Discrete optimization – exact methods

Mikael Rönnqvist
Norwegian School of Economics and Business Administration
Bergen, Norway
mikael.ronnqvist@nhh.no

eVITA Winter School 2009 - Optimization

Agenda (Wednesday 15-16:30, 16:30-18:30)

- Applications – introduction
- Modeling and standard models
- Solution methods
- Applications – revisited
  - Results/Impacts
- Case (if time)
- Concluding comments

Objective

- Understanding of the main solution approaches for discrete optimization and their characteristics.
- Understanding of modeling and standard discrete optimization models.
- Insights in some applications and how discrete optimization can be used to solve them.

Seminar based on material from

- J. Lundgren, M. Rönnqvist and P. Värbrand, Optimeringslära, Studentlitteratur, Sweden, 537 pages, 2008. (English version available during Spring 2009)
  - P. Eveborn, M. Rönnqvist, M. Almroth, M. Eklund, H. Einarsdóttir and K. Lidèn, Operations Research (O.R.) Improves Quality and Efficiency in Home Care, to appear in *Interfaces*
  - H. Gunnarsson, M. Rönnqvist and D. Carlsson, A combined terminal location and ship routing problem, *Journal of the Operational Research Society*, Vol. 57, 928-938, 2006.
  - D. Bredström, J. T. Lundgren, M. Rönnqvist, D. Carlsson and A. Mason, Supply chain optimization in the pulp mill industry – IP models, column generation and novel constraint branches, *European Journal of Operational Research*, Vol 156, pp 2-22, 2004.
  - H. Broman, M. Frisk and M. Rönnqvist, Supply chain planning of harvest operations and transportation after the storm Gudrun, to appear in *INFOR*

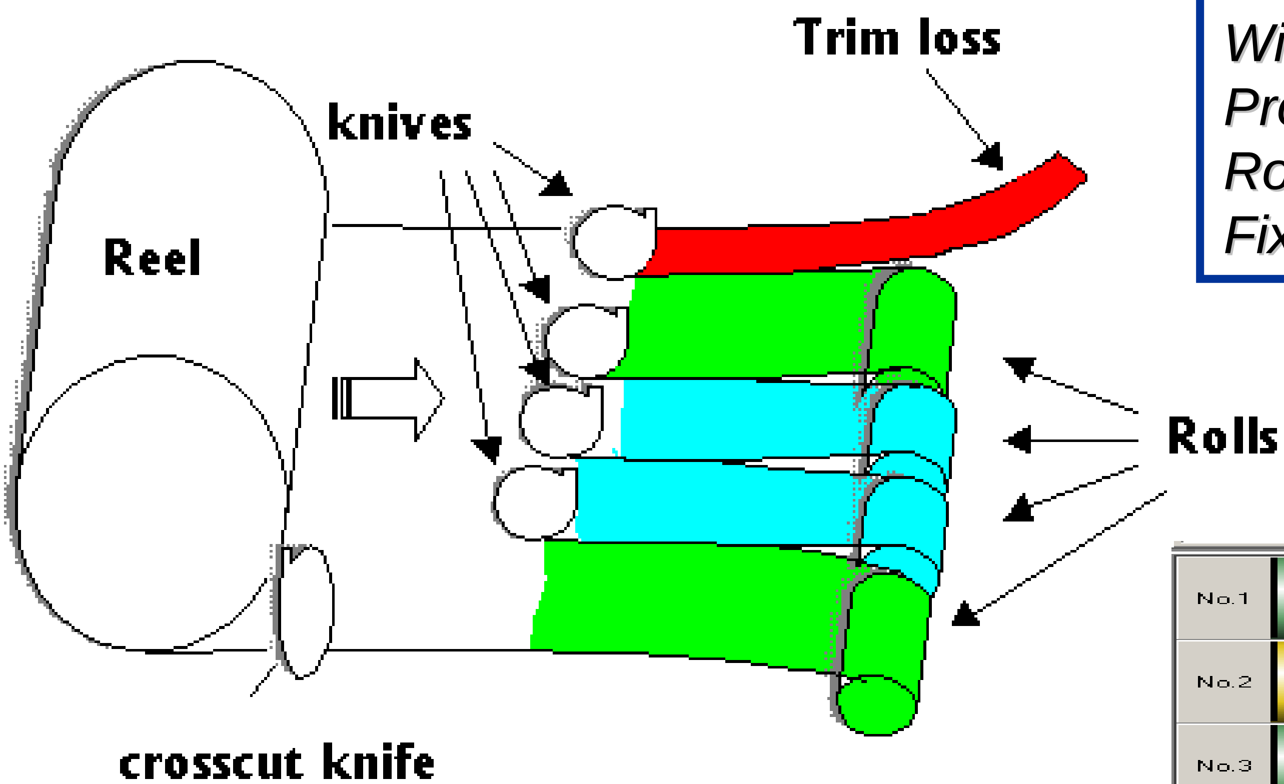
Applications

Applications

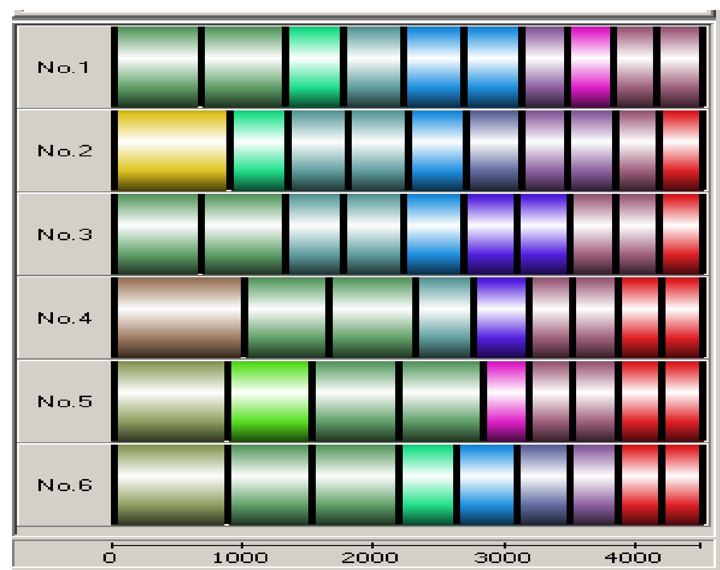
- Paper roll cutting
- 2D - Board cutting
- Terminal location
- Production planning
- Routing/ Scheduling
- Game

