketing plan for three new products.
  - Five TV spots purchased for commercials on national television networks.
  - Max 3 spots for each product
  - Each spot will feature a single product.
- How many spots should be allocated to each of the three products?

	Estimated Profits (Millions)		
Number of TV Spots	**Product 1**	**Product 2**	**Product 3**
0	\$1	\$2	\$0
1	2	3	4
2	3	5	5
3	4	6	6

BLP Example: Marketing Plan (Cont'd)

Decision variables: x_{ij} are binary (i = 1, 2, 3; j = 1, 2, 3,4).

Maximize Profit = $1x_{11}+2x_{12}+3x_{13}+4x_{14} +2x_{21}+3x_{22}+5x_{23}+6x_{24} +0x_{31}+4x_{32}+5x_{33}+6x_{34}$

Number of TV Spots	Estimated Profits (Millions) Product 1	Product 2	Product 3
0	$1	$2	$0
1	2	3	4
2	3	5	5
3	4	6	6

subject to:

Mutually Exclusive:

Product 1: $x_{11}+x_{12}+x_{13}+x_{14}= 1$

Product 2: $x_{21}+x_{22}+x_{23}+x_{24}= 1$

Product 3: $x_{31}+x_{32}+x_{33}+x_{34}= 1$

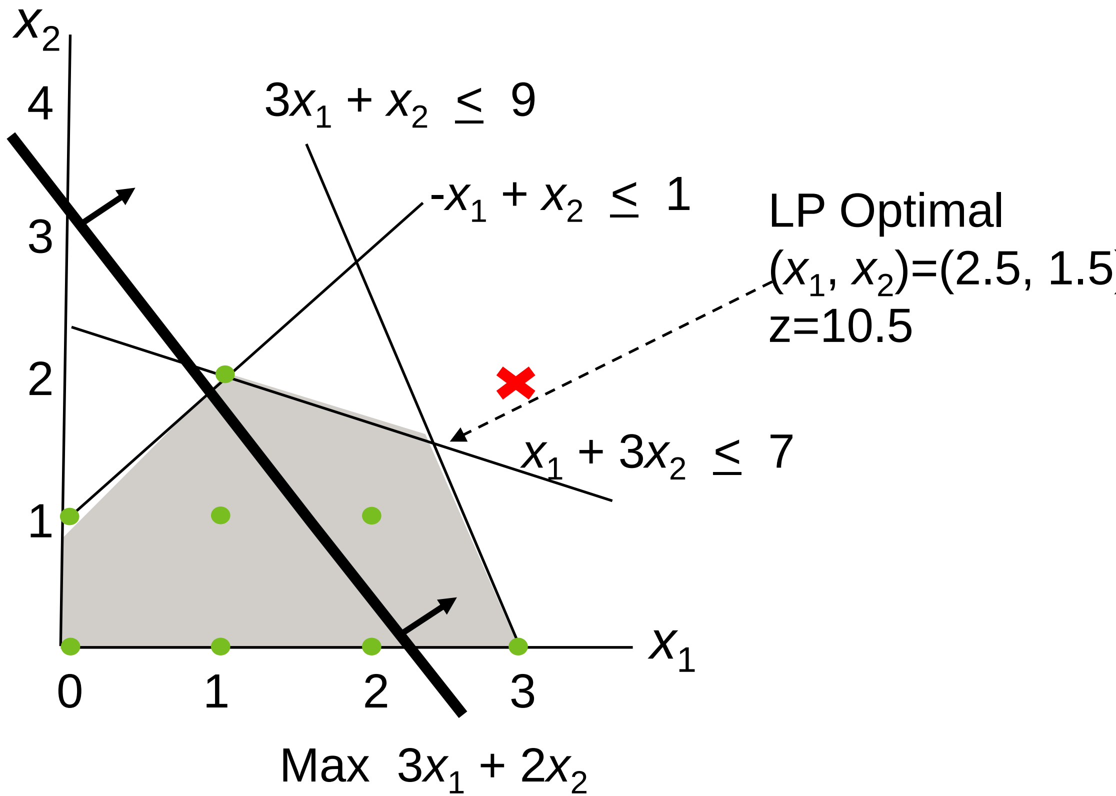
Total available spots: $1x_{12}+2x_{13}+3x_{14} +1x_{22}+2x_{23}+3x_{24} +1x_{32}+2x_{33}+3x_{34}\leq 5$

Other Examples of BLP Applications

- Investment Analysis
  - Should we make a certain fixed investment?
- Site Selection
  - Should a certain site be selected for the location of a new facility?
- Designing a Production and Distribution Network
  - Should a certain plant (distribution center) remain open?
  - Should a certain site be selected for a new plant (or distribution center)?
  - Should a distribution center remain open?
  - Should a certain distribution center be assigned to serve a certain market area?
- Scheduling Interrelated Activities
  - Should a certain activity begin in a certain time period?
- Airline Applications:
  - Should a certain type of airplane be assigned to a certain flight leg?
  - Should a certain sequence of flight legs be assigned to a crew?

Cautionary note about solving (M)ILP problems by "rounding"

- Trying to solve the problem by first solving the LP relaxation and then rounding-up gives an infeasible solution:
  - RoundUP(2.5, 1.5)=(3,2)✖

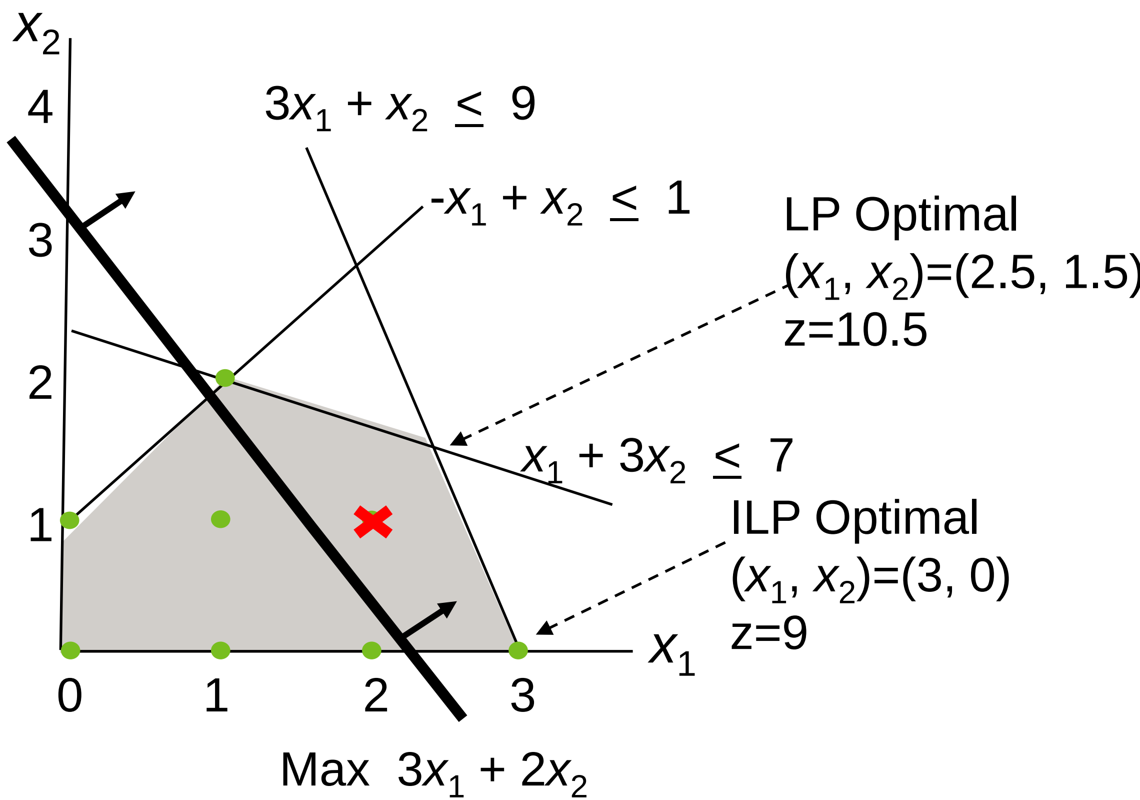


Cautionary note about solving (M)ILP problems by "rounding"

- Trying to solve the problem by first solving the LP relaxation and then rounding-down gives an sub-optimal solution:

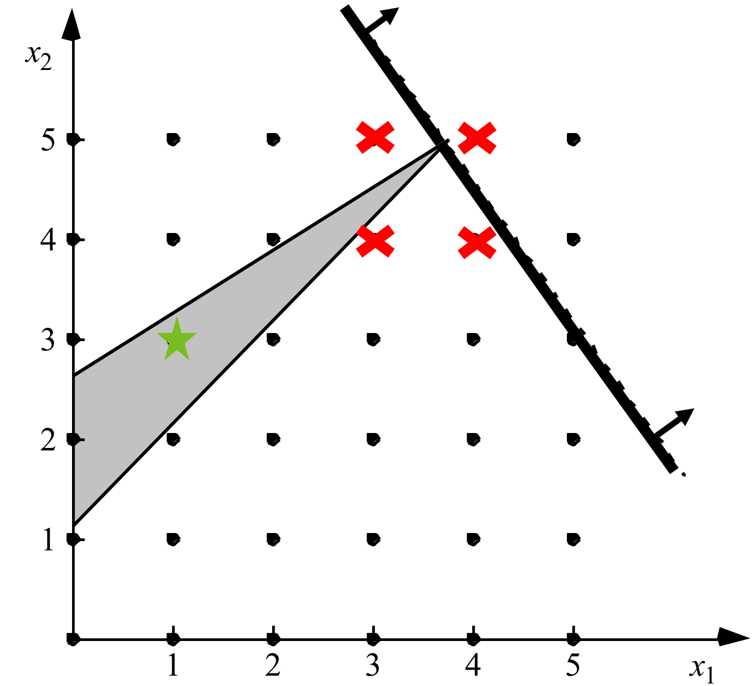
  RoundDW(2.5, 1.5)=(2,1) ✖

  z'=3(2) + 2(1)=8 < 9



Cautionary note about solving (M)ILP problems by "rounding"

- Rounded solution may not be feasible.
- Rounded solution may not be close to optimal.
- There can be *many* rounded solutions.
  - Example: Consider a problem with 30 variables that have non-integer values in the LP-solution. How many possible rounded solutions are there?



Solutions obtained by rounding optimal solutions of the LP relaxation (are infeasible)

ILP optimal solution

Cautionary note about solving (M)ILP problems by "rounding"

When are "non-integer" solutions okay?

- Solution is naturally divisible (e.g., $, pounds, hours)
- Solution represents a rate (e.g., units per week)

When is rounding okay?

- When numbers are large
  - e.g., rounding 114.286 to 114 is *probably* okay.

When is rounding not okay?

- When numbers are small
  - e.g., rounding 2.6 to 2 or 3 may be a problem.
- Binary variables
  - yes-or-no decisions

Cautionary note about sensitivity analysis in (M)ILP problems

- A B&B algorithm usually does not provide information on the solution sensitivity (cf. Simplex algorithm)

  →No “Sensitivity report” for (M)ILP problems
- Yet, analyzing sensitivity of (M)ILP problems is important
  - Maybe more important than for LP problems
- Sensitivity analysis requires re-optimizing the problem

$$\max z = 40x_1 + 60x_2 + 70x_3 + 160x_4$$
$$16x_1 + 35x_2 + 45x_3 + 85x_4 \leq \mathbf{100}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

→Optimum $x = (1,1,1,0), z = 170$

$$\max z = 40x_1 + 60x_2 + 70x_3 + 160x_4$$
$$16x_1 + 35x_2 + 45x_3 + 85x_4 \leq \mathbf{101}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

→Optimum $x = (1,0,0,1), z = 200$

MILP Example: Fixed-Charge Problem

- A product can be assembled on any of the five assembly lines.
  - For each line, the table below gives the cost of assembling a product, the assembly time required per product, the start-up cost, and the maximum number of hours the line can be operated during the next month.
- At least 350 units of product must be assembled next month.

Line	Start-up Cost	Prod. Cost/unit	Prod.Time (hrs./unit)	Maximum Prod.hrs.
A	\$ 6,000	\$ 80	5	510
B	\$10,000	\$ 60	6	480
C	\$ 2,000	\$110	10	600
D	\$ 7,500	\$ 75	4	440
E	\$15,000	\$ 40	3	480

MILP Example: Fixed-Charge Problem (cont'd)

$$\min \quad 80x_A + \ldots + 40x_E + 6000y_A + \ldots + 15000y_E$$

s.t.

1. $x_A + x_B + \ldots + x_E \geq 350$
2. $5x_A \leq 510\, y_A$
3. $6x_B \leq 480\, y_B$
4. $10x_C \leq 600\, y_C$
5. $4x_D \leq 440\, y_D$
6. $3x_E \leq 480\, y_E$

$y_A, \ldots, y_E \in \{0,1\}$

$x_A, \ldots, x_E \geq 0$

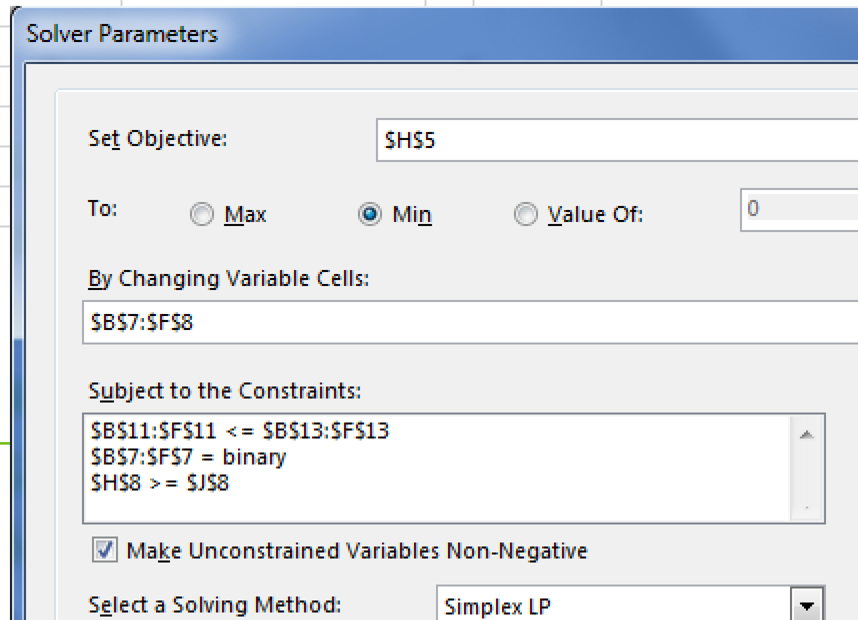
Line	Start-up Cost	Prod. Cost/unit	Prod.Time (hrs./unit)	Maximum Prod.hrs.
A	\$ 6,000	\$ 80	5	510
B	\$10,000	\$ 60	6	480
C	\$ 2,000	\$110	10	600
D	\$ 7,500	\$ 75	4	440
E	\$15,000	\$ 40	3	480

Question

- Interpret the decision variables, the objective function and the constraints

MILP Example: Fixed-Charge Problem (cont'd)

	A	B	C	D	E	F	G	H	I	J
1										
2		Production Lines						MINIMIZE		
3		A	B	C	D	E		TOTAL		
4	Start-up cost	6000	10000	2000	7500	15000		COST		
5	Cost/unit	80	60	110	75	40		49150		
6										
7	Start? (y)	1	0	0	1	1		Units produced total		minimum
8	Units produced (x)	70	1.42E-14	0	110	170		350	>=	350
9										
10	Prod time in hours / unit	5	6	10	4	3				
11	Hours total	350	8.53E-14	0	440	510				
12		<=	<=	<=	<=	<=				
13	Hours available	510	0	0	440	510				
14	Hours max	510	480	600	440	510				
15										



MILP Example: Fixed-Charge Problem (cont'd)

Question: Help the management to formulate the additional restrictions:

1. If line E is operated, then line B must also be operated

$$y_E \leq y_B$$

2. If line A is operated, then lines D and E may not be operated

$$2y_A + y_D + y_E \leq 2$$

3. If line B is operated, then at least 50 units must be produced on that line

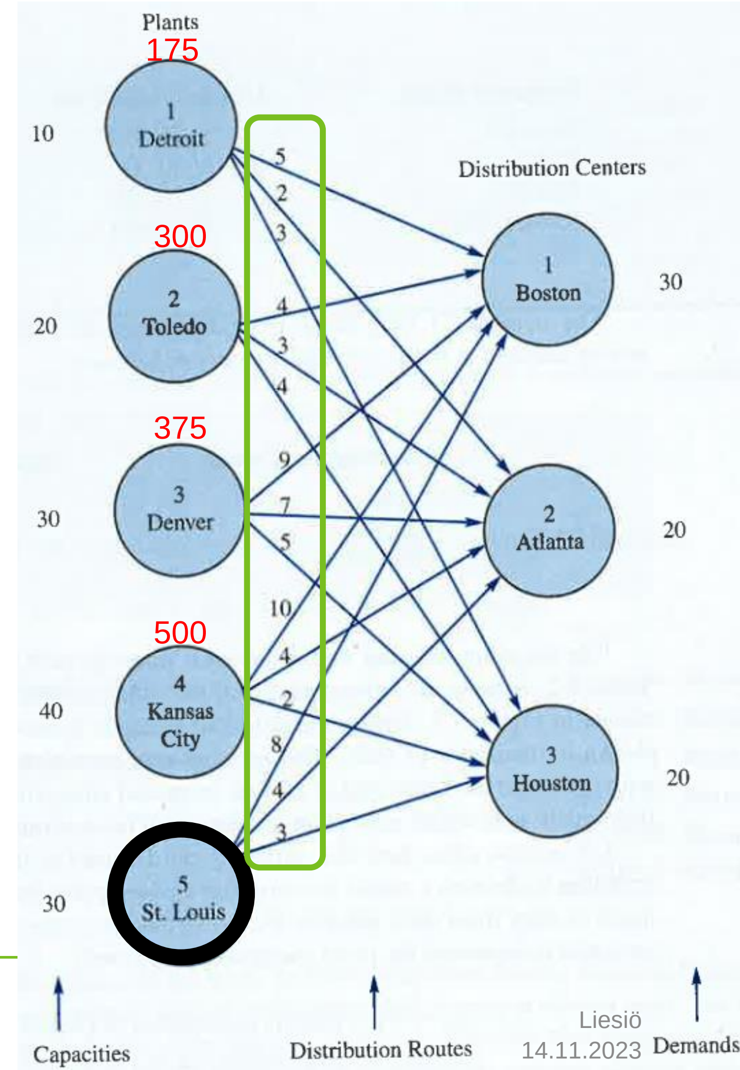
$$50y_B \leq x_B$$

4. If line C is operated, then no more than 150 units may be produced on lines B and D combined

$$x_B + x_D \leq 1000 - 850y_C$$

MILP Example: Distribution System Design

- Company operates a plant in St. Louis
  - Annual capacity of 30,000 units
- Products shipped to 3 distribution centers in Boston, Atlanta and Houston
  - Different demands
- Possible locations for new plants: Detroit, Toledo, Denver, Kansas City
  - Differ in terms of annual fixed operating costs and cost of shipment to distribution centers
- In which cities should new plants be built?
  - Satisfy demands with minimal cost