Paper roll cutting

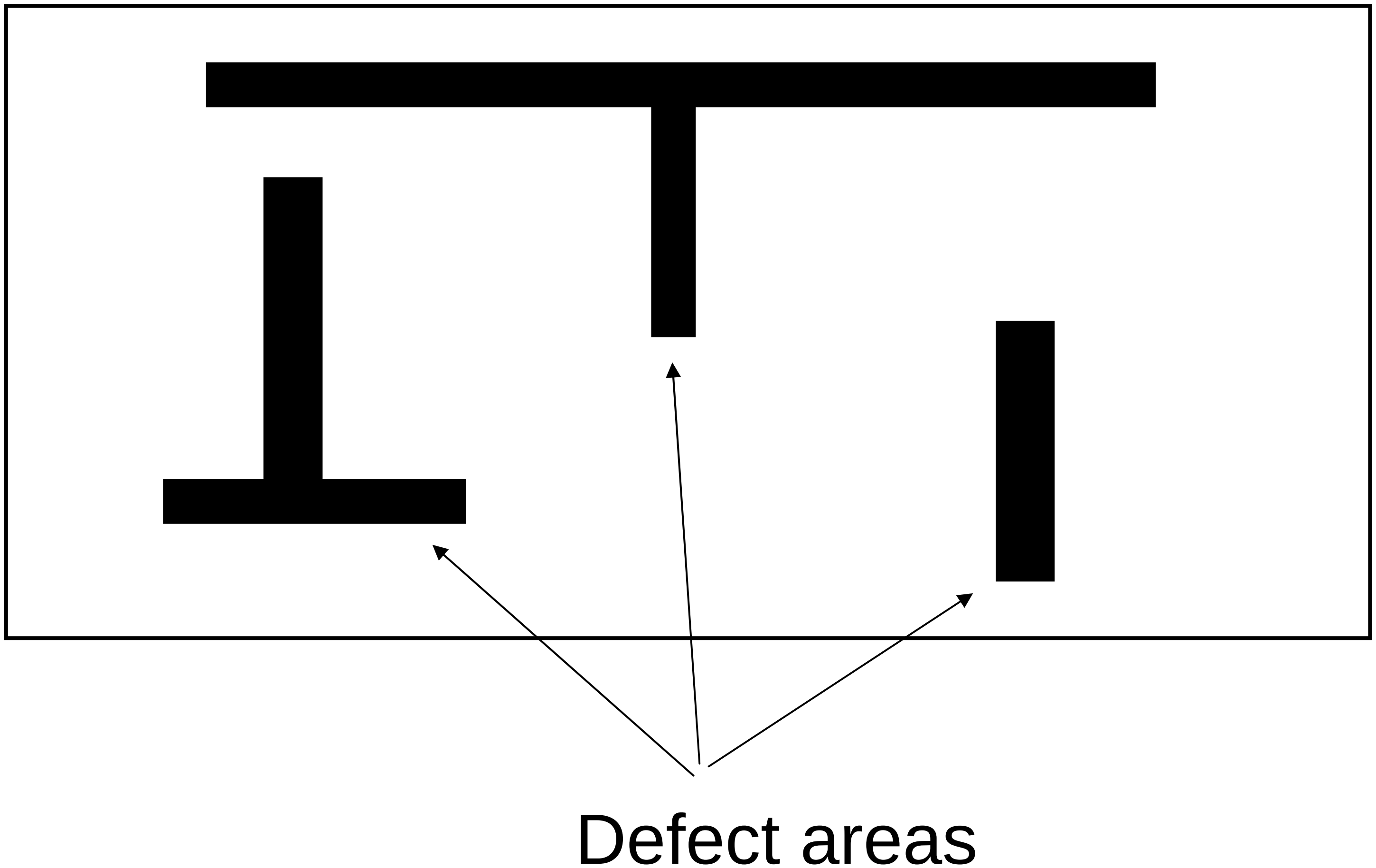
Roll cutting at paper mills

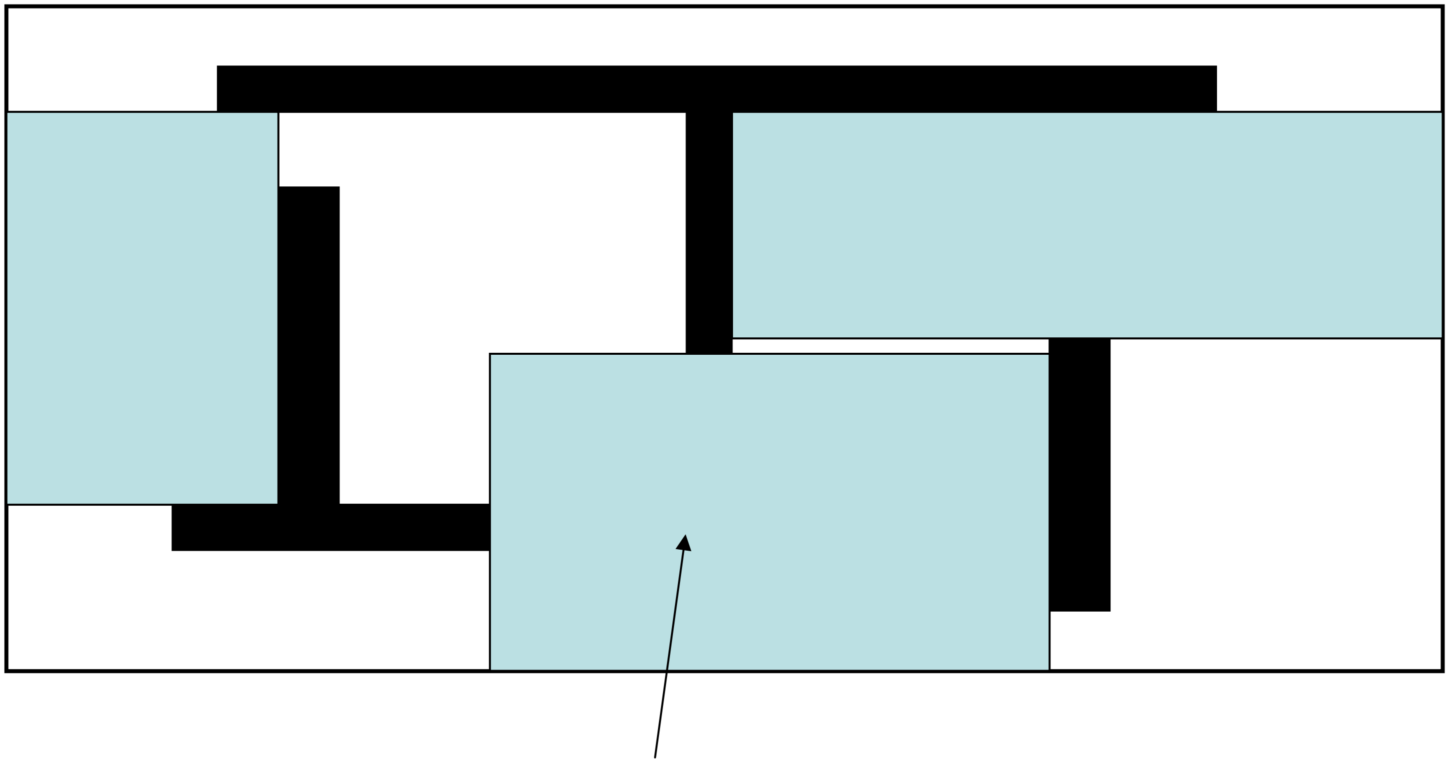


Length: 30,000 meters
Width: 5-10 meters
Products: 0.3-1.0 m
Roll: 5,000 meters
Fixed demand



2D Board cutting

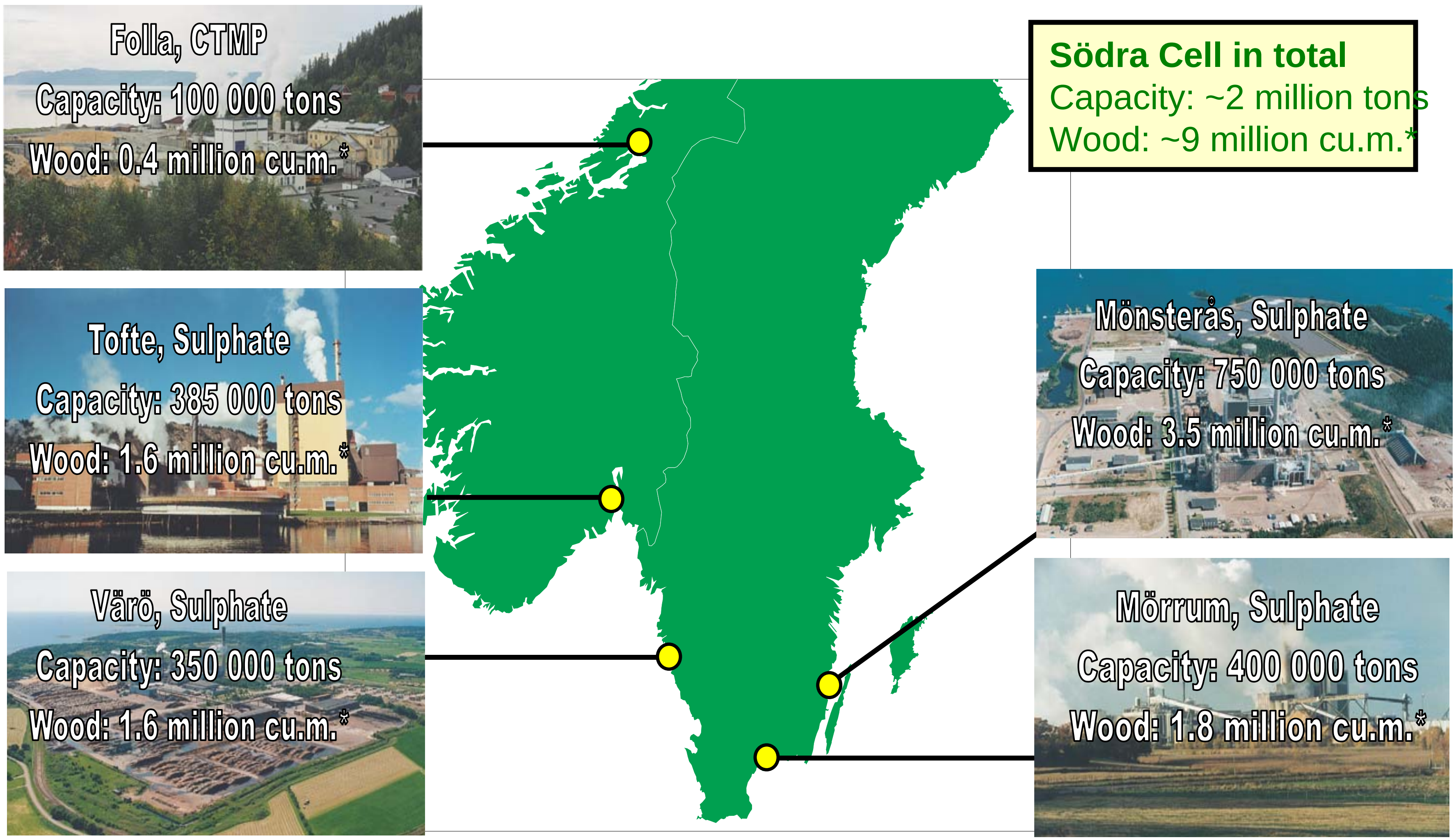

Defect areas



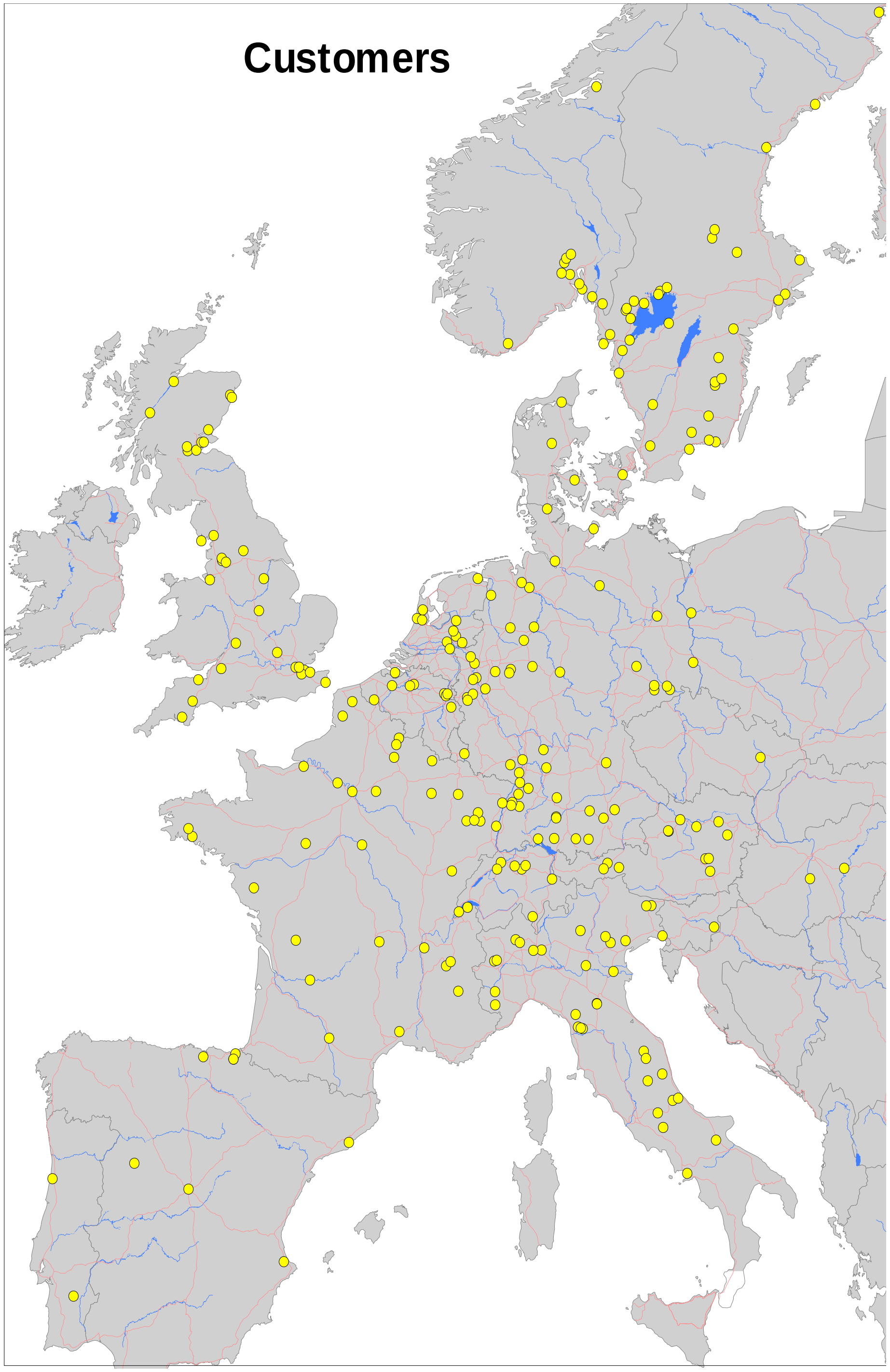
Weighted area of products determines the board's quality/value