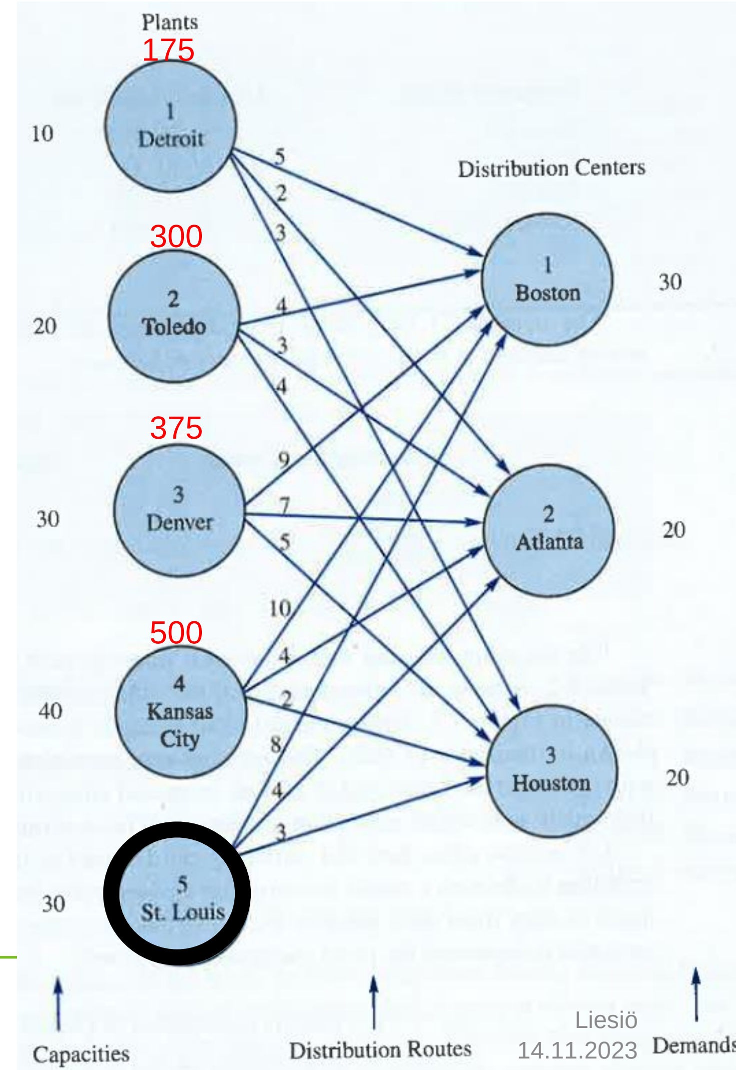
Aalto University
School of Business

MILP Example: Distribution System Design (Cont'd)

$$\min 5x_{11} + 2x_{12} + 3x_{13} + \cdots + 4x_{52} + 3x_{53}$$
$$+175y_1 + 300y_2 + 375y_3 + 500y_4$$
$$x_{11} + x_{12} + x_{13} \leq 10y_1$$
$$x_{21} + x_{22} + x_{23} \leq 20y_2$$
$$x_{31} + x_{32} + x_{33} \leq 30y_3$$
$$x_{41} + x_{42} + x_{43} \leq 40y_4$$
$$x_{51} + x_{52} + x_{53} \leq 30$$
$$x_{11} + x_{21} + x_{31} + x_{41} + x_{51} = 30$$
$$x_{12} + x_{22} + x_{32} + x_{42} + x_{52} = 20$$
$$x_{13} + x_{23} + x_{33} + x_{43} + x_{53} = 20$$
$$x_{ij} \geq 0,\ y_k \in \{0,1\},$$

Question:

- Interpret the model, i.e., decision variables, objective function, constraints

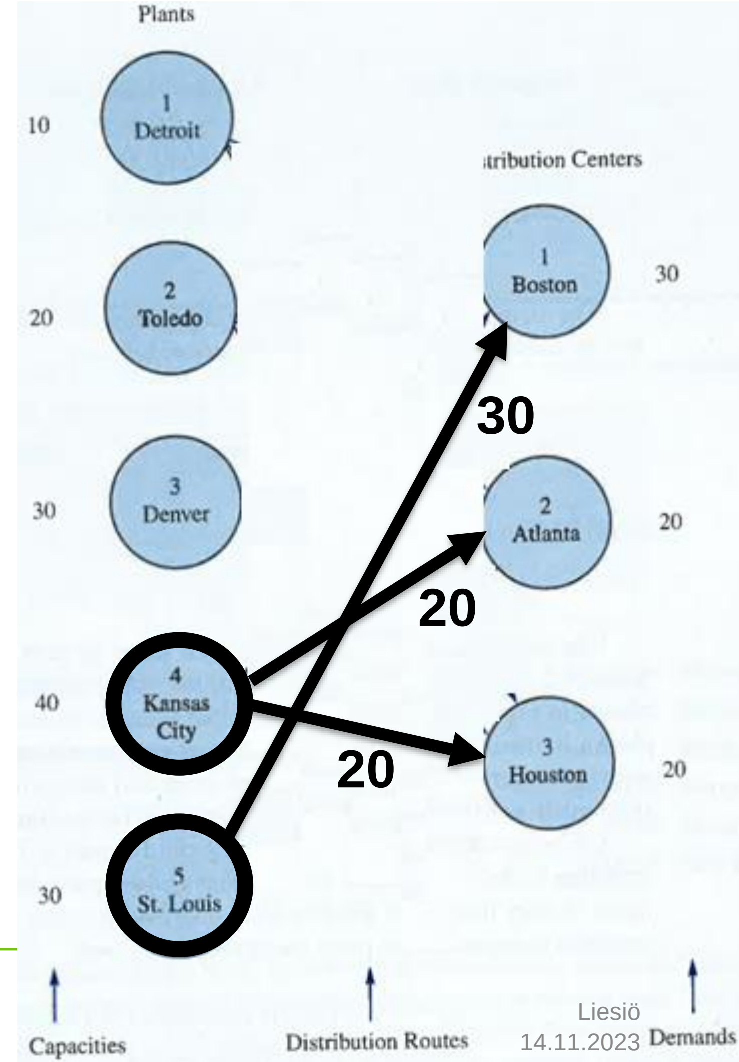


MILP Example: Distribution System Design (Cont'd)

$$\min 5x_{11} + 2x_{12} + 3x_{13} + \cdots + 4x_{52} + 3x_{53}$$
$$+175y_1 + 300y_2 + 375y_3 + 500y_4$$
$$x_{11} + x_{12} + x_{13} \leq 10y_1$$
$$x_{21} + x_{22} + x_{23} \leq 20y_2$$
$$x_{31} + x_{32} + x_{33} \leq 30y_3$$
$$x_{41} + x_{42} + x_{43} \leq 40y_4$$
$$x_{51} + x_{52} + x_{53} \leq 30$$
$$x_{11} + x_{21} + x_{31} + x_{41} + x_{51} = 30$$
$$x_{12} + x_{22} + x_{32} + x_{42} + x_{52} = 20$$
$$x_{13} + x_{23} + x_{33} + x_{43} + x_{53} = 20$$
$$x_{ij} \geq 0, y_k \in \{0,1\},$$

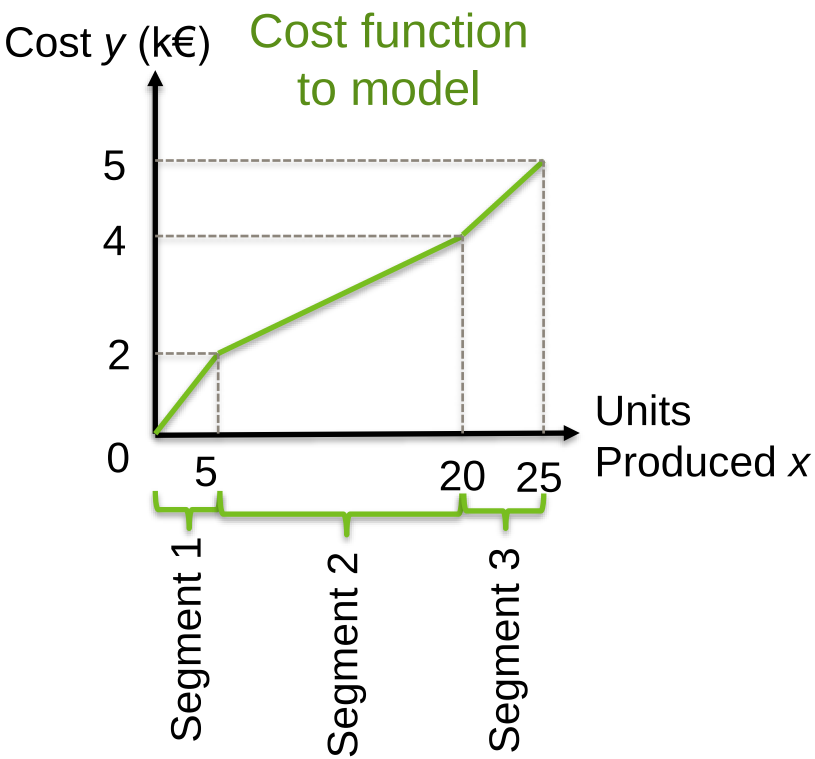
- Optimal solution (non-zero dec. var.):

  $$x_{42} = 20, x_{43} = 20, x_{51} = 30, y_4 = 1$$



Modelling non-constant marginal costs with MILP

- MILP can capture arbitrary piecewise linear functions in the objective function or in the constraints



MILP formulation

- Decision variables:
  - y: cost
  - x: number of units produced
  - $z_1, z_2, z_3 \in \{0,1\}$:
    x located on segment with z_i=1
  - $c_1, c_2, c_3, c_4 \in [0,1]$:
    'Weights for segment borders'
- Constraints:

$$0c_1 + 5c_2 + 20c_3 + 25c_4 = x$$
$$c_1 + c_2 + c_3 + c_4 = 1$$
$$z_1 + z_2 + z_3 = 1$$
$$c_1 \leq z_1$$
$$c_2 \leq z_1 + z_2$$
$$c_3 \leq z_2 + z_3$$
$$c_4 \leq z_3$$
$$y = 0c_1 + 2c_2 + 4c_3 + 5c_4$$

Modelling variable marginal costs with MILP (cont'd)

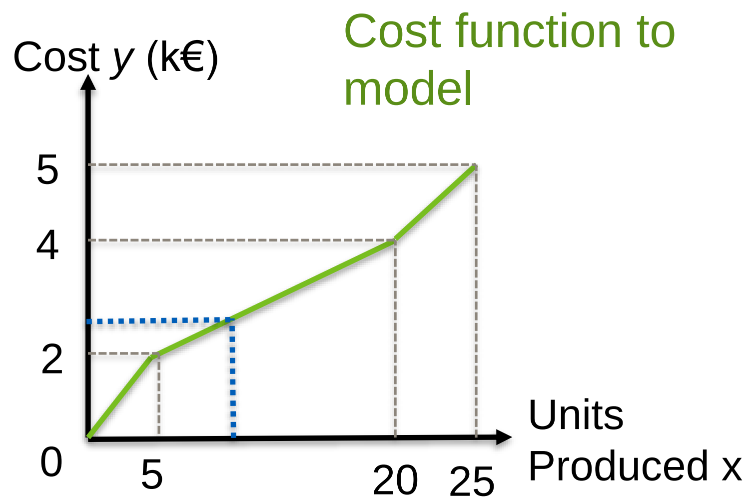
- Consider x=10 units are produced
- Then MILP has to find $c_i \in [0,1]$ such that

$$0c_1 + 5c_2 + 20c_3 + 25c_4 = 10$$

- However, only two consecutive c_i:s can have a non-zero value due to z_i:s:
  - z_1=1 and $0c_1 + 5c_2 = 10$ ➔ infeasible
  - z_2=1 and $5c_2 + 20c_3 = 10$ ➔ c_2=2/3, c_3=1/3
  - z_3=1 and $20c_3 + 25c_4 = 10$ ➔ infeasible
- Hence, the cost is equal to

$$y = 0(0) + 2\left(\frac{2}{3}\right) + 4\left(\frac{1}{3}\right) + 5(0) = \frac{8}{3} = 2\,\frac{2}{3},$$

which is inline with the original cost function!



$$0c_1 + 5c_2 + 20c_3 + 25c_4 = x$$
$$c_1 + c_2 + c_3 + c_4 = 1$$
$$z_1 + z_2 + z_3 = 1$$
$$c_1 \leq z_1$$
$$c_2 \leq z_1 + z_2$$
$$c_3 \leq z_2 + z_3$$
$$c_4 \leq z_3$$
$$y = 0c_1 + 2c_2 + 4c_3 + 5c_4$$

Integer Linear Programming - Summary

- Different types: MILP, pure ILP, BLP,….

- Compared to LP models, MILP models can capture
  - Yes/no decisions
  - Logical dependencies among decisions
  - Fixed-charges (e.g. start-up costs)
  - Piecewise linear functions (e.g. non-constant marginal production costs)

- Solving of ILPs is fundamentally different from solving LPs
  - Solution time may increase exponentially as a function of the problem size

Aalto University
School of Business

Optimization software

- *Software for optimization*
- *Example: Python + PuLP*
- *Extra example: Python + Gurobi*

Optimization Software beyond Excel Solver

- Optimization solvers (especially MILP)
  - Commercial: Gurobi, IBM CPLEX Optimizer, FICO Xpress, MOSEK, …
  - Open source: lp_solve, GLPK, Open Solver (http://opensolver.org/), PuLP

- Optimization models are usually build with some “programming language” which then calls the solver
  - E.g. R, Matlab, C++, Java, AMPL, Python
  - Excel interfaces exists for most solvers (At least through Visual Basic)

- Next a demo: Model is written in Python and then solved with PuLP
  - Extra slides: Python+Gurobi implementation of the same model
    - Free academic license for Gurobi available here: https://www.gurobi.com/downloads/free-academic-license/

P&P transportation problem revisited: LP formulation

$$\begin{aligned}\min\ & 53x_{11} + 95x_{12} + 136x_{13} + 160x_{14} \\ & + 60x_{21} + 120x_{22} + 132x_{23} + 140x_{24} \\ & + 210x_{31} + 190x_{32} + 89x_{33} + 71x_{34}\end{aligned}$$

Minimize total transportation costs

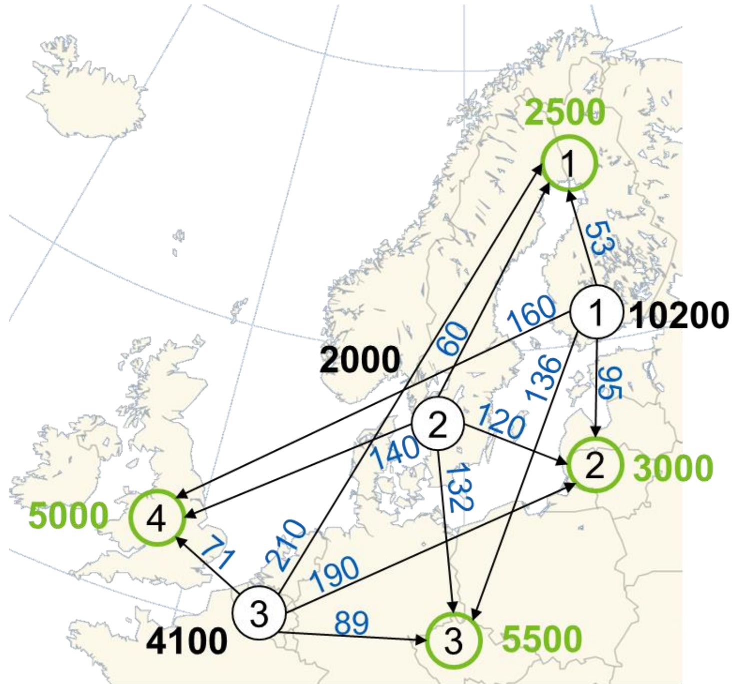
$$\begin{aligned} x_{11} + x_{21} + x_{31} &= 2500 \\ x_{12} + x_{22} + x_{32} &= 3000 \\ x_{13} + x_{23} + x_{33} &= 5500 \\ x_{14} + x_{24} + x_{34} &= 5000 \end{aligned}$$

Satisfy demand

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &\le 10200 \\ x_{21} + x_{22} + x_{23} + x_{24} &\le 2000 \\ x_{31} + x_{32} + x_{33} + x_{34} &\le 4100 \end{aligned}$$

Do not exceed production capacities

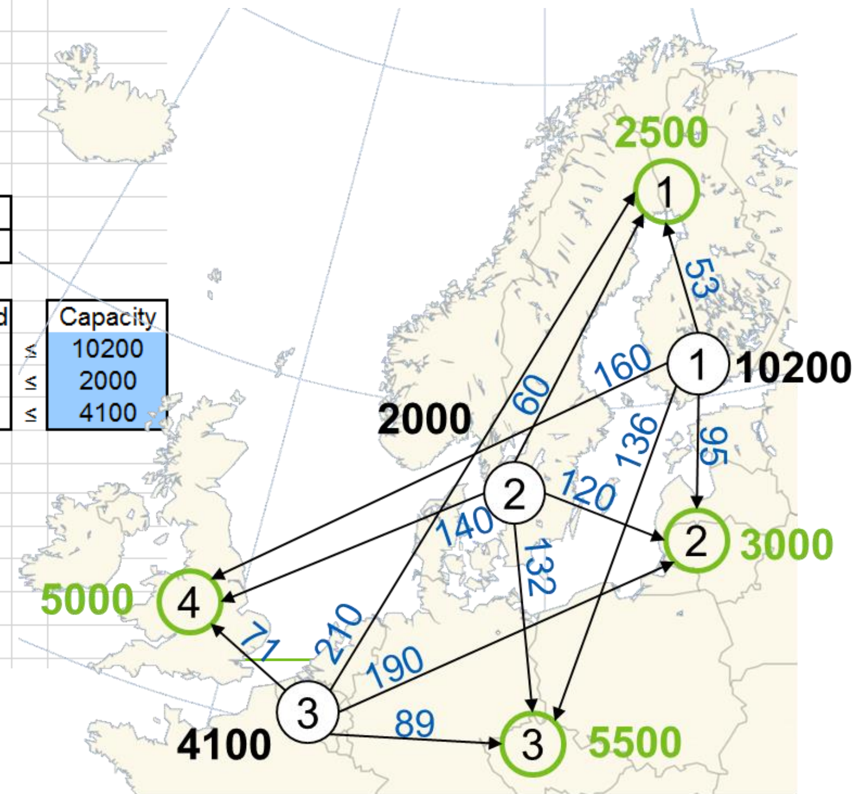
$$x_{ij} \ge 0, i = 1, \ldots, 3, j = 1, \ldots 4$$



P&P transportation problem revisited: Spreadsheet implementation

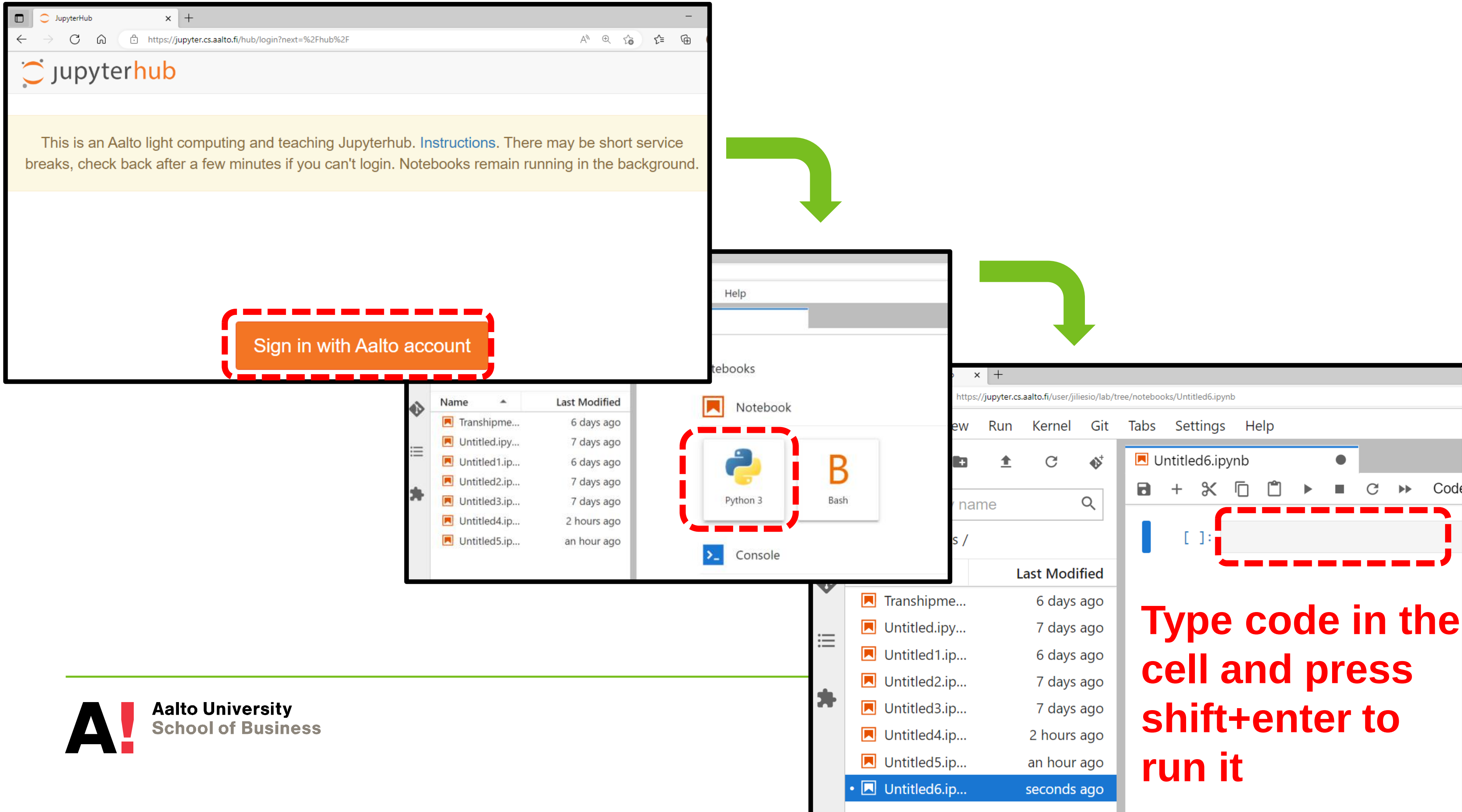
G9 =SUMPRODUCT($C$5:$F$7;$C$12:$F$14)

	A	B	C	D	E	F	G	H	I
1	**Pulp&Paper Company**								
2									
3	**Transportation**		Warehouse						
4	**cost per ton**		1. Finland	2. Lithuania	3. Czech	4. UK			
5	Mill	1. Finland	53 €	95 €	136 €	160 €			
6		2. Sweden	60 €	120 €	132 €	140 €			
7		3. Belgium	210 €	190 €	89 €	71 €			
8							Total Cost		
9							1 578 200 €		
10	**Transported**		Warehouse						
11	**quantity x_{ij} (tons)**		1. Finland	2. Lithuania	3. Czech	4. UK	Total Shipped		Capacity
12	Mill	1. Finland	2500	3000	4400	0	9900	≤	10200
13		2. Sweden	0	0	1100	900	2000	≤	2000
14		3. Belgium	0	0	0	4100	4100	≤	4100
15		Total Received	2500	3000	5500	5000			
16			=	=	=	=			
17		Demand	2500	3000	5500	5000			
18									
19									
20	Parameters								
21	Decision variables								



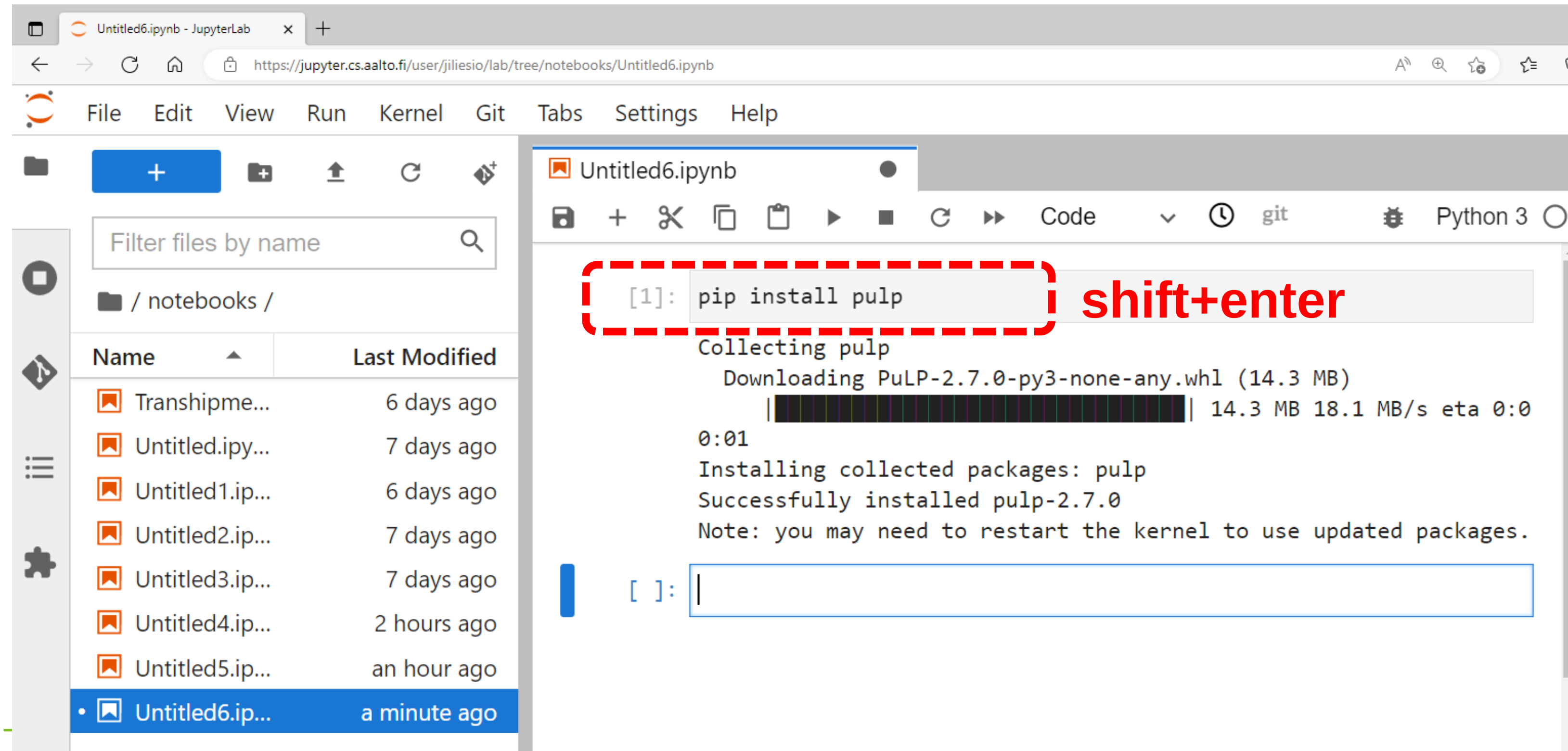
Aalto University
School of Business

Login to jupyter.cs.aalto.fi



Type code in the cell and press shift+enter to run it

Install the PuLP solver for MILP problems



P&P transportation problem revisited: Indices and parameters

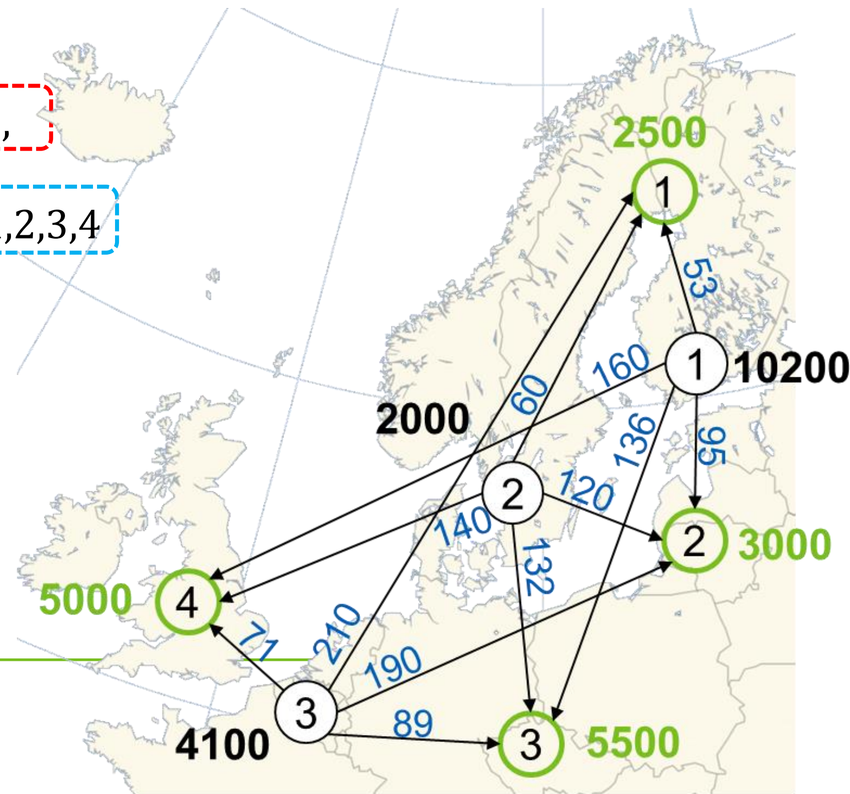
PPcompany.ipynb

Code git

```
from pulp import *

#Indices
mills = list(range(3))
warehouses =  list(range(4))

#Parameters
costs=[[53,95,136,160],
[60,120,132,140],
[210, 190, 89, 71]]
capacities=[10200, 2000, 4100]
demands = [2500, 3000, 5500, 5000]

#Create a optimization model (minimization)
model = LpProblem("PP_model", LpMinimize)
```

mill $i = 1,2,3,$

warehouse $j = 1,2,3,4$



P&P transportation problem revisited: Decision variables

Decision variables x_{ij}:

Tons of carton transported form mill i

to warehouse j:

$$x_{ij} \geq 0,$$

$$\boxed{i = 1,2,3,}$$

$$\boxed{j = 1,2,3,4}$$

PPcompany.ipynb

```
#Create decision variables x_ij
x=[[LpVariable("x_"+str(i+1)+str(j+1),0,None) for j in warehouses] for i in mills]
```

Upper bound

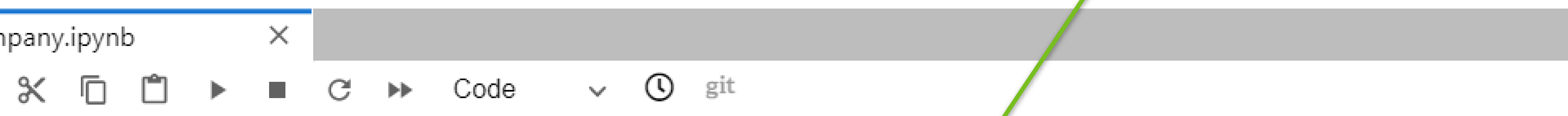
Lower bound

Arbitrary name for the decision variable; here we use x_11,…,x_34

str( ) method returns a string presentation of an integer

P&P transportation problem revisited: Capacity constraints

$$\left.\begin{aligned} x_{11}+x_{12}+x_{13}+x_{14} &\leq 10200 \\ x_{21}+x_{22}+x_{23}+x_{24} &\leq 2000 \\ x_{31}+x_{32}+x_{33}+x_{34} &\leq 4100 \end{aligned}\right\} \Leftrightarrow \boxed{\sum_{j=1}^{4} x_{ij}} \leq s_i, \boxed{i = 1,2,3}$$




```
#Add capacity constraints at the mills:
for i in mills:
    model += (lpSum([x[i][j] for j in warehouses])<=capacities[i], "Capacity_at_mill_"+str(i+1))
```

Constraints are added to the model object

Name for the constraint

P&P transportation problem revisited: Demand constraints

$$\left.\begin{aligned} x_{11} + x_{21} + x_{31} &= 2500 \\ x_{12} + x_{22} + x_{32} &= 3000 \\ x_{13} + x_{23} + x_{33} &= 5500 \\ x_{14} + x_{24} + x_{34} &= 5000 \end{aligned}\right\} \Leftrightarrow \boxed{\sum_{i=1}^{3} x_{ij}} = d_j, \boxed{j = 1,2,3,4}$$

ompany.ipynb

Code git

```
#Add balance constraints at the warehouses:
for j in warehouses:
    model += (lpSum([x[i][j] for i in mills])==demands[j], "Demand_at_warehouse_"+str(j+1))
```

P&P transportation problem revisited: Objective function

$$\min 53x_{11} + 95x_{12} + 136x_{13} + 160x_{14} + 60x_{21} + 120x_{22} + 132x_{23} + 140x_{24} + 210x_{31} + 190x_{32} + 89x_{33} + 71x_{34}$$

$$\Leftrightarrow \min \boxed{\sum_{i=1}^{3} \sum_{j=1}^{4} c_{ij} x_{ij}}$$

PPcompany.ipynb

Code git

```
#Add objective function:
model += (lpSum([costs[i][j]*x[i][j] for i in mills for j in warehouses]) , "Transportation_costs")
```

P&P transportation problem revisited: Solve the model and print the optimal solution

company.ipynb

Code Validate

```python
#Print optimal objective function value:
print("Optimal transportation cost "+str(model.objective.value())+" .")


#Print optimal decision variable values:
for i in mills:
    print("------")
    for j in warehouses:
        if (x[i][j].varValue>0):
            print("Transport "+str(x[i][j].varValue)+" tons from mill "+str(i+1)+" to warehouse "+str(j+1)+".")
```

P&P transportation problem revisited: Run the program

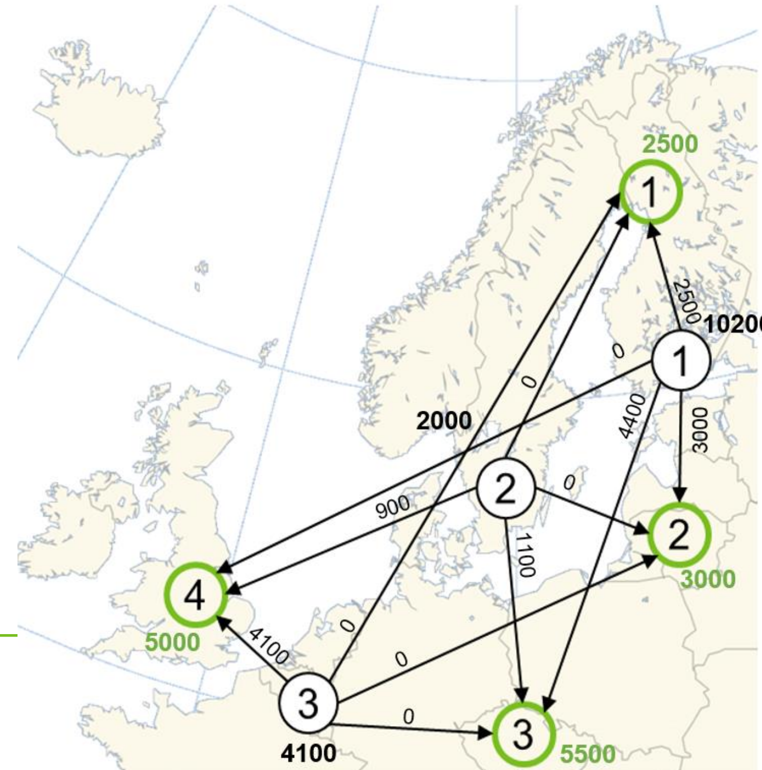
PPcompany.ipynb

```
from pulp import *

#Indices
mills = list(range(3))
warehouses =  list(range(4))

#Parameters
costs=[[53,95,136,160],
[60,120,132,140],
[210, 190, 89, 71]]
capacities=[10200, 2000, 4100]
demands = [2500, 3000, 5500, 5000]

#Create a optimization model (minimization)
model = LpProblem("PP_model", LpMinimize)

#Create decision variables x_ij
x=[[LpVariable("x_"+str(i+1)+str(j+1),0,None) for j in warehouses] for i in mills]

#Add objective function:
model += (lpSum([costs[i][j]*x[i][j] for i in mills for j in warehouses]) , "Transportation_costs")

#Add capacity constraints at the mills:
for i in mills:
    model += (lpSum([x[i][j] for j in warehouses])<=capacities[i], "Capacity_at_mill_"+str(i+1))

#Add balance constraints at the warehouses:
for j in warehouses:
    model += (lpSum([x[i][j] for i in mills])==demands[j], "Demand_at_warehouse_"+str(j+1))

model.solve() #Solve the optimal solution to model

#Print optimal objective function value:
print("Optimal transportation cost "+str(model.objective.value())+" .")


#Print optimal decision variable values:
for i in mills:
    print("------")
    for j in warehouses:
        if (x[i][j].varValue>0):
            print("Transport "+str(x[i][j].varValue)+" tons from mill "+str(i+1)+" to warehouse "+str(j+1)+".")
```

shift+enter

P&P transportation problem revisited: Program output

```
Welcome to the CBC MILP Solver
Version: 2.10.3
Build Date: Dec 15 2019

command line - /opt/software/lib/python3.10/site-packages/pulp/solverdir/cbc/linux/
64/cbc /tmp/6a293f80378e420883abbd5fa5fbc186-pulp.mps timeMode elapsed branch print
ingOptions all solution /tmp/6a293f80378e420883abbd5fa5fbc186-pulp.sol (default str
ategy 1)
At line 2 NAME          MODEL
At line 3 ROWS
At line 12 COLUMNS
At line 49 RHS
At line 57 BOUNDS
At line 58 ENDATA
Problem MODEL has 7 rows, 12 columns and 24 elements
Coin0008I MODEL read with 0 errors
Option for timeMode changed from cpu to elapsed
Presolve 7 (0) rows, 12 (0) columns and 24 (0) elements
0  Obj 0 Primal inf 16000 (4)
7  Obj 1578200
Optimal - objective value 1578200
Optimal objective 1578200 - 7 iterations time 0.002
Option for printingOptions changed from normal to all
Total time (CPU seconds):       0.00   (Wallclock seconds):       0.00

Optimal transportation cost 1578200.0 .
------
Transport 2500.0 tons from mill 1 to warehouse 1.
Transport 3000.0 tons from mill 1 to warehouse 2.
Transport 4400.0 tons from mill 1 to warehouse 3.
------
Transport 1100.0 tons from mill 2 to warehouse 3.
Transport 900.0 tons from mill 2 to warehouse 4.
------
Transport 4100.0 tons from mill 3 to warehouse 4.
```