Terminal location

Case: Södra Cell - mills



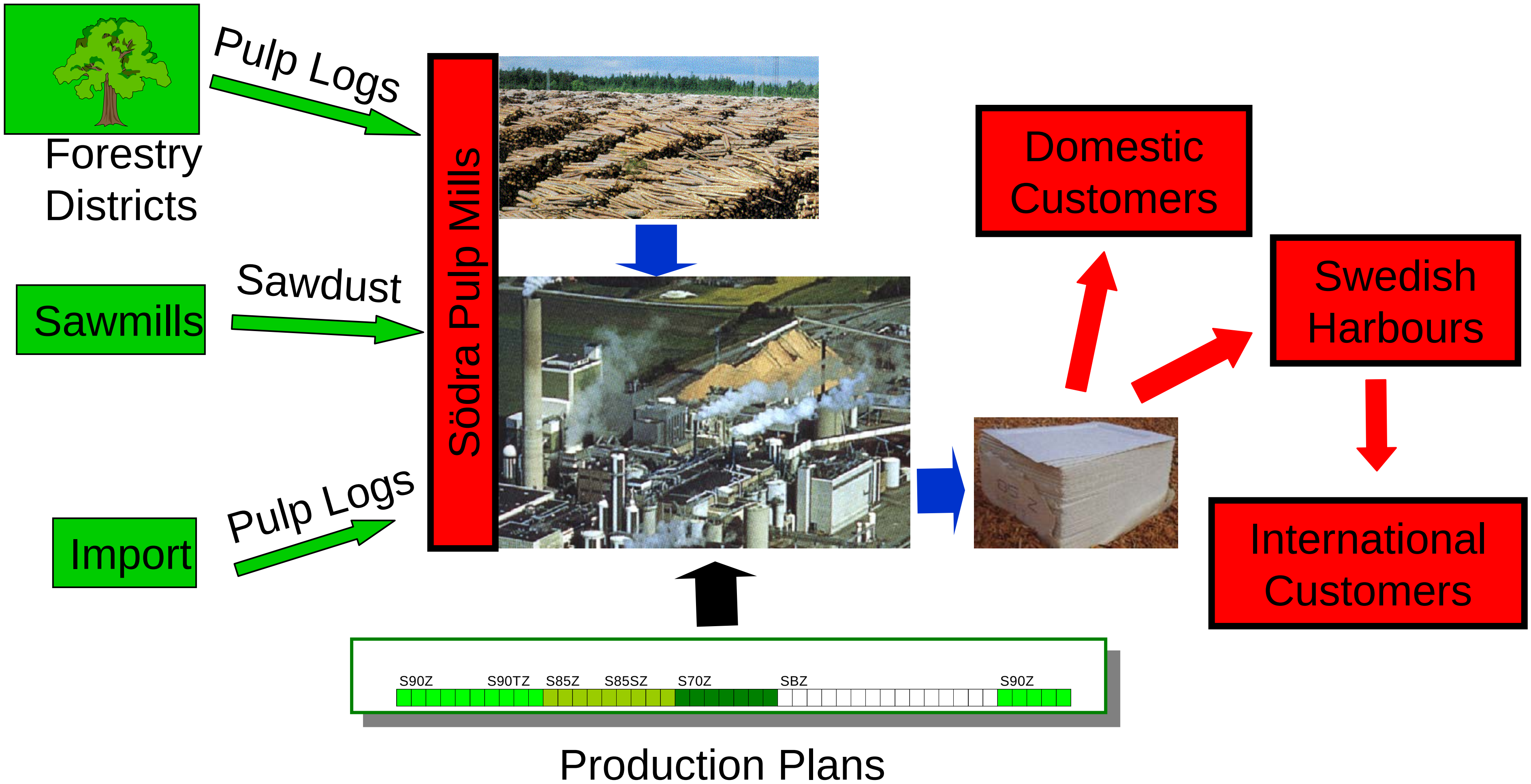
cu.m. = cubic meters, solid under bark


Customers

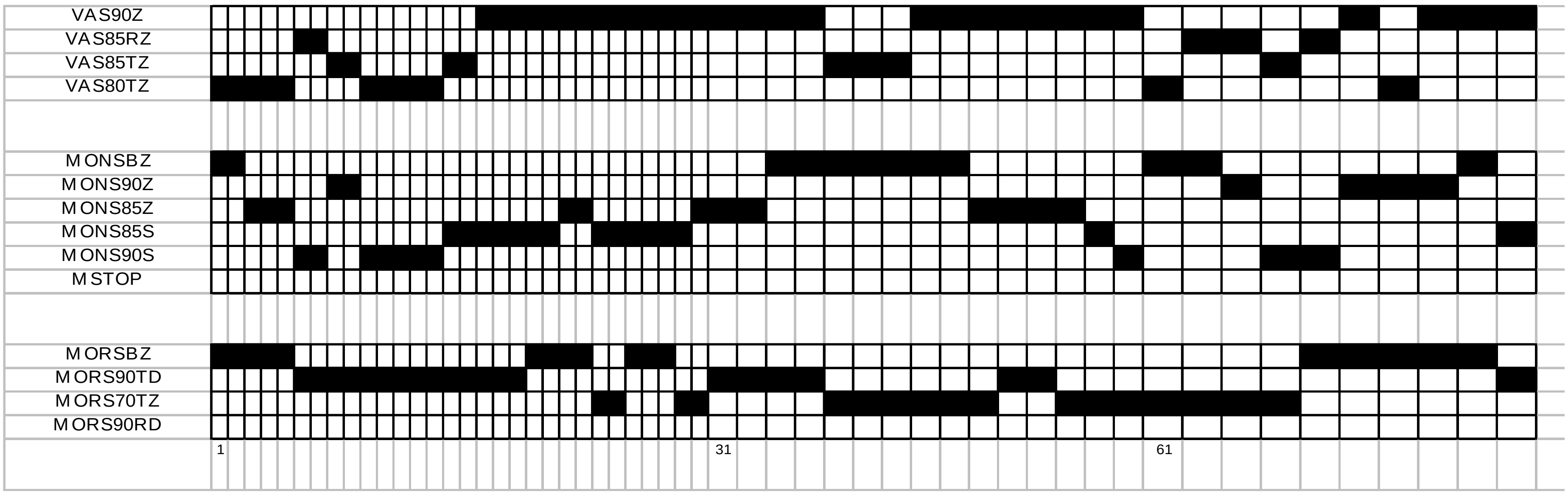

Terminals
Terminaler
Folla (4)
Svenska (17)
Svenska, Tofte (2)
Tofte (7)
SKOGN
HONEFOSS
HALDEN
INVERNESS
ABERDEEN
CORPACH
DUNDEE
GRANGEMOUTH
SUNDERLAND
KIEL
HULL
GRIMSBY
EMDEN
STETTIN
BREMEN
LOWESTOFT
NORTHFLEET
TERNEUZEN
WESEL
CHATHAM
ANTWERPEN
GHENT
BOULOGNE
BASEL
LA PALLICE
GENUA
PASAJES
AVEIRO
GANDIA

Production planning

Supply chain structure



Production plans



Home care operations

Headlines in Swedish newspapers

- *“Anna, 89, found dead after missed home care services.”*
- *“My mother met 57 different persons from the home care services during the last two months.”*
- *“Sick leave among staff members above 30 percent in the elderly care.”*

Quality and Efficiency in Home Care

A background

- City of Stockholm: 2 million in the region and 794,000 in town
- Long tradition in providing Social Care “Cutting edge” methods and alternatives in Home Care