P&P transportation problem revisited: Inspecting the model object

PPcompany.ipynb

Code   git

[6]: model   shift+enter

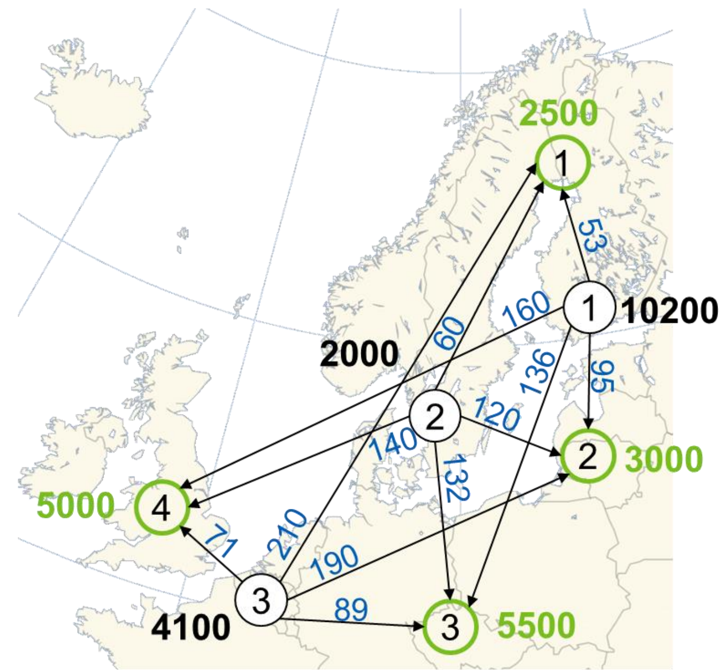
```
[6]: PP_model:
MINIMIZE
53*x_11 + 95*x_12 + 136*x_13 + 160*x_14 + 60*x_21 + 120*x_22 + 132*x_23 + 140*x_24 + 210*x_31 + 190*x_32 + 89*x_33 + 71*x_34 + 0
SUBJECT TO
Capacity_at_mill_1: x_11 + x_12 + x_13 + x_14 <= 10200

Capacity_at_mill_2: x_21 + x_22 + x_23 + x_24 <= 2000

Capacity_at_mill_3: x_31 + x_32 + x_33 + x_34 <= 4100

Demand_at_warehouse_1: x_11 + x_21 + x_31 = 2500

Demand_at_warehouse_2: x_12 + x_22 + x_32 = 3000

Demand_at_warehouse_3: x_13 + x_23 + x_33 = 5500

Demand_at_warehouse_4: x_14 + x_24 + x_34 = 5000

VARIABLES
x_11 Continuous
x_12 Continuous
x_13 Continuous
x_14 Continuous
x_21 Continuous
x_22 Continuous
x_23 Continuous
x_24 Continuous
x_31 Continuous
x_32 Continuous
x_33 Continuous
x_34 Continuous
```

Extra example: P&P transportation problem in Python + Gurobi

Aalto University
School of Business

P&P transportation problem revisited: Python + Gurobi -implementation

```
#Import Gurobi solver library
from gurobipy import *

#Data
costs=[[53,95,136,160],
[60,120,132,140],
[210, 190, 89, 71]]

capacities=[10200, 2000, 4100]
demands = [2500, 3000, 5500, 5000]
```



P&P transportation problem revisited: Python + Gurobi -implementation

```
#Create the LP model in gurobi
model = Model("PP company transportation")

#Indexes
mills = range(3)
warehouses = range(4)

#Decision variables x_ij
x=[]
for i in mills:
    x.append([])
    for j in warehouses:
        x[i].append(model.addVar(lb=0, name="x%d.%d" % (i, j)))
model.update()
```

Decision variables x_{ij}:
Tons of carton transported form mill i to warehouse j:

$$x_{ij} \geq 0,$$
$$i = 1,2,3,$$
$$j = 1,2,3,4$$

P&P transportation problem revisited: Python + Gurobi -implementation

$$\left.\begin{aligned} x_{11}+x_{12}+x_{13}+x_{14} &\le 10200 \\ x_{21}+x_{22}+x_{23}+x_{24} &\le 2000 \\ x_{31}+x_{32}+x_{33}+x_{34} &\le 4100 \end{aligned}\right\} \Leftrightarrow \boxed{\sum_{j=1}^{4}} x_{ij} \le s_i, \boxed{i = 1,2,3}$$

```python
#Capacity constraints
for i in mills:
    model.addConstr(quicksum(x[i][j] for j in warehouses)<=capacities[i],"Capacity constraint")
model.update()
```

P&P transportation problem revisited: Python + Gurobi -implementation

$$\left.\begin{aligned} x_{11}+x_{21}+x_{31} &= 2500 \\ x_{12}+x_{22}+x_{32} &= 3000 \\ x_{13}+x_{23}+x_{33} &= 5500 \\ x_{14}+x_{24}+x_{34} &= 5000 \end{aligned}\right\} \Leftrightarrow \boxed{\sum_{i=1}^{3}} x_{ij} = d_j, \boxed{j = 1,2,3,4}$$

```
#Demand constraints
for j in warehouses:
    model.addConstr(quicksum(x[i][j] for i in mills)==demands[j],"Demand constraint")
model.update()
```

P&P transportation problem revisited: Python + Gurobi -implementation

$$\min 53x_{11} + 95x_{12} + 136x_{13} + 160x_{14} + 60x_{21} + 120x_{22} + 132x_{23} + 140x_{24} + 210x_{31} + 190x_{32} + 89x_{33} + 71x_{34}$$

$$\Leftrightarrow \min \boxed{\sum_{i=1}^{3}} \boxed{\sum_{j=1}^{4}} c_{ij}x_{ij}$$

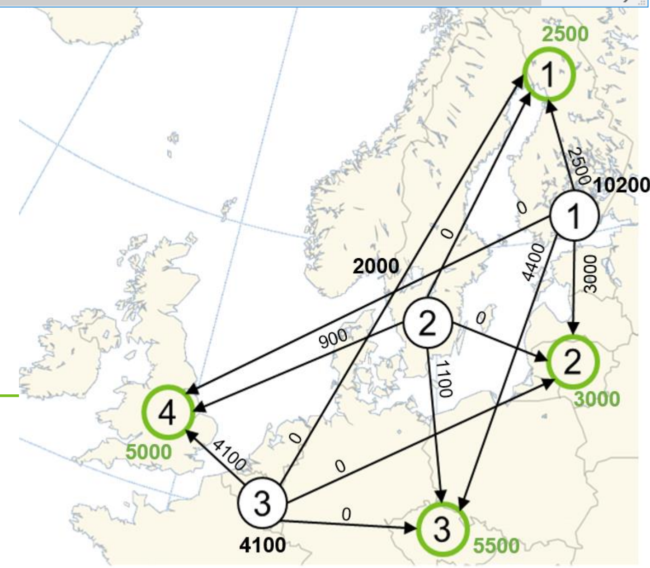
```
#objective function
model.setObjective(quicksum(costs[i][j]*x[i][j] for i in mills for j in warehouses))
model.modelSense = GRB.MINIMIZE
model.update()
```

P&P transportation problem revisited: Python + Gurobi -implementation

```python
#Find optimal solution
model.optimize()

#Collect optimal decision variable
#values to array x_optimal
x_optimal=[]
for i in mills:
    x_optimal.append([])
    for j in warehouses:
        x_optimal[i].append(x[i][j].x)

#Print optimal transportation
#cost and solution
print("Optimal transportation cost:")
print(model.objVal)
print("Optimal transportation quantities:")
print(x_optimal)
```

Windows PowerShell (5)

```
PS C:\Users\jiliesio\Dropbox (Aalto)\Business Decisions 1\2019\Lectures> gurobi '.\7 - PPcompany.py'
Using license file C:\Users\jiliesio\gurobi.lic
Academic license - for non-commercial use only
Gurobi Optimizer version 9.0.0 build v9.0.0rc2 (win64)
Optimize a model with 7 rows, 12 columns and 24 nonzeros
Model fingerprint: 0x215c731e
Coefficient statistics:
  Matrix range     [1e+00, 1e+00]
  Objective range  [5e+01, 2e+02]
  Bounds range     [0e+00, 0e+00]
  RHS range        [2e+03, 1e+04]
Presolve time: 0.00s
Presolved: 7 rows, 12 columns, 24 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    1.2620000e+06   6.400000e+03   0.000000e+00      0s
       3    1.5782000e+06   0.000000e+00   0.000000e+00      0s

Solved in 3 iterations and 0.00 seconds
Optimal objective  1.578200000e+06
Optimal transportation cost:
1578200.0
Optimal transportation quantities:
[[2500.0, 3000.0, 4400.0, 0.0], [0.0, 0.0, 1100.0, 900.0], [0.0, 0.0, 0.0, 4100.0]]
PS C:\Users\jiliesio\Dropbox (Aalto)\Business Decisions 1\2019\Lectures>
```



More examples: www.gurobi.com/resource/functional-code-examples/

Hae | Aalto-yliopisto | Functional Code Examp

gurobi.com/resource/functional-code-examples

Documentation | Downloads & Licenses | Support | My Account | English

GUROBI OPTIMIZATION

Products | Customers | Resources | Academia | Company | Partners | Free Trial

	data is already in this format.	C, C++, C#, Java, Python, VB
genconstr	Demonstrates the use of general constraints.	C, C++, C#, Java, Python, VB
multiobj	Demonstrates the use of multi-objective optimization.	C, C++, C#, Java, Python, VB
piecewise	Demonstrates the use of piecewise-linear objective functions.	C, C++, C#, Java, MATLAB, Python, R, VB
poolsearch	Demonstrates the use of solution pools.	C, C++, C#, Java, Python, VB

A few simple applications

Example	Description	Available Languages
diet	Builds and solves the classic diet problem. Demonstrates model construction and simple model modification – after the initial model is solved, a constraint is added to limit the number of dairy servings.	C, C++, C#, Java, MATLAB, Python, VB
diet2, diet3, diet4, dietmodel	Python-only variants of the diet example that illustrate model-data separation.	diet2.py, diet3.py, diet4.py, dietmodel.py
facility	Simple facility location model: given a set of plants and a set of warehouses, with transportation costs between them, this example finds the least expensive set of plants to open in order to satisfy product demand. This example demonstrates the use of MIP starts — the example computes an initial, heuristic solution and passes that solution to the MIP solver.	C, C++, C#, Java, Python, VB
netflow	A Python-only example that solves a multi-commodity network flow model. It demonstrates the use of several Python modeling constructs, including dictionaries, tuples, and tuplelist objects.	Python
portfolio	A Python-only example that solves a financial portfolio optimization model, where the historical return data is stored using the pandas package and the result is plotted using the matplotlib package. It demonstrates the use of pandas, NumPy, and Matplotlib in conjunction with Gurobi.	Python

Aalto University
School of Business

Nonlinear Programming (NLP)

- *Geometric illustration and solution*
- *Computer Solution (gradient search)*
- *Lagrange multiplier*
- *Global and local optima*
- *Convex and concave NLPs (Convex sets, convex/concave functions)*
- *Computer solution (evolutionary algorithms)*

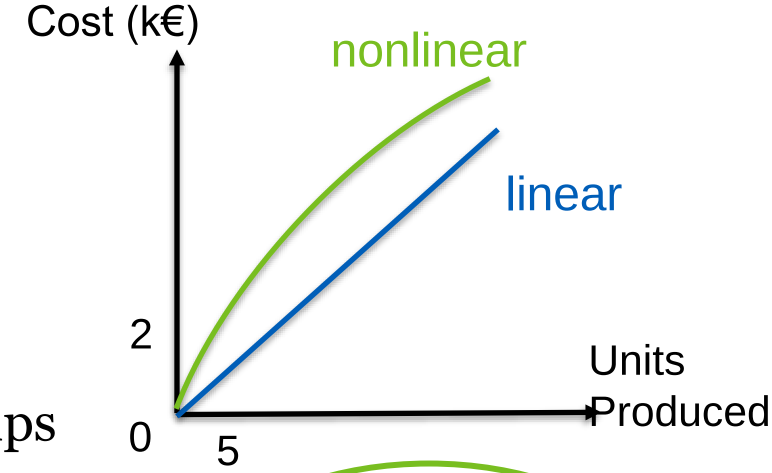
Next Monday's guest lecture:

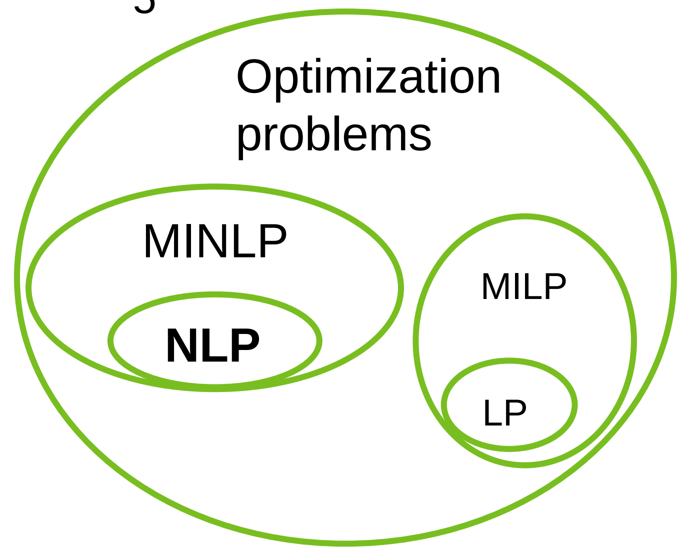
"Optimisation in Energy Transition"

Matti Vuorinen, Director, Digital Solutions in UPM Energy

Non-linear Programming (NLP)

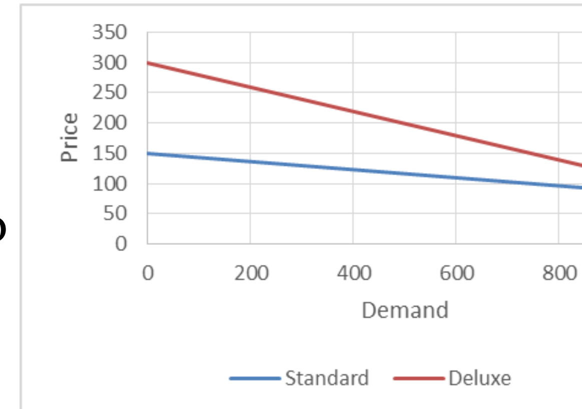
- LP models *proportional* relationships
  - Linear constraints and objective function
- NLP models *non-proportional* relationships
  - Nonlinear objective function and constraints
- Terminology
  - Even if only one of the constraints or the objective function is nonlinear $\rightarrow$ NLP model
  - NLPs can have integer variables $\rightarrow$ Mixed Integer Non-linear Programming (MINLP)





NLP Example: Production planning

- Par inc. manufactures standard and deluxe golf bags
  - Productions costs: \$70 and \$150
  - Production constraints:
    - Cutting & dying, sewing, finishing, inspection & packing
  - Demand (d) and price (p) have an inverse relationship



$$d_S = 2250 - 15p_S \Leftrightarrow p_s = (150 - \frac{1}{15}d_S)$$

$$d_D = 1500 - 5p_D \Leftrightarrow p_D = \left(300 - \frac{1}{5}d_D\right)$$

  - Total profit from producing x_S and x_D is thus:

$$\left(150 - \frac{1}{15}x_s\right)x_S - 70x_s + \left(300 - \frac{1}{5}x_D\right)x_D - 150x_D$$

$$= 80x_s - \frac{1}{15}x_S^2 + 150x_D - \frac{1}{5}x_D^2$$

NLP Example: Production planning (Cont'd)

$$\max 80x_s - \frac{1}{15}x_s^2 + 150x_D - \frac{1}{5}x_D^2$$

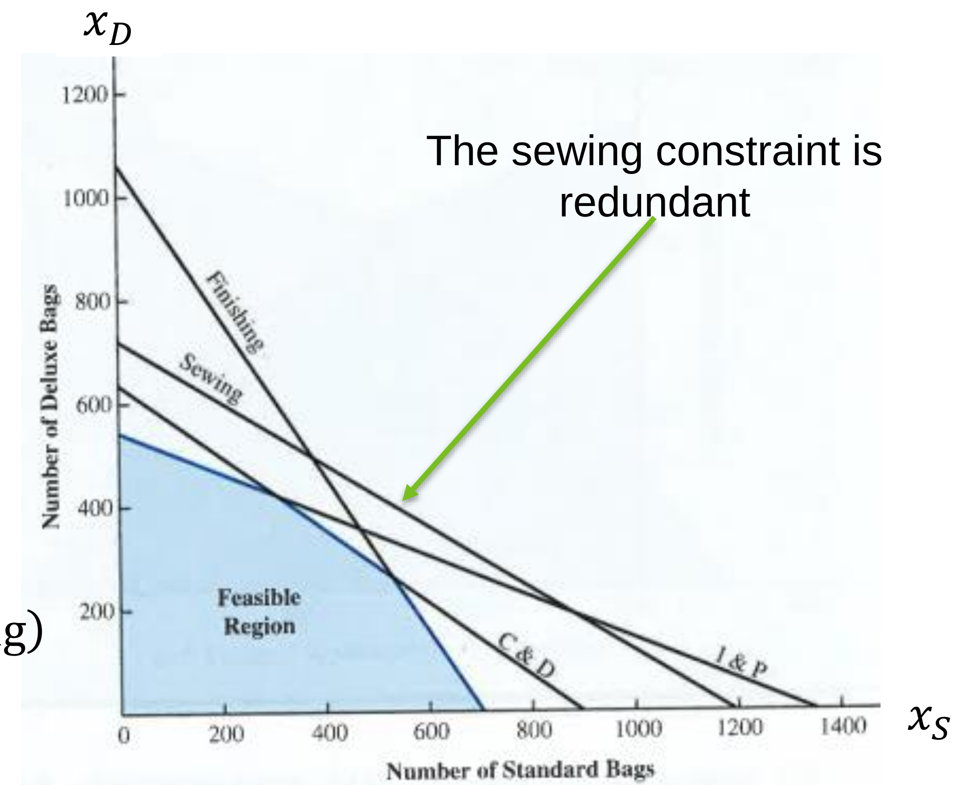
$$\frac{7}{10}x_S + x_D \leq 630 \text{ (cutting \& dying)}$$

$$\frac{1}{2}x_S + 5/6x_D \leq 600 \text{ (sewing)}$$

$$x_S + \frac{2}{3}x_D \leq 708 \text{ (finishing)}$$

$$\frac{1}{10}x_S + \frac{1}{4}x_D \leq 135 \text{ (inspection \& packing)}$$

$$x_S, x_D \geq 0$$



NLP Example: Production planning (Cont'd)

$$\max 80x_s - \frac{1}{15}x_s^2 + 150x_D - \frac{1}{5}x_D^2$$

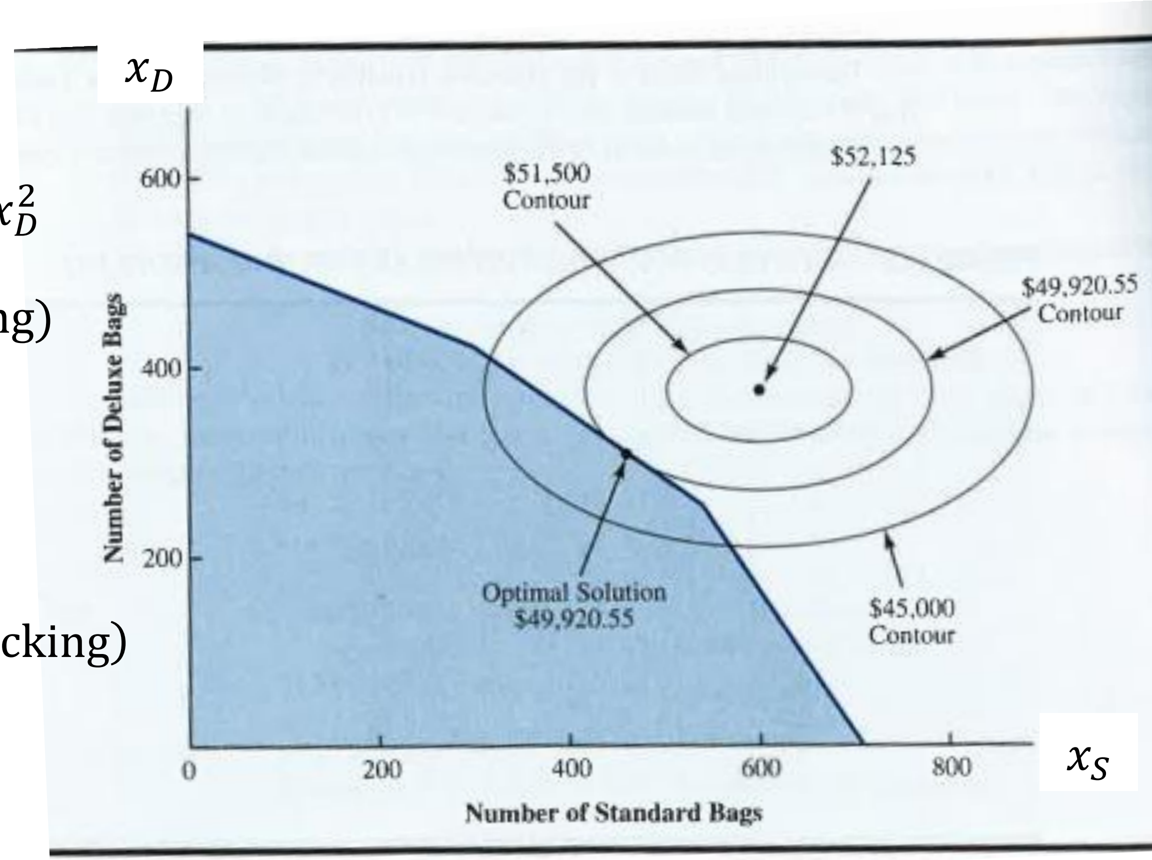
$$\frac{7}{10}x_S + x_D \leq 630 \text{ (cutting \& dying)}$$

$$\frac{1}{2}x_S + 5/6x_D \leq 600 \text{ (sewing)}$$

$$x_S + \frac{2}{3}x_D \leq 708 \text{ (finishing)}$$

$$\frac{1}{10}x_S + \frac{1}{4}x_D \leq 135 \text{ (inspection \& packing)}$$

$$x_S, x_D \geq 0$$



NLP Example: Production planning (Cont'd)

$$\max 80x_s - \frac{1}{15}x_s^2 + 150x_D - \frac{1}{5}x_D^2$$

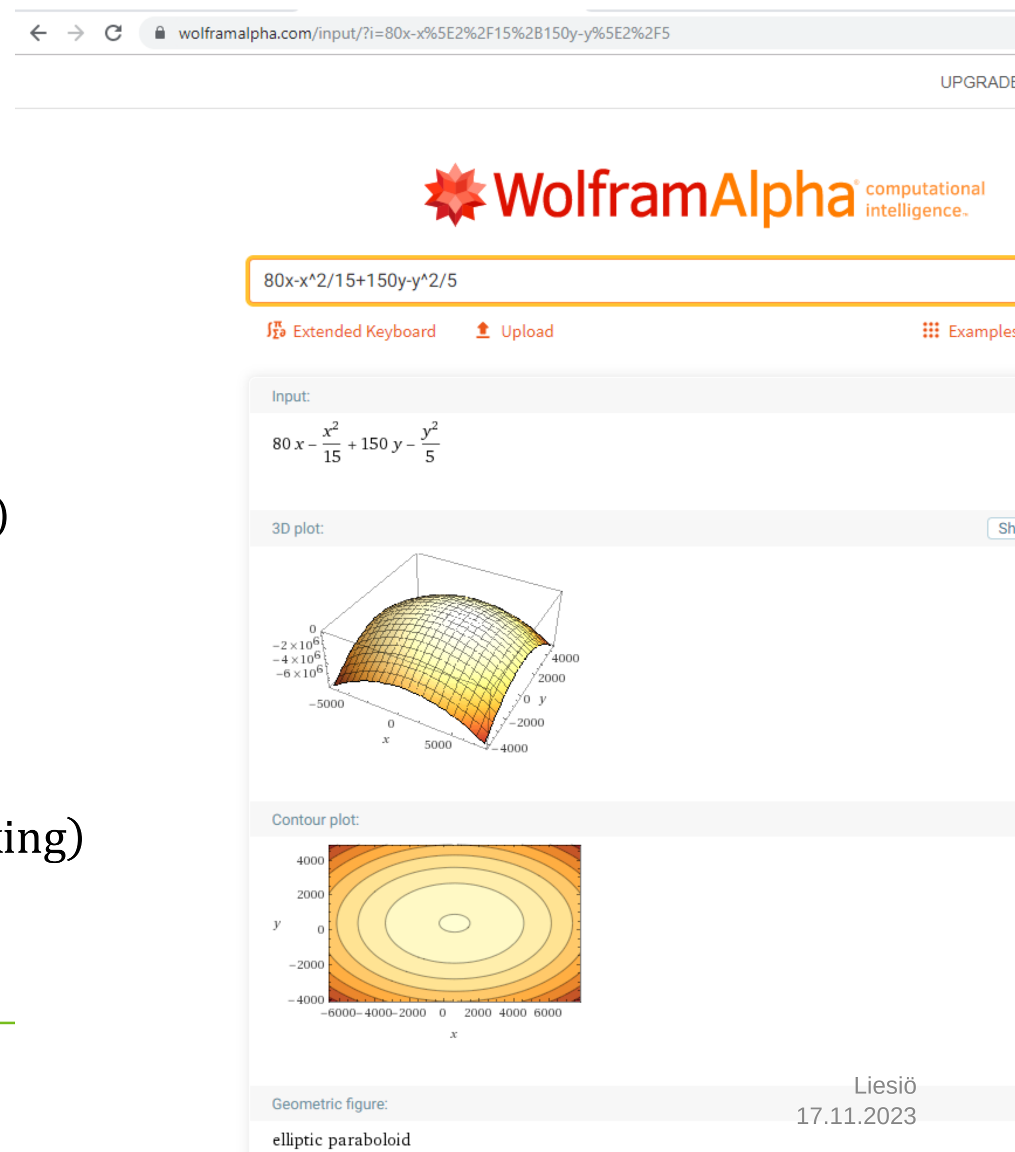
$$\frac{7}{10}x_S + x_D \leq 630 \text{ (cutting \& dying)}$$

$$\frac{1}{2}x_S + 5/6x_D \leq 600 \text{ (sewing)}$$

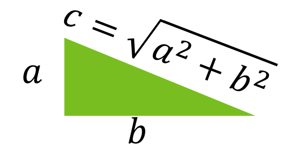
$$x_S + \frac{2}{3}x_D \leq 708 \text{ (finishing)}$$

$$\frac{1}{10}x_S + \frac{1}{4}x_D \leq 135 \text{ (inspection \& packing)}$$

$$x_S, x_D \geq 0$$



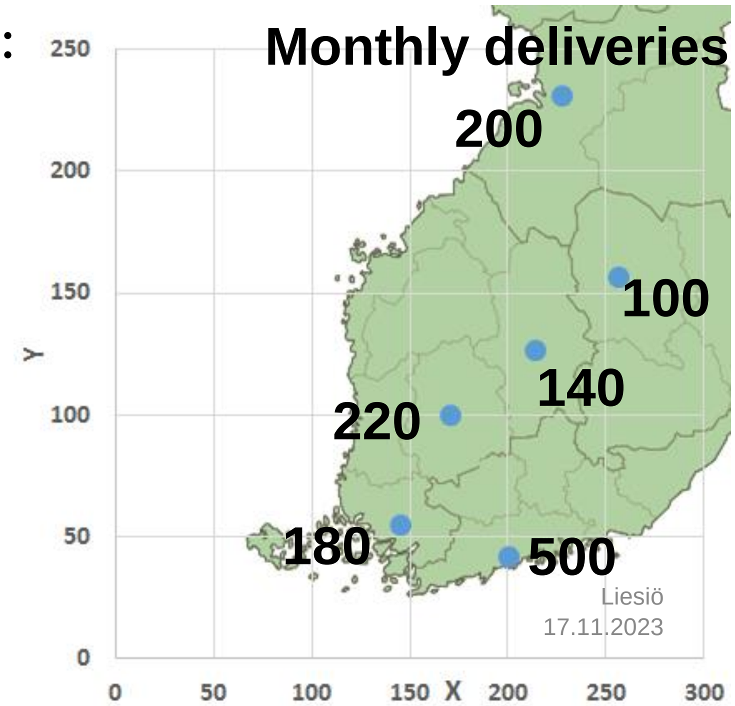
Aalto University
School of Business



NLP Example: Location problem

- T-group is planning to use unmanned helicopters to deliver groceries across Finland
  - Market research has identified five areas with most demand (see map)
    - Helsinki (inc. Vantaa and Espoo), Turku, Tampere, Oulu, Jyväskylä, and Kuopio
- T-group has constructed the following NLP to choose a location for the new distribution center serving these areas:

$$
\begin{aligned}
\min\ & 500\sqrt{(x-200)^2+(y-42)^2} \\
& +220\sqrt{(x-170)^2+(y-100)^2} \\
& +200\sqrt{(x-227)^2+(y-231)^2} \\
& +180\sqrt{(x-145)^2+(y-55)^2} \\
& +140\sqrt{(x-214)^2+(y-127)^2} \\
& +100\sqrt{(x-256)^2+(y-157)^2} \\
& x, y \geq 0
\end{aligned}
$$



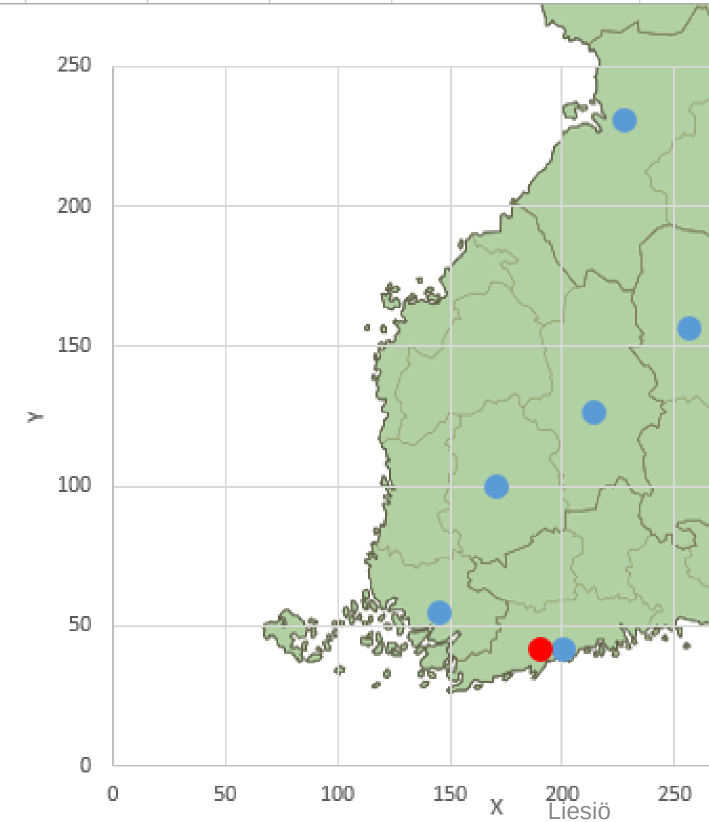
Question:

Interpret the NLP

NLP Example: Location problem – The spreadsheet

G15 =SQRT(($E$21-E15)^2+($F$21-F15)^2)

i	Area	Deliveries per month	x	y	Distance	Total dist. Per month
1	Helsinki	500	200	42	10	5000
2	Tampere	220	170	100	61	13497
3	Oulu	200	227	231	193	38518
4	Turku	180	145	55	47	8431
5	Jyväskylä	140	214	127	88	12365
6	Kuopio	100	256	157	133	13259
	DISTRIBUTION CENTER		190	42		91071



NLP Example: Location problem – The Solver

G15 =SQRT(($E$21-E15)^2+($F$21-F15)^2)

i	Area	Deliveries per month	x	y	Distance	Total dist. Per month
1	Helsinki	500	200	42	10	5000
2	Tampere	220	170	100	61	13497
3	Oulu	200	227	231	193	38518
4	Turku	180	145	55	47	8431
5	Jyväskylä	140	214	127	88	12365
6	Kuopio	100	256	157	133	13259
	DISTRIBUTION CENTER		190	42		91071

Solver Parameters

Set Objective: $H$21

To: Max Min Value Of: 0

By Changing Variable Cells: $E$21:$F$21

Subject to the Constraints:

Make Unconstrained Variables Non-Negative

Select a Solving Method: GRG Nonlinear

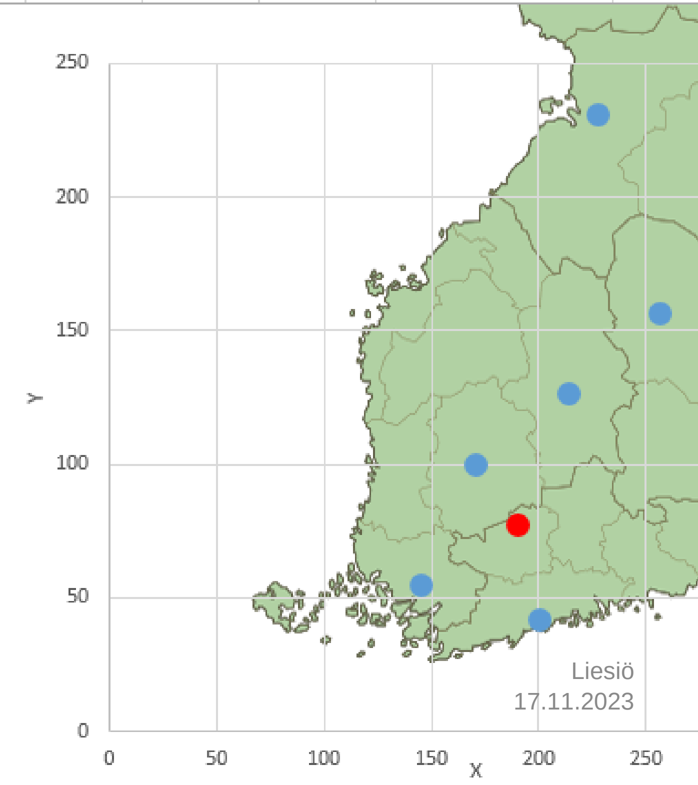
NLP Example: Location problem – Optimal solution

Question:

- Near which city should the distribution center be located?

H21 =SUM(H15:H20)

i	Area	Deliveries per month	x	y	Distance	Total dist. Per month
1	Helsinki	500	200	42	37	18437
2	Tampere	220	170	100	30	6625
3	Oulu	200	227	231	158	31580
4	Turku	180	145	55	50	9058
5	Jyväskylä	140	214	127	55	7701
6	Kuopio	100	256	157	103	10332
	DISTRIBUTION CENTER		190	77		83733



Formulas in Excel that result in a NLP model

Problem parameters (data) located in cells D1:D6

Decision variables located in cells C1:C6.

SUMPRODUCT(D4:D6, C4:C6)
[(D1 + D2) / D3] * C4
SUM(D4:D6)
2*C1 + 3*C4 + C6
C1 + C2 + C3
IF(D1>D2; 1; 0)

→ **LP model**

SUMPRODUCT(C4:C6, C1:C3)
[(C1 + C2) / C3] * D4
ABS(C1)
SQRT(C1)
C1 * C2
C1 / C2
C1 ^2
IF(C1>D2; 1; 0)

→ **NLP model**

NLP Example: Markowitz Portfolio optimization

- Hauck Financial Services allocates capital to 6 funds
  - Historical fund returns are used to construct 5 samples of possible returns for 2022 (scenarios)
  - Hauck aims at a 10% expected return with minimal risk

Question: Interpret the NLP model:

$$\min 0.2 \sum_{s=1}^{5} (r_s - \bar{r})^2$$

$$r_1 = 10.06x_A + 17.64x_B + \cdots + 24.56x_F$$
$$r_2 = 13.12x_A + 3.25x_B + \cdots + 25.32x_F$$
$$\dots$$
$$r_5 = -21.93x_A + 7.36x_B + \cdots + 17.31x_F$$
$$\bar{r} = 0.2r_1 + 0.2r_2 + 0.2r_3 + 0.2r_4 + 0.2\, r_5$$
$$x_A + x_B + x_C + x_D + x_E + x_F = 1$$
$$\bar{r} \geq 10$$
$$x_A, \dots, x_F \geq 0$$

	Historical returns				
Fund	**2017**	**2018**	**2019**	**2020**	**2021**
A	10.06	13.12	13.47	45.42	-21.93
B	17.64	3.25	7.51	-1.33	7.36
C	32.41	18.71	33.28	41.46	-23.26
D	32.36	20.61	12.93	7.06	-5.37
E	33.44	19.4	3.85	58.68	-9.02
F	24.56	25.32	-6.7	5.43	17.31

Statistics recap: For a random variable *R* that receives value r_i with probability p_i the expected value is $\mathrm{E}[R] = \sum_i p_i r_i$ and the variance is $\mathrm{Var}[R] = \sum_i p_i (r_i - E[R])^2$

NLP Example: Markowitz Portfolio optimization

L13 | =0.2*SUM(F13:J13)

	Scenario						
Fund	1	2	3	4	5	L	
						1	
A	10.06	13.12	13.47	45.42	-21.93	0.158407	
B	17.64	3.25	7.51	-1.33	7.36	0.52548	
C	32.41	18.71	33.28	41.46	-23.26	0.042065	
D	32.36	20.61	12.93	7.06	-5.37	0	
E	33.44	19.4	3.85	58.68	-9.02	0	
F	24.56	25.32	-6.7	5.43	17.31	0.274048	
Scenario-specific return	18.96	11.51	5.64	9.73	4.16	10.00	Mean return
Squared deviation from mean mean return	80.23	2.29	18.98	0.07	34.12	27.14	Variance of return
						5.21	St. Dev. of return

Return target
10

Solver Parameters

Set Objective: $L$13

To: Max / Min (selected: Min)

By Changing Variable Cells: $L$4:$L$9

Subject to the Constraints:
$L$2 = 1
$L$11 >= $P$11

Make Unconstrained Variables Non-Negative (checked)

Select a Solving Method: GRG Nonlinear

Constraints

Cell	Name	Final Value	Lagrange Multiplier
$L$2	Scenario	1	-7.523319825
$L$11	Scenario-specific return	10.00000002	6.179574311