- Growing number of elderly citizens with rising costs
- Great need for better and more effective organisation

THE CITY OF STOCKHOLM

Daily planning problem

- *Address (location)*
- *Gender*
- *Language*

- *Service (medical etc..)*
- *Care time*
- *Time windows*

- *Availability*
- *Working hours*
- *Competence/ skills*

Sudoku

Sudoku

- Given initial data
- Fill digits 1-9 into boxes such that
- Every digit 1-9 appears in every row, column, and 3x3 box

1					6	3		8
		2	3				9	
						7	1	6
7		8	9	4				2
		4				9		
9				2	5	1		4
6	2	9						
	4				7	6		
5		7	6					3

Modeling

Formulating the right model is crucial in integer programming.

Laurence Wolsey

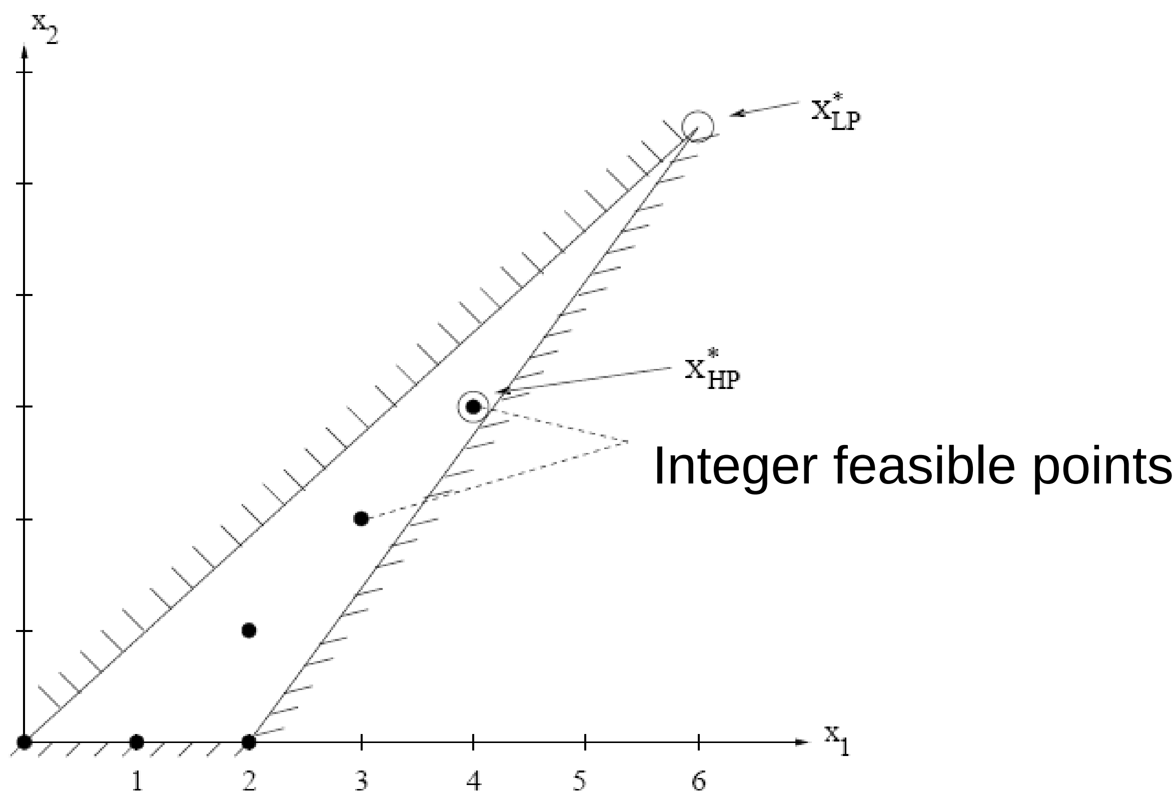
An *integer programming problem* is a problem where one or several variables are restricted to integer values. It is more correct to say that we have discrete variables i.e. they can only take a set of discrete values. Examples on discrete variables are:

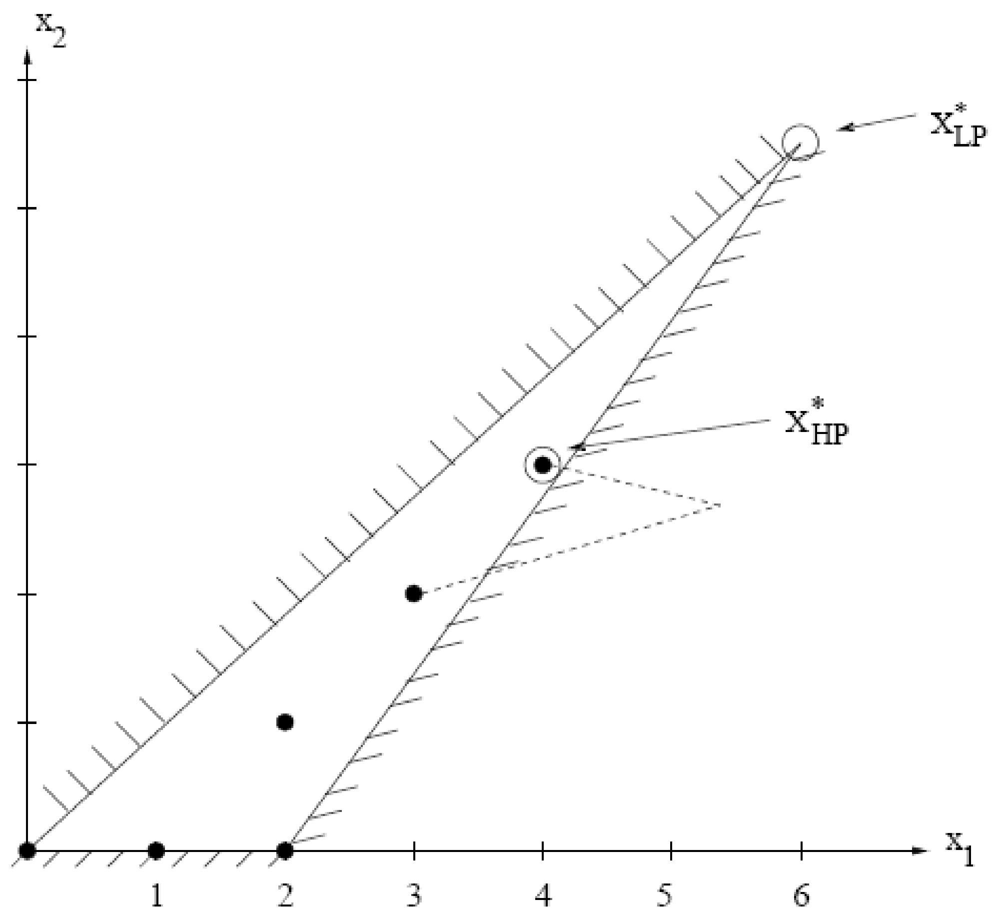
$$x_j \in \{0, 1, 2, 3, 4, 5, 6\}$$

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

$$x_j \in \{0, \tfrac{1}{4}, \tfrac{2}{4}, \tfrac{3}{4}, 1\}$$

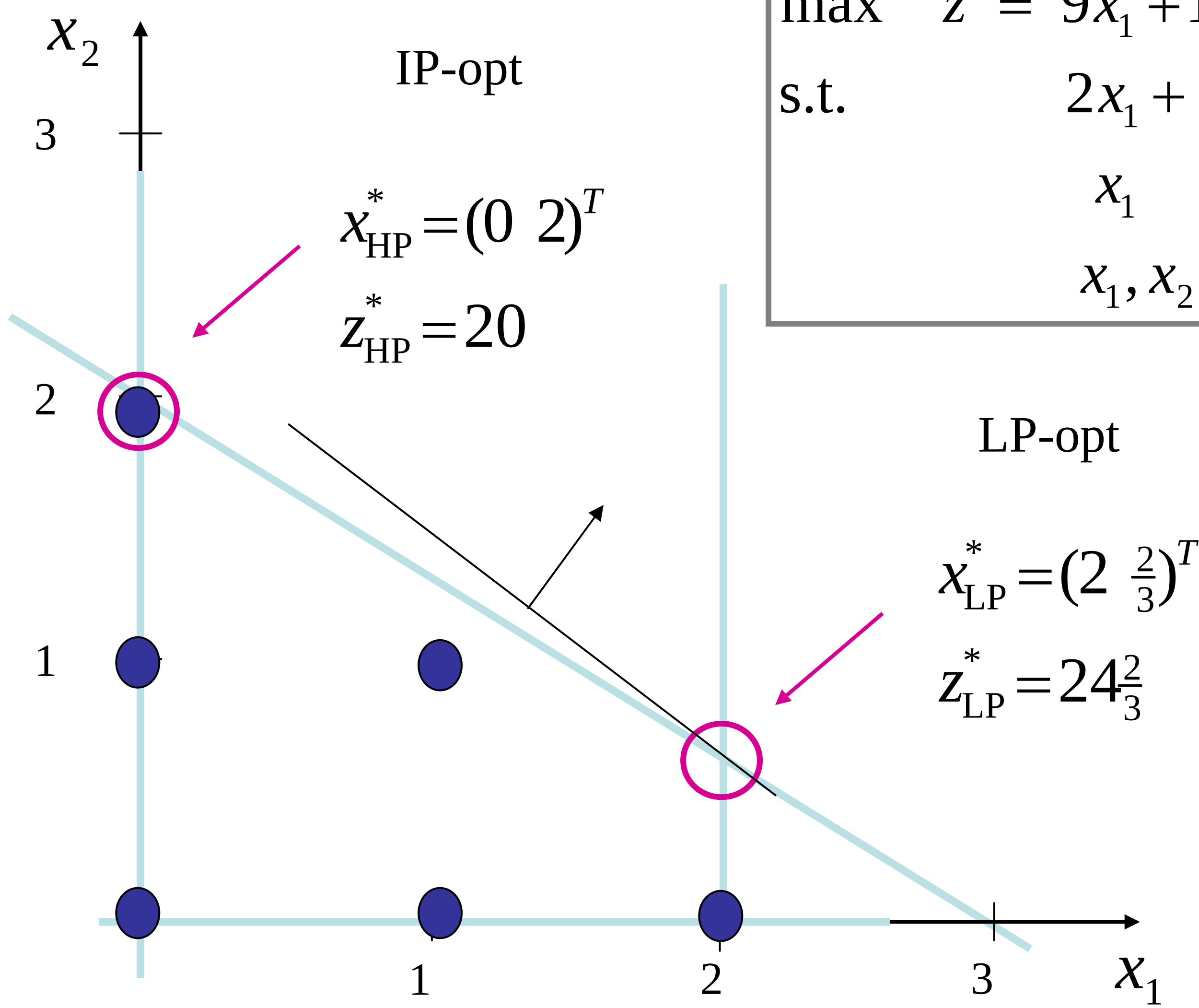
$$
\begin{array}{llrcrcl}
\max & z = & x_1 + x_2 & & & & \\
\text{s.t.} & & 11x_1 & - & 8x_2 & \leq & 22 \\
& & 11x_1 & - & 12x_2 & \geq & 0 \\
& & & & x_1, x_2 & \geq & 0, \quad \text{integer}
\end{array}
$$