NLP Example: Markowitz Portfolio optimization

- Increasing return target by 0.1 units increases variance by about 0.62 (=27.76-27.14).
  - Consistent with "Lagrange multiplier 6.179" on the previous slide since 0.1*6.179=0.62

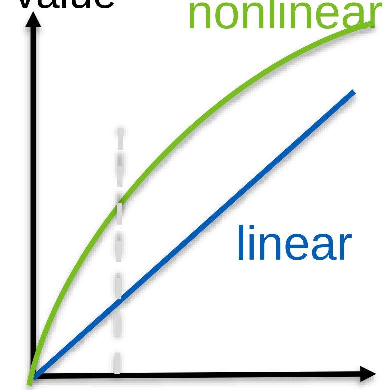
Fund	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	
						1
A	10.06	13.12	13.47	45.42	-21.93	0.152692
B	17.64	3.25	7.51	-1.33	7.36	0.518809
C	32.41	18.71	33.28	41.46	-23.26	0.049069
D	32.36	20.61	12.93	7.06	-5.37	0
E	33.44	19.4	3.85	58.68	-9.02	0
F	24.56	25.32	-6.7	5.43	17.31	0.27943
Scenario-specific return	19.14	11.68	5.71	9.80	4.17	10.10 Mean return
Squared deviation from mean mean return	81.74	2.50	19.24	0.09	35.22	27.76 Variance of return
						5.27 St. Dev. of return

Return target
10.1

Sensitivity analysis in NLP

- Shadow price in **LP** is the rate of change in the objective function as the RHS of a constraint increases (all other data unchanged)
  - This rate is constant for a range of RHS values ("range of feasibility")

- In **NLP**s this rate is called "Lagrange multiplier"
  - However, in NLP the rate does not generally remain constant
  - It can be guaranteed to hold only for the current RHS value
    - Cf. A range of feasibility where both range upper and lower bound are equal to current RHS value

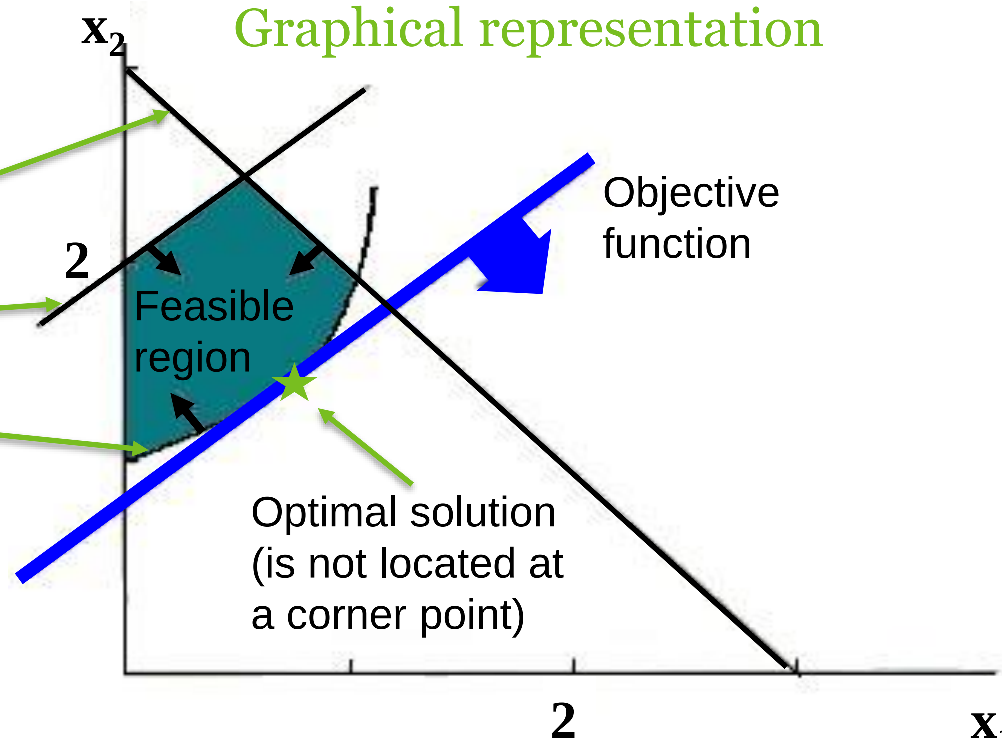


NLP Example: Linear objective function

Mathematical formulation

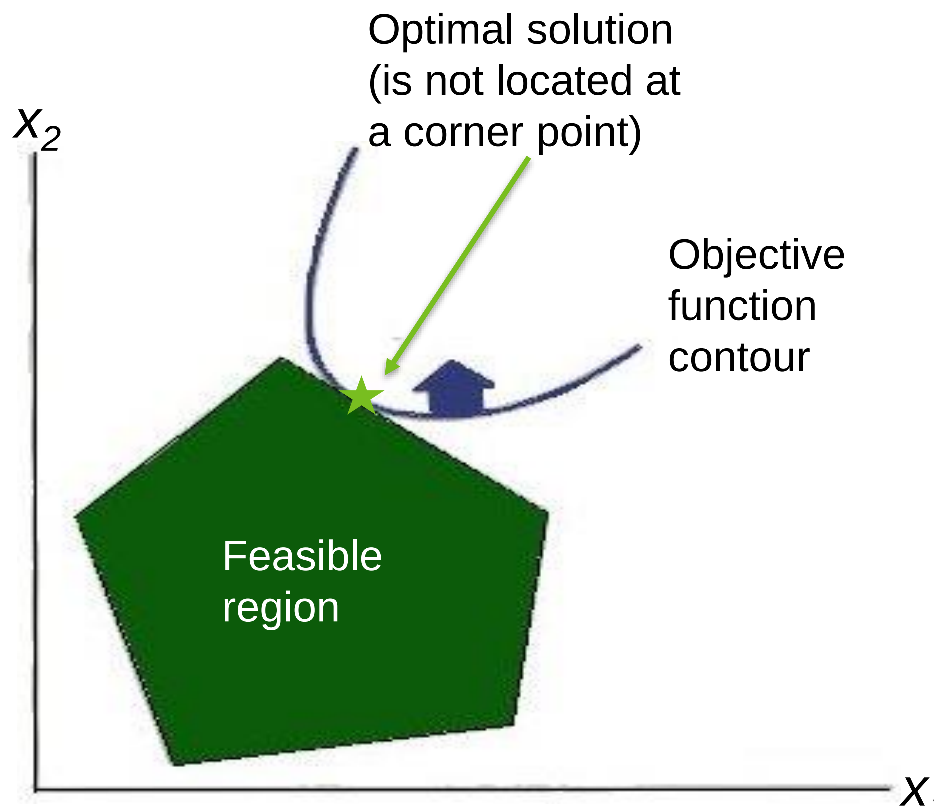
$$\max\ x_1 - x_2$$
$$\begin{aligned} s.t.\ \ & x_1 + x_2 \leq 3 \\ & -x_1 + x_2 \leq 2 \\ & -x_1^2 + x_2 \geq 1 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Graphical representation



NLP example: All constraints linear

- This NLP problem has
  - 2 decision variables
  - 5 linear constraints
  - a nonlinear objective function

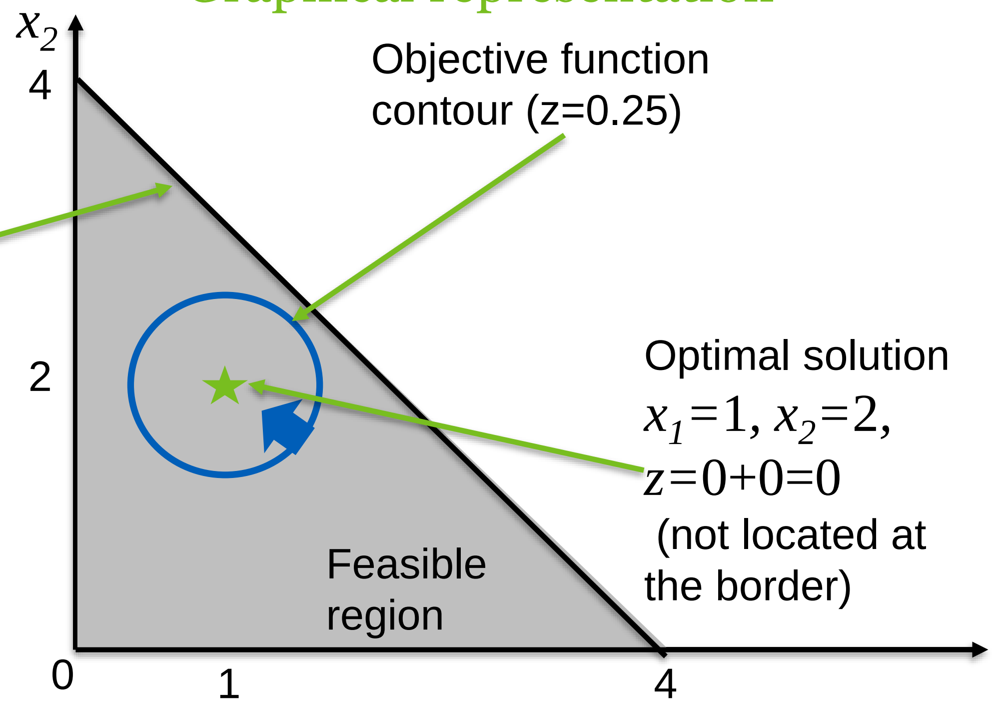


NLP Example: Optimum not on the border of the feasible region

Mathematical formulation

$$\min \; z=(x_1-1)^2 + (x_2-2)^2$$
$$s.t. \;\; x_1 + x_2 \le 4$$
$$x_1, x_2 \ge 0$$

Graphical representation



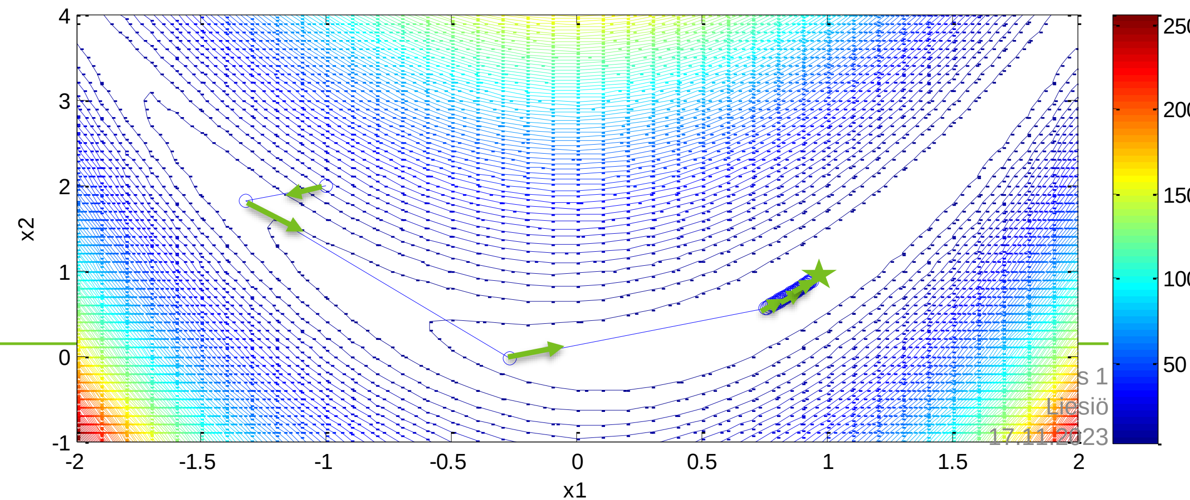
Computer solution to NLP problems

- The GRG algorithm in Solver is based on gradient search ("hill-climbing")
  - With the initial starting solution, a direction is computed that most rapidly improves the objective function value
  - Solution is moved (values of decision variables changed) to this direction until
    - a constraint boundary is encountered OR
    - the objective function value no longer improves
  - A new direction is computed with the new solution and the process is repeated until no further improvement in any direction is possible
- Example:

  min $(x_2-x_1^2)^2 + (1-x_1)^2$

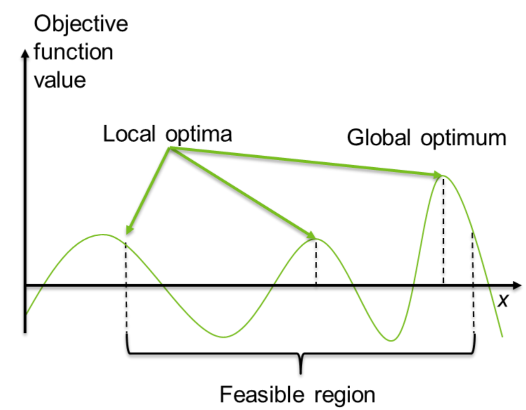
  s.t. $x_1,x_2 \geq -2$

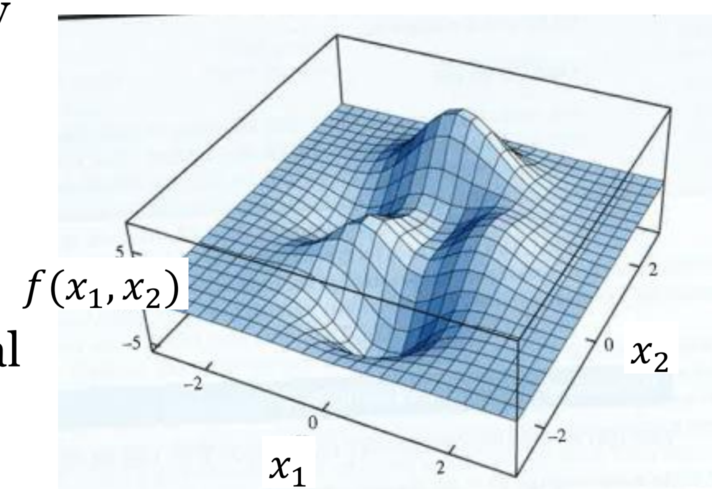
  Initial solution (-1,2)



Global and local optimal solutions

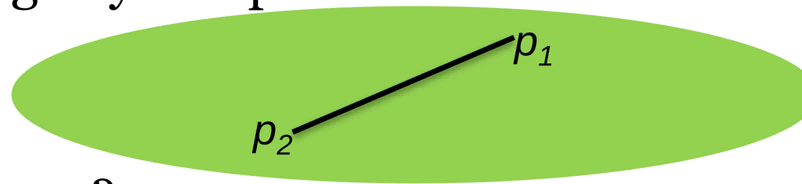
- For NLP problems we often do **not** have a guarantee that the optimal solution is a true **global** optimal solution
  - I.e. no other feasible solution provides a better objective function value
- Most NLP algorithms terminate when they have found a **local** optimal solution
  - I.e. a feasible solution such that all neighboring feasible solutions are worse
- Special case:
  - If NLP is "convex" or "concave" then any local optimal solution is a global optimal solution.



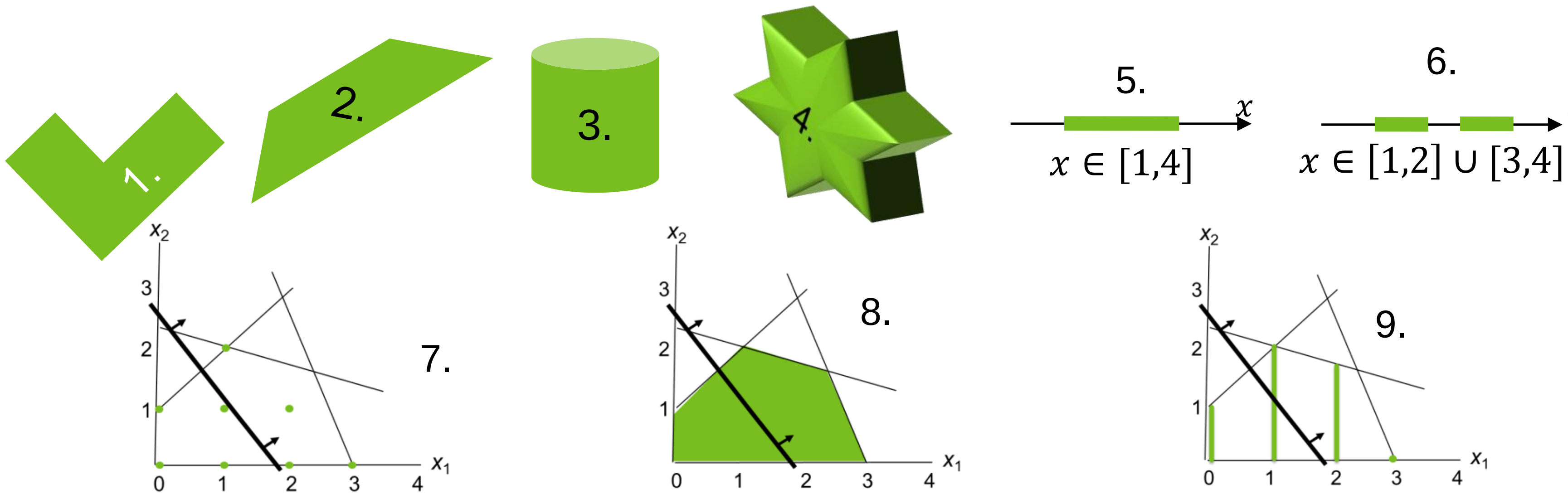


Convex sets

- **A set is convex** if a line connecting any two points in the set is contained entirely in the set

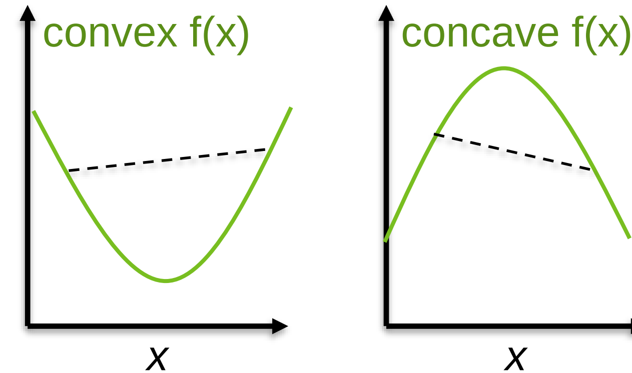


Question: Which of these sets are convex?

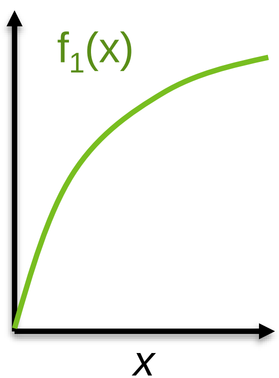


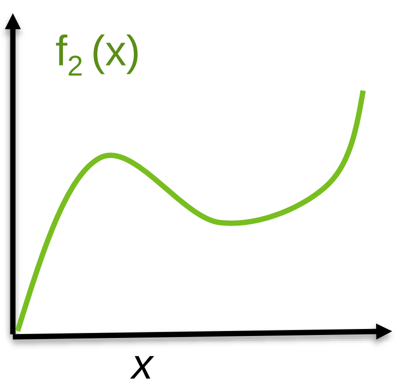
Convex and Concave functions

- **A function is convex** if a line connecting any two points lies above the function
- **A function is concave** if a line connecting any two points lies below the function

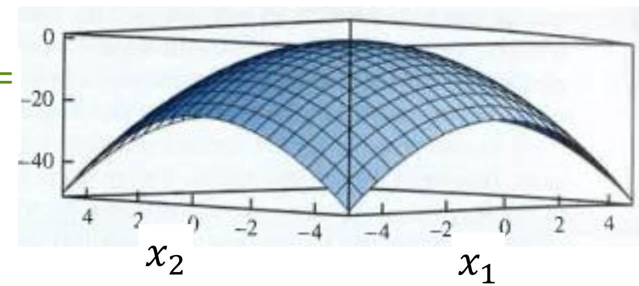


Question: Which of these 4 functions are concave and which are convex?