The optimal solution is $\mathbf{x}^*_{HP} = (4\ 3)^T$ with $z^*_{HP} = 7$.
If we remove the requirement on integer solution we get the *LP-relaxation*.
The optimal LP-relaxation is $\mathbf{x}^*_{LP} = (6\ 5.5)^T$ with $z^*_{LP} = 11.5$.

$$
\begin{aligned}
\max \quad & z = 9x_1 + 10x_2 \\
\text{s.t.} \quad & 2x_1 + 3x_2 \le 6 \\
& x_1 \le 2 \\
& x_1, x_2 \ge 0, \ \text{integer}
\end{aligned}
$$



$z^*_{LP} \ge z^*_{HP}$ (max problem)

General IP

$$\max\,(\min)\;\; z \;=\; \sum_{j=1}^{n} c_j x_j$$

s.t.

$$\sum_{j=1}^{n} a_{ij} x_j \;\le\; b_i,\; i = 1,..,m$$

$$x_j \ge 0,\;\; j = 1,..,n,\;\; \text{integer}$$

All variables integer ⇨ Pure IP

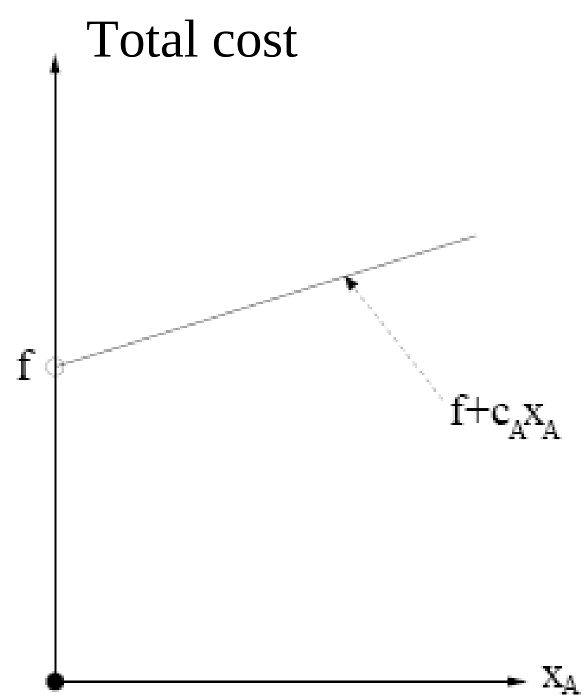
Some variables integer ⇨ Mixed IP

Binary variables (0 or 1) ⇨ 0/1 - problem

Example (fixed cost)

Production of a product A is done in a machine where the direct cost is proportional to the amount produced. At the start there is a need to configure the machine and this takes a given time and has a fixed cost. The total cost is hence a combination of a fixed cost f and a moving cost c_A for each unit. The cost is 0 *in case of no production. Suppose that x_A denote the number of A produced. The total cost is then*

$$z = \begin{cases} f + cx_A, \text{ if } x_A > 0 \\ 0, \text{ otherwise} \end{cases}$$



$$
\begin{aligned}
z = fy + c_A x_A & \\
x_A &\leq My \\
x_A &\geq 0 \\
y &\in \{0,1\}
\end{aligned}
$$

Example **(non-convex area)**

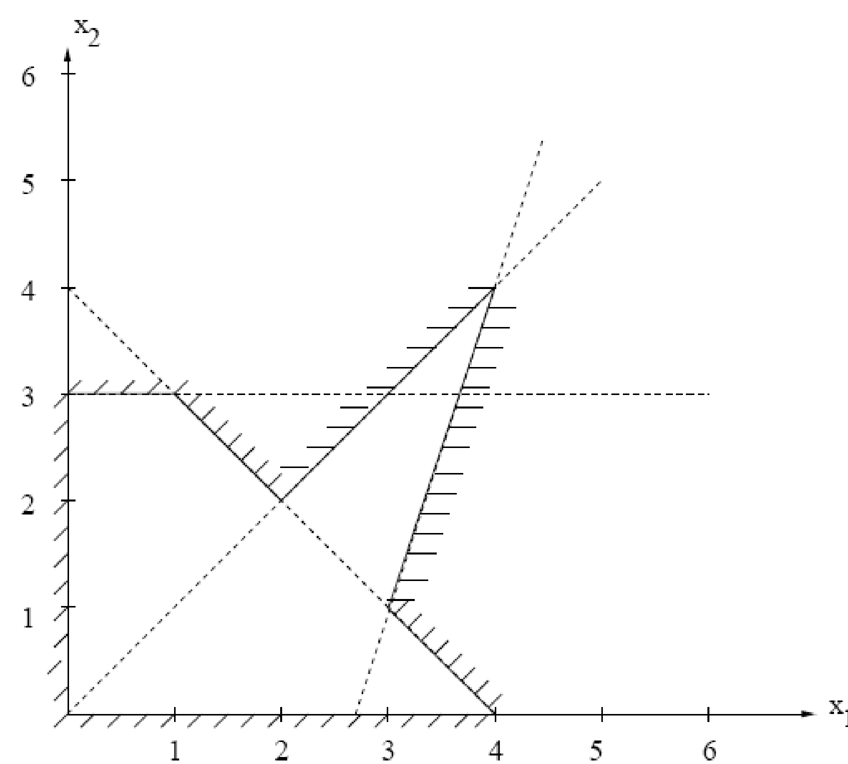
Suppose we have a problem where the feasible points are defined as laying in area 1 or area 2 (or alternatively in both). The areas are defined as

Area 1:

$$\begin{aligned} x_2 &\leq 3 \\ x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$