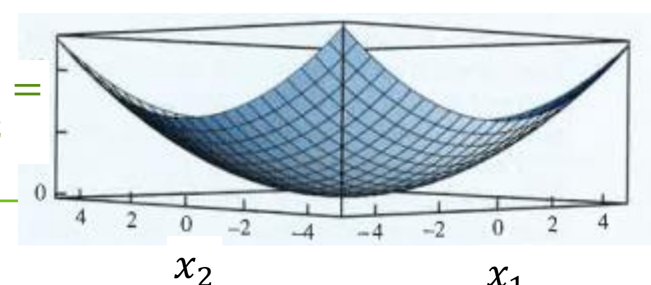




$$f_3(x_1, x_2) = -x_1^2 - x_2^2$$

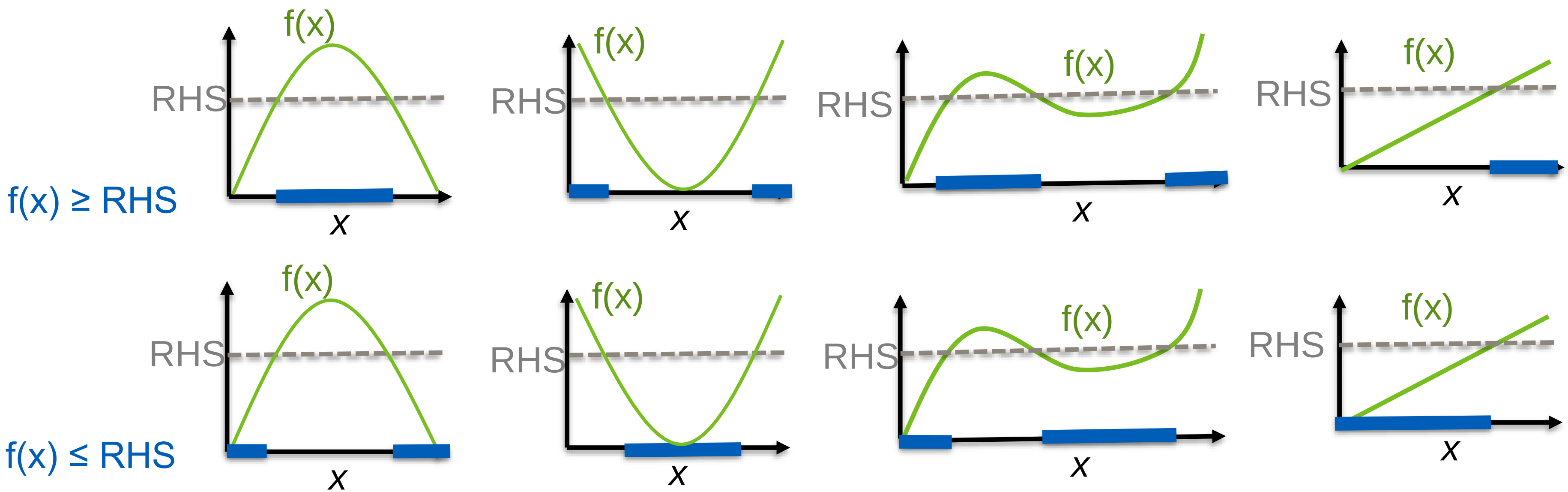


$$f_4(x_1, x_2) = x_1^2 + x_2^2$$



Convexity of sets defined by inequalities

- Constraint types in NLP: $f(x) \leq$ RHS or $f(x) \geq$ RHS (or both: $f(x)$=RHS)
- Question: When is the feasible region (= the set of values for x that satisfy the constraint) convex?



Convex and Concave NLPs

The result from the previous slide can be generalized:

- If in an NLP the LHS function of each ≤ (≥) constraint is convex (concave) then the feasible region is convex

Convex NLP:

- A convex objective function is minimized
- The feasible region is convex

Concave NLP:

- A concave objective function is maximized
- The feasible region is convex

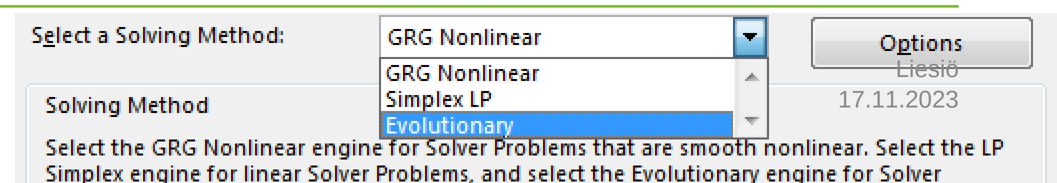
Same condition!

Property of concave and convex NLPs:

- Any local optimal solution is necessarily a global optimal solution.

Evolutionary optimization algorithms

- If an optimization problem is non-linear but its not a convex/concave NLP, how to solve it?
  - One possibility: Evolutionary Algorithms.
- Evolutionary algorithms are **heuristic**, i.e., provide a feasible solution with a "good" objective value, but no guarantees that it is optimal
  - Idea: A large set of solutions (”population”) simulated through multiple iterations (”generations”)
  - On each iteration :
    - Solutions with best objective function value (”fitness”) are combined to produce new solutions (”reproduction”)
    - Random changes to some solutions (”mutation”)
    - Infeasible solution and those with poor objective function value (”unfit”) are deleted
- Excel solver includes an evolutionary algorithm



Non-linear programming (NLP) - Summary

- Can contain a nonlinear objective function or one or more nonlinear constraints
  - Some constrains or the objective function can be linear
- Relaxation of constraints has the same effect as in LP
  - Cannot make objective function value worse
- Lagrange multiplier captures the effect of changes in the RHS of constraints
  - Holds only locally, not for a range of RHS values
- Most NLP algorithms do not ensure a global optimal solution
- For concave/convex NLPs a local optimal solution is also global
  - I.e., "max concave function"/"min convex function" over a convex feasible region

Extra slides

Equivalence between the two formulations of the Markowitz model

- Return of the ith asset is a random variable R_i such that:
  - $\mathrm{P}(R_i = r_{si}) = \frac{1}{n},\ s = 1, \dots, n$
  - $\mathrm{E}[R_i] = \bar{r}_i$

- Portfolio return:
  - In scenario s: $r_s = \sum_i x_i r_{si}$
  - Expected:

  $\bar{r} = \mathrm{E}[\sum_i x_i R_i] = \sum_i x_i \mathrm{E}[R_i] = \sum_i x_i \bar{r}_i$

Classical formulation

$$\boldsymbol{Var}\left(\sum_i x_i R_i\right) = \sum_i \sum_j x_i x_j \mathbf{Cov}(R_j, R_i)$$

$$= \sum_i \sum_j x_i x_j \left[\frac{1}{n} \sum_{s=1}^{n} (r_{si} - \bar{r}_i)(r_{sj} - \bar{r}_j)\right]$$

$$= \frac{1}{n} \sum_s \sum_i x_i (r_{si} - \bar{r}_i) \sum_j x_j (r_{sj} - \bar{r}_j)$$

$$= \frac{1}{n} \sum_s \sum_i (x_i r_{si} - x_i \bar{r}_i) \sum_j (x_j r_{sj} - x_j \bar{r}_j)$$

$$= \frac{1}{n} \sum_s (r_s - \bar{r})(r_s - \bar{r}) = \frac{1}{n} \sum_s (r_s - \bar{r})^2$$

Scenario based formulation

Multi-Objective Programming (MOP)

- *Terminology*
- *Graphical representation of MOP problems*
- *Efficient solutions: Definition*
- *Examples*
- *The weighted sum approach for solving efficient solutions*
- *Goal programming*

Next Monday's guest lecture:

"Optimisation in Energy Transition"

Matti Vuorinen, Director, Digital Solutions in UPM Energy

Multi-objective programming problems

- Many problems have multiple objectives:
  - Planning the national budget
    - improve social security, reduce debt, cut taxes, build national defense
  - Admitting students to college
    - high SAT or GMAT, high GPA, diversity
  - Planning an advertising campaign
    - reach, expenses, target groups
  - Designing a distribution system:
    - minimize transportation costs, minimize CO_2emissions
  - Choosing taxation levels
    - raise money for government, incentives for work, minimize flight of business
  - Planning an investment portfolio
    - maximize expected returns, minimize risk

MOP and MOO terminology

- **O**ptimization/**P**rogramming problems with multiple objective functions are called Multi-objective (**MO**)
  - MOLP = Multi-Objective Linear Programming
  - MOILP =
  - MOZOLP =
  - MONLP =
  - MOINLP =

- The term "Bi-objective" is sometimes used to highlight that a problem has only two objective functions
- Both the terms "criteria" and "objectives" are used
  - E.g. Multi-criteria linear programming

MOLP Example

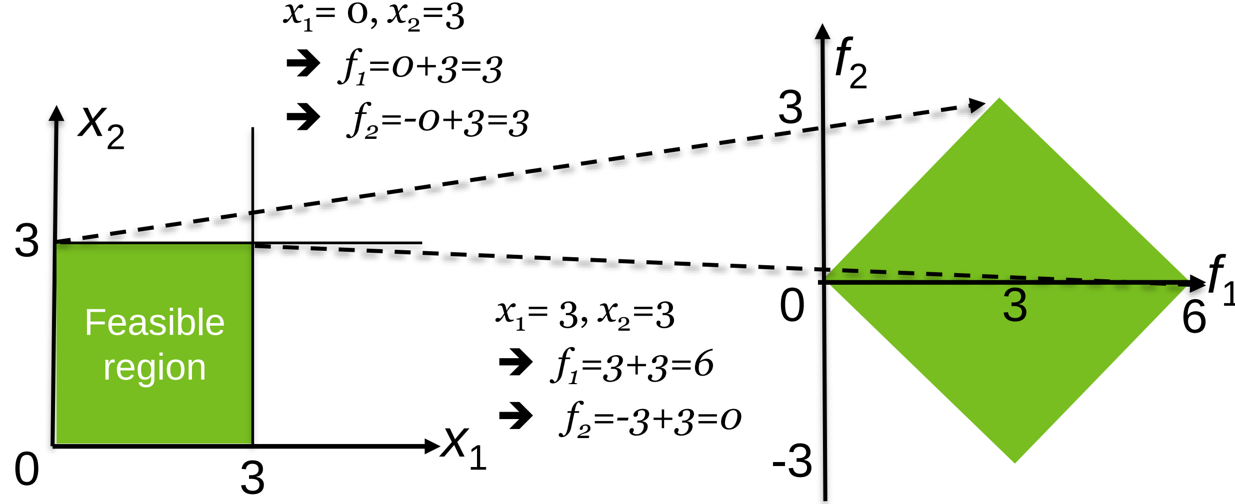
Graphical representation in the...

decision variable space

objective function space

Math. formulation

$$\text{Max } f_1 = x_1 + x_2$$
$$\text{Max } f_2 = -x_1 + x_2$$
$$\text{s.t. } x_1 \le 3$$
$$x_2 \le 3$$
$$x_1, x_2 \ge 0$$



What is an “optimal solution” to a MOP problem?

- Generally, there does not exist a feasible solution that simultaneously optimizes all the objective functions

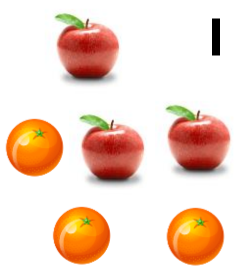
- Assume your objectives are to

  (i) maximize the number of oranges

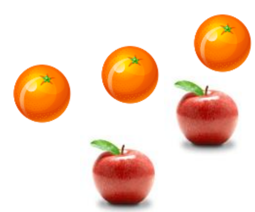
  (ii) minimize the number of apples

  Which of the fruit baskets would you choose?

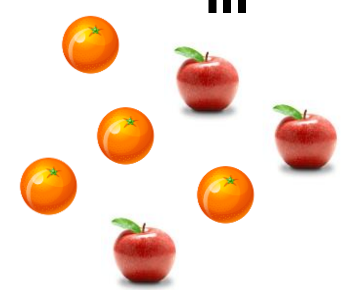
I



II



III

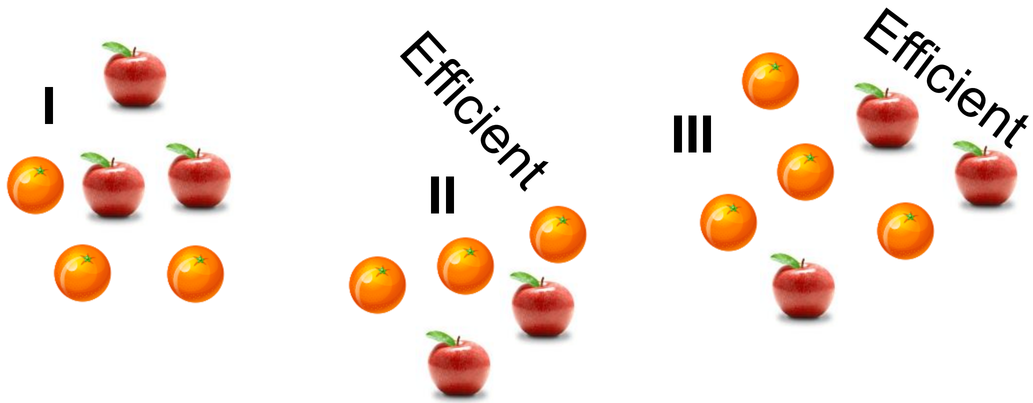


Efficient solutions: Definition

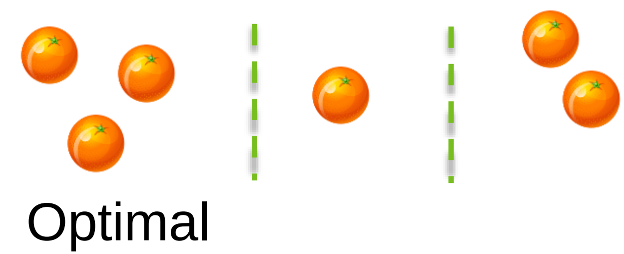
- **Definition**: A feasible solution to MOP problem is <u>efficient</u>, if there does not exist another feasible solution which yields

  (i) a better or equal value in each objective function AND

  (ii) a strictly better value in some objective function.



C.f. in single objective optimization problems a feasible solution is optimal if there does not exist another feasible solution which yields a strictly better objective function value



I is not efficient since II is
- Better or equal in both oranges and apples
- Strictly better in apples

Efficient solutions: Alternative equivalent definition

- Consider a MOP problem with n objective functions $f_1(x), \dots, f_n(x)$ to be maximized

- **Definition:** Solution x <u>dominates</u> solution x' if
  $f_i(x) \geq f_i(x')$ for all $i \in \{1, \dots, n\}$, and
  $f_i(x) > f_i(x')$ for some $i \in \{1, \dots, n\}$.

- **Definition:** A feasible solution x is <u>efficient</u> if it is not dominated by any other feasible solution

- The term "non-dominated solution" is sometimes used instead of the term "efficient solution"

Efficient solutions example: Marketing Plan

- The Supersuds Corporation is developing its next year's marketing plan
  - Spots on five TV shows purchased under limited budget
    - $x_j \in \{0,1\}, j = 1, \dots, 5$: purchase spot in jth show
  - Objective is to maximize reach in three important consumer groups
    - $f_i, i = 1, \dots, 3$: reach in the ith consumer group

Question:

- Supersuds has identified four feasible solutions: Which of them are efficient solutions?

	$f_1(x)$	$f_2(x)$	$f_3(x)$
A: $x = (1,1,1,0,0)$	1000	700	200
B: $x = (1,0,1,1,0)$	700	500	300
C: $x = (1,0,1,0,1)$	100	800	400
D: $x = (0,0,1,1,1)$	900	600	150

D is not efficient since A has a better value in each objective function

Example: Efficient solutions in MOLP

Graphical representation in the...

Math. formulation

Max $f_1 = x_1 + x_2$

Max $f_2 = -x_1 + x_2$

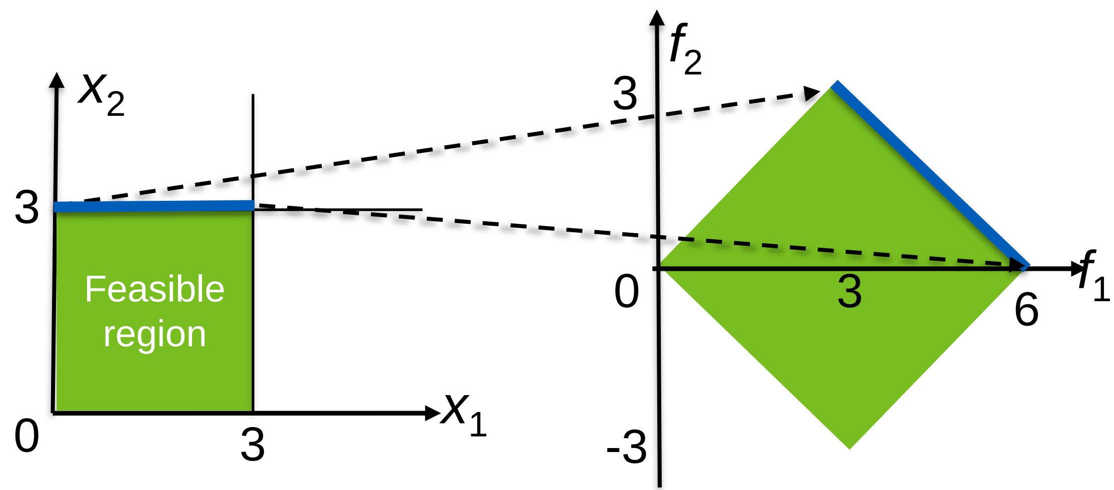
s.t. $x_1 \leq 3$

$x_2 \leq 3$

$x_1, x_2 \geq 0$

Decision variable space

Objective function space



Solving efficient solutions

- For some special classes of problems (e.g. MOLP, MOILP, MOMILP) there are algorithms that identify the entire set of efficient solutions

- Most available methods transform the MOP problem into a single objective problem and then solve it using standard algorithms
  - E.g., Simplex for LPs; B&B for MILPs; Gradient search for NLPs
  - These methods generate one efficient solution on each run
  - Approaches:
    - Set target levels (=constraints) for all but one of the objective functions
      - C.f. return at least 13% and minimize risk (=variance)
    - Maximize the **weighted sum** of the objectives functions (next slides)
      - E.g. max 1*(# of oranges)-2*(# of apples)

Weighted sum approach

General MOP formulation

$$\max \ f_1(x_1,\ldots,x_m)$$
$$\max \ f_2(x_1,\ldots,x_m)$$
$$\ldots\ldots$$
$$\max \ f_n(x_1,\ldots,x_m)$$

subject to constraints on decision variables $x_1,\ldots,x_m$

The general formulation can always be obtained by replacing "min $f_i(x_1,\ldots,x_m)$" with "max $-f_i(x_1,\ldots,x_m)$"

- Weighted sums approach
    1. Select (at random) positive weights $w_1,\ldots,w_n$ for the objective functions
    2. Solve the single objective optimization problem ➔ Solution is efficient
  - Repeat Steps 1 and 2 until enough efficient solutions have been found

Weighted sum formulation of MOP

$$\max \ \sum_{i=1}^{n} w_i f_i(x_1,\ldots,x_m)$$

subject to constraints on decision variables $x_1,\ldots,x_m$

Example: Weighed sum approach in MOLP

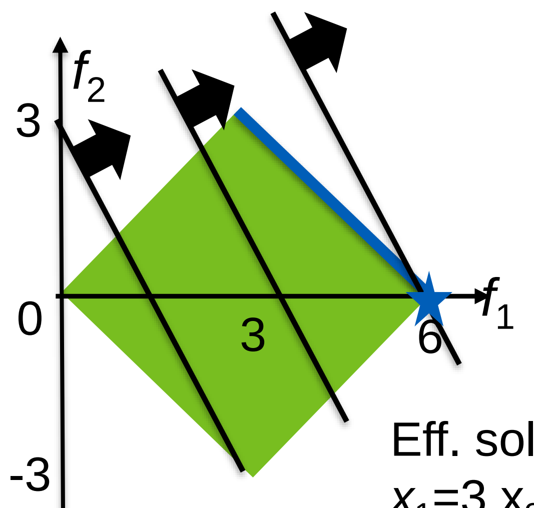
MOLP math. formulation

$$\begin{aligned} &\text{Max} \quad f_1 = x_1 + x_2 \\ &\text{Max} \quad f_2 = -x_1 + x_2 \\ &\text{s.t.} \quad x_1 \le 3 \\ &\qquad\;\; x_2 \le 3 \\ &\qquad\;\; x_1, x_2 \ge 0 \end{aligned}$$

Weighted sum formulation

$$\begin{aligned} &\text{Max} \quad w_1(x_1 + x_2) + w_2(-x_1 + x_2) \\ &\text{s.t.} \quad x_1 \le 3 \\ &\qquad\;\; x_2 \le 3 \\ &\qquad\;\; x_1, x_2 \ge 0 \end{aligned}$$

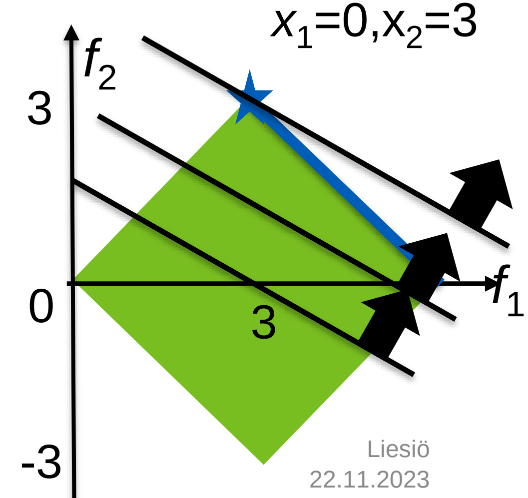
w_1=2, w_2=1: "Unit increase in objective 1 is equally important to increase of two units in objective 2"



Eff. solution x_1=3,x_2=3

w_1=1, w_2=2: "Unit increase in objective 2 is equally important to increase of two units in objective 1"

Eff. solution x_1=0,x_2=3



MONLP Example: The Markowitz Model revisited

- Hauck Financial Services allocates capital to 6 funds
  - Historical fund returns are used to construct 5 samples of possible returns for 2019 (scenarios)
  - Objective: max. expected return & min. standard deviation of return

$$\max 0.2 \sum_{s=1}^{5} r_s$$

$$\min \sqrt{0.2 \sum_{s=1}^{5} (r_s - \bar{r})^2}$$

$$r_1 = 10.06x_A + 17.64x_B + \cdots + 24.56x_F$$
$$r_2 = 13.12x_A + 3.25x_B + \cdots + 25.32x_F$$
$$\ldots$$
$$r_5 = -21.93x_A + 7.36x_B + \cdots + 17.31x_F$$
$$x_A + x_B + x_C + x_D + x_E + x_F = 1$$
$$x_A, \ldots, x_F \geq 0$$

	Historical returns				
Fund	**2013**	**2014**	**2015**	**2016**	**2017**
A	10.06	13.12	13.47	45.42	-21.93
B	17.64	3.25	7.51	-1.33	7.36
C	32.41	18.71	33.28	41.46	-23.26
D	32.36	20.61	12.93	7.06	-5.37
E	33.44	19.4	3.85	58.68	-9.02
F	24.56	25.32	-6.7	5.43	17.31

Statistics recap: For a random variable *R* that receives value r_i with probability p_i the expected value is $\mathrm{E}[R] = \sum_i p_i r_i$, the variance is $\mathrm{Var}[R] = \sum_i p_i (r_i - E[R])^2$, and the standard deviation is $\sqrt{\mathrm{Var}[R]}$

Weighed sum approach for the Markowitz Model

- MONLP with all objective functions maximized:

$$\max 0.2 \sum_{s=1}^{5} r_s$$

$$\max - \sqrt{0.2 \sum_{s=1}^{5} (r_s - \bar{r})^2}$$

$$r_1 = 10.06x_A + 17.64x_B + \cdots + 24.56x_F$$
$$r_2 = 13.12x_A + 3.25x_B + \cdots + 25.32x_F$$
$$\cdots$$
$$r_5 = -21.93x_A + 7.36x_B + \cdots + 17.31x_F$$
$$x_A + x_B + x_C + x_D + x_E + x_F = 1$$
$$x_A, \ldots, x_F \geq 0$$

- Weighted sum formulation:

$$\max w_1 \left(0.2 \sum_{s=1}^{5} r_s \right) + w_2 \cdot - \sqrt{0.2 \sum_{s=1}^{5} (r_s - \bar{r})^2}$$

$$r_1 = 10.06x_A + 17.64x_B + \cdots + 24.56x_F$$
$$r_2 = 13.12x_A + 3.25x_B + \cdots + 25.32x_F$$
$$\cdots$$
$$r_5 = -21.93x_A + 7.36x_B + \cdots + 17.31x_F$$
$$x_A + x_B + x_C + x_D + x_E + x_F = 1$$
$$x_A, \ldots, x_F \geq 0$$

Weighed sum approach for the Markowitz Model

EXPON.DIST   =P11*L11-P14*L14

	Fund	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	L
							1
	A	10.06	13.12	13.47	45.42	-21.93	0
	B	17.64	3.25	7.51	-1.33	7.36	0
	C	32.41	18.71	33.28	41.46	-23.26	0.110229
	D	32.36	20.61	12.93	7.06	-5.37	0
	E	33.44	19.4	3.85	58.68	-9.02	0.889771
	F	24.56	25.32	-6.7	5.43	17.31	0
Scenario-specific return		33.33	19.32	7.09	56.78	-10.59	21.19
ion from mean mean return		147.36	3.47	198.62	1266.97	1009.78	525.24
							22.92

		Weights	
21.19	Mean return	0.87	Weighted Sum
525.24	Variance of return		P14*L14
22.92	St. Dev. of return	0.13	

Solver Parameters

Set Objective: $R$13

To: Max (selected) Min

By Changing Variable Cells:
$L$4:$L$9

Subject to the Constraints:
$L$2 = 1

Make Unconstrained Variables Non-Negative (checked)

Select a Solving Method: GRG Nonlinear

Weighed sum approach for the Markowitz Model

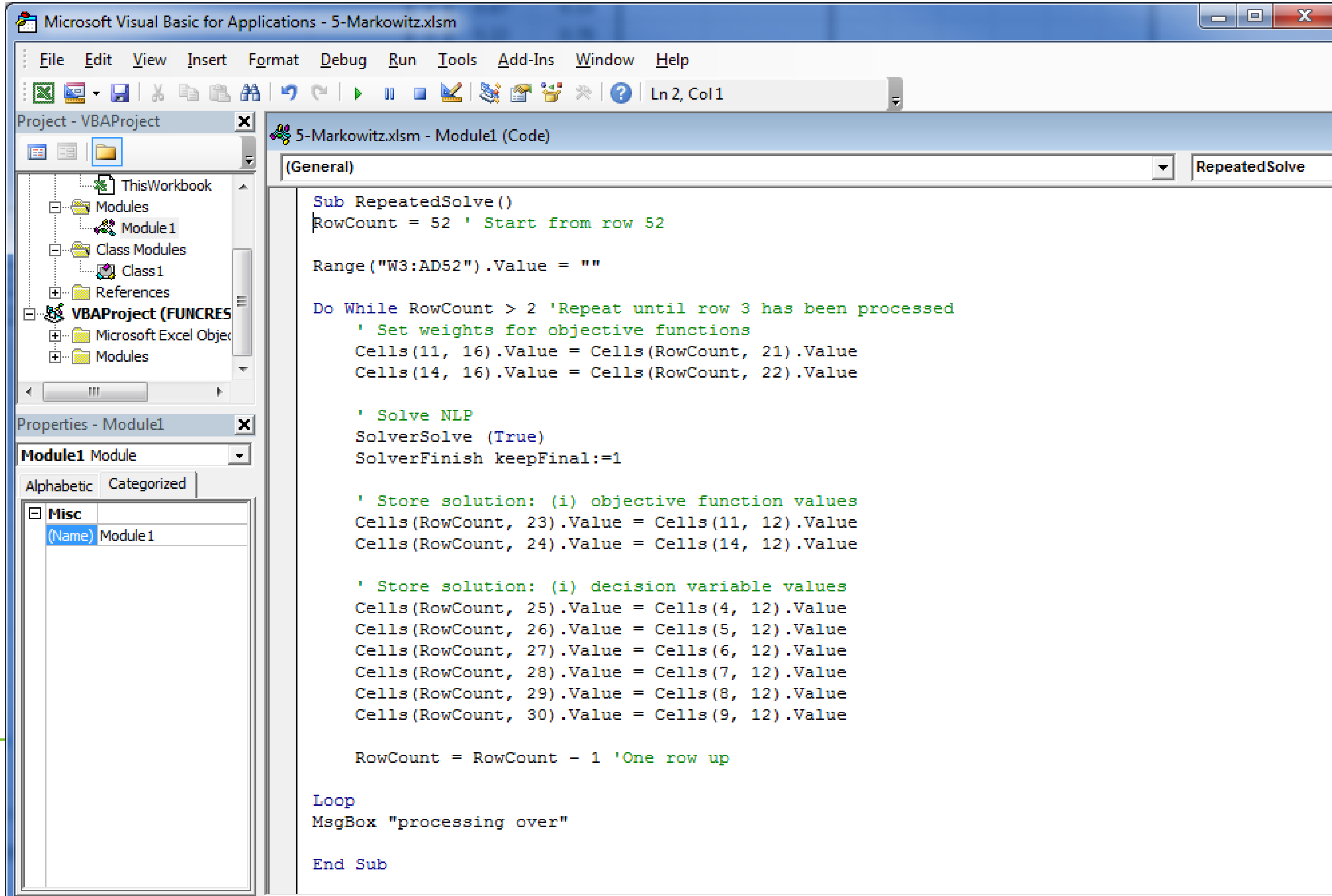
- Lets solve it for 50 different weights generated randomly…

	K	L	M	N	O	P	Q	R	S
1									
2		1							
3									
4		0							
5		0							
6		0.11023							
7		0							
8		0.88977							
9		0							
10						Weights			
11		21.19	Mean return			0.87		Weighted	
12								Sum	
13		525.24	Variance of return					15.55	
14		22.92	St. Dev. of return			0.13			

	T	U	V	W	X	Y	Z	AA	AB	AC	AD
1		Obj. Fun. Weights		Opt. Obj. Fun. Values		Optimal solution (decision var. values)					
2		w_1	w_2	E[R]	Std. R	A	B	C	D	E	F
3	1	0.87	0.13								
4	2	0.22	0.78								
5	3	0.46	0.54								
6	4	1.00	0.00								
7	5	0.41	0.59								
8	6	0.61	0.39								
9	7	0.39	0.61								
10	8	0.70	0.30								
11	9	0.48	0.52								
12	10	0.21	0.79								
13	11	0.58	0.42								
14	12	0.24	0.76								

Weighed sum approach for the Markowitz Model

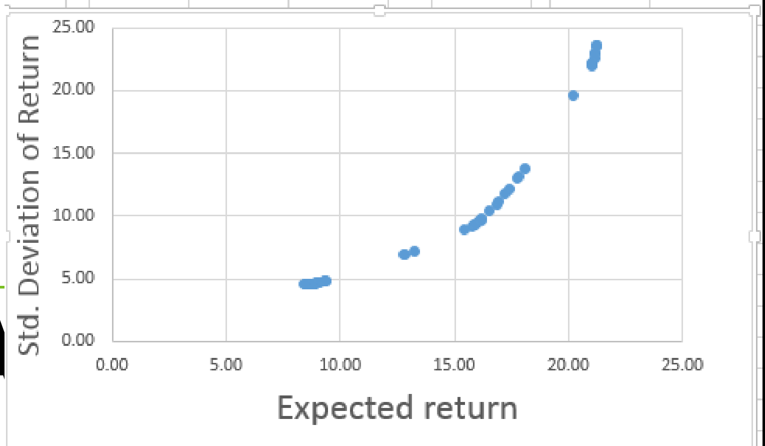
- A lot of work … lets use a macro!



Weighed sum approach for the Markowitz Model

- Running the macro gives 50 efficient solutions

J	K	L	M	N	O	P	Q	R	S
		1							
5									
1.93		0							
.36		0							
3.26		0.110229							
.37		0							
.02		0.889771							
'.31		0							
						Weights			
).59		21.19	Mean return			0.87		Weighted	
								Sum	
)9.78		525.24	Variance of return					15.55	
		22.92	St. Dev. of return			0.13			



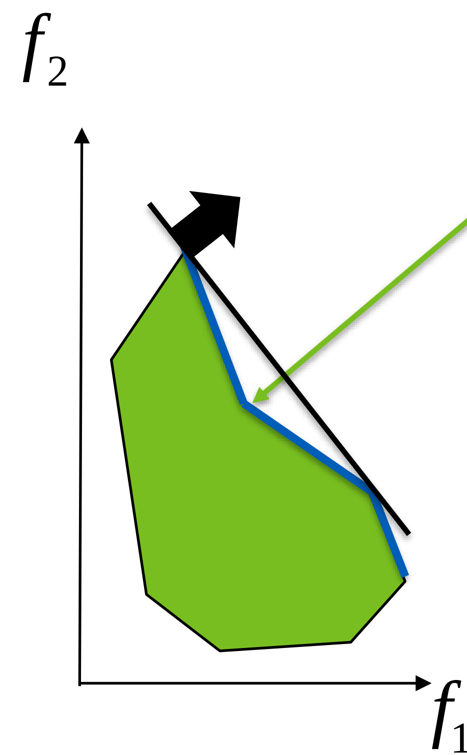
T	U	V	W	X	Y	Z	AA	AB	AC	AD
	Obj. Fun. Weights		Opt. Obj. Fun. Values		Optimal solution (decision var. values)					
	w_1	w_2	E[R]	Std. R	A	B	C	D	E	F
1	0.87	0.13	21.19	22.92	0.00	0.00	0.11	0.00	0.89	0.00
2	0.22	0.78	8.87	4.63	0.18	0.65	0.00	0.00	0.00	0.17
3	0.46	0.54	15.46	8.94	0.00	0.03	0.34	0.00	0.00	0.63
4	1.00	0.00	21.27	23.55	0.00	0.00	0.00	0.00	1.00	0.00
5	0.41	0.59	13.24	7.22	0.00	0.29	0.25	0.00	0.00	0.46
6	0.61	0.39	16.21	9.78	0.00	0.00	0.41	0.00	0.00	0.59
7	0.39	0.61	12.83	6.94	0.00	0.33	0.24	0.00	0.00	0.43
8	0.70	0.30	17.43	12.20	0.00	0.00	0.40	0.00	0.17	0.44
9	0.48	0.52	15.78	9.22	0.00	0.00	0.35	0.00	0.00	0.65
10	0.21	0.79	8.86	4.63	0.18	0.65	0.00	0.00	0.00	0.17
11	0.58	0.42	16.05	9.53	0.00	0.00	0.39	0.00	0.00	0.61
12	0.24	0.76	8.94	4.65	0.18	0.64	0.00	0.00	0.00	0.18
13	0.27	0.73	9.04	4.68	0.18	0.62	0.00	0.00	0.00	0.19
[illegible]	0.36	0.64	9.39	4.85	0.19	0.57	0.00	0.00	0.00	0.24
15	0.90	0.10	21.27	23.55	0.00	0.00	0.00	0.00	1.00	0.00
[illegible]	0.02	0.98	8.46	4.57	0.17	0.72	0.00	0.00	0.00	0.11
[illegible]	0.02	0.98	8.47	4.57	0.17	0.72	0.00	0.00	0.00	0.11
[illegible]	0.18	0.82	8.77	4.60	0.18	0.67	0.00	0.00	0.00	0.16
19	0.67	0.33	16.85	10.93	0.00	0.00	0.40	0.00	0.09	0.51
20	0.72	0.28	18.07	13.77	0.00	0.00	0.39	0.00	0.25	0.36
21	0.13	0.87	8.66	4.58	0.17	0.69	0.00	0.00	0.00	0.14
22	0.24	0.76	8.94	4.65	0.18	0.64	0.00	0.00	0.00	0.18
23	0.68	0.32	16.97	11.18	0.00	0.00	0.40	0.00	0.10	0.49
24	0.69	0.31	17.24	11.76	0.00	0.00	0.40	0.00	0.14	0.46
25	0.10	0.90	8.61	4.58	0.17	0.69	0.00	0.00	0.00	0.13
26	0.71	0.29	17.82	13.14	0.00	0.00	0.39	0.00	0.22	0.39
27	0.10	0.90	8.61	4.58	0.17	0.69	0.00	0.00	0.00	0.13

Cautionary note about the weighted sum approach

- Every solution generated by the weighted sum approach is efficient
  - Assuming all weights are strictly positive

- However, if the feasible region is not convex, there can be efficient solutions that the weighted approach cannot find
  - These solution do not maximize the weighted sum for any weights

- For instance, MILP, ILP and BLP problems do not usually have a convex feasible region



Here are efficient solutions that the weighted sum approach cannot find

Goal programming (GP)

- Idea: set goal for each objective function
  - The goals are listed in the order of their importance.
  - Begin by minimizing deviation from the most important goal
  - Do the same for the second most important goal, but require that the deviation from the from the first goal is not increased
  - Continue to the following goals, always requiring that the deviations from the previously optimized goals do not increase

Major drawback: May lead to a solution that is not efficient

- Different flavors exist: Preemptive GP (above), weighted GP,…

Multi-Objective Programming - Summary

- Optimization problems with multiple objective functions

- Instead of an optimal solution there is a (possibly infinite) set of efficient solutions
  - Definition: A feasible solution is efficient if no other feasible solution provides (i) an equal or better value in each objective function, and (ii) a strictly better value in at least one objective function
  - Terms "Pareto optimal solution" and "Non-dominated solution" widely used as synonyms for "Efficient solution"

- The are several methods for generating efficient solutions which make use of standard (i.e., single objective) solution algorithms
    - E.g. weighted sums approach