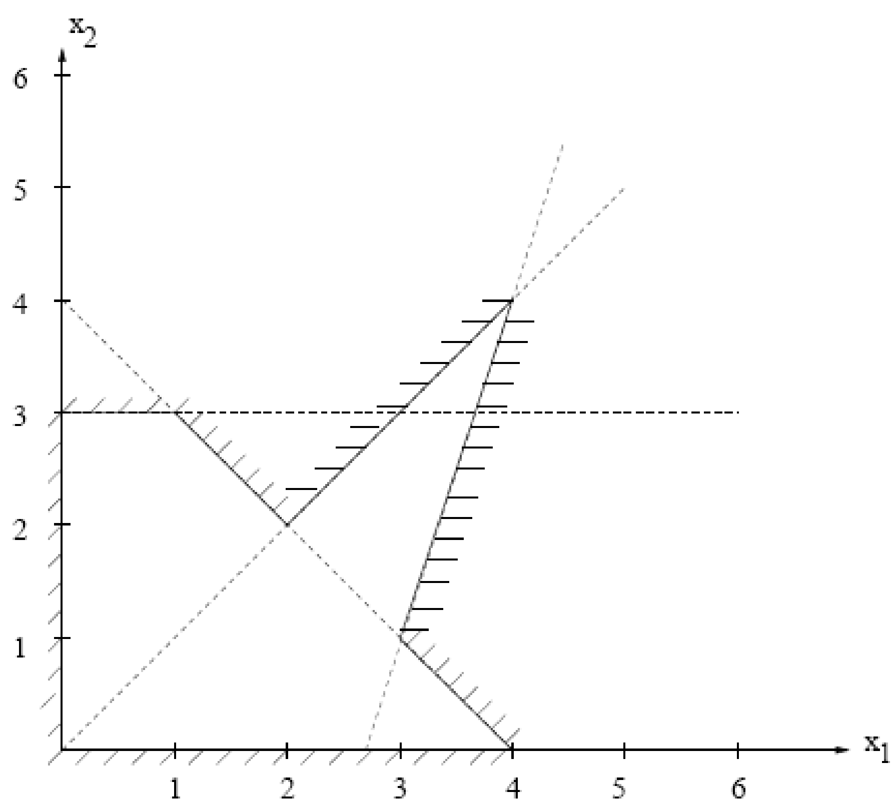
Area 2:

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



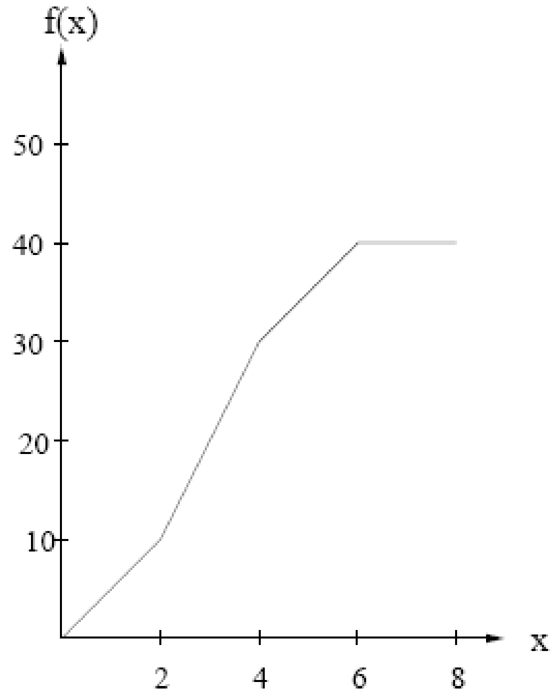
Solution: *We introduce two 0/1 variables as*

$$y_i = \begin{cases} \text{1, if the point is in area } i, \text{ i=1,2} \\ \text{0, otherwise} \end{cases}$$



$$\begin{aligned}
x_2 &\leq 3 + M(1 - y_1) \\
x_1 + x_2 &\leq 4 + M(1 - y_1) \\
-x_1 + x_2 &\leq 0 + M(1 - y_2) \\
3x_1 - x_2 &\leq 8 + M(1 - y_2) \\
y_1 + y_2 &\geq 1 \\
x_1, x_2 &\geq 0 \\
y_1, y_2 &\in \{0, 1\}
\end{aligned}$$

Describe the non-convex function below in a model.



Solution: *Firstly we identify the break points (inclusive the end points) $j = 1, \ldots, 5$ in the function and the segments between $i = 1, \ldots, 4$. We introduce the variables*

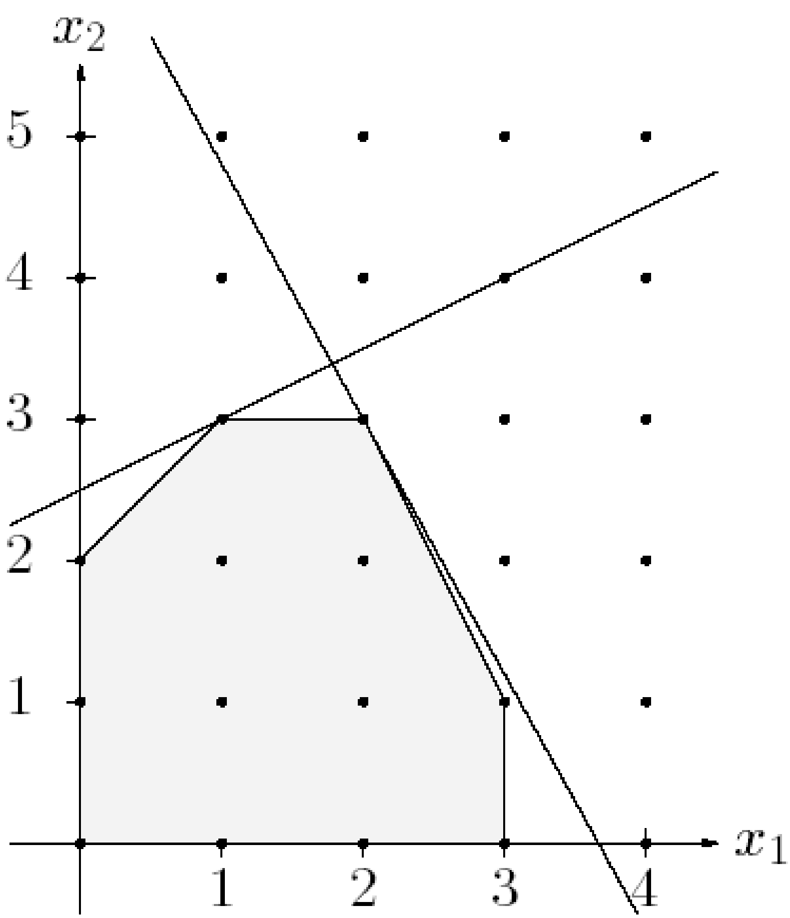
$$y_i = \begin{cases} \text{1, if segment } i \text{ is used} \\ \text{0, otherwise} \end{cases}, \quad i = 1, \ldots, 4$$

$$w_j = \text{weight for break point } j, \quad j = 1, \ldots, 5$$

The function can now be expressed with the constraints

$$
\begin{aligned}
f(x) &= 0w_1 + 10w_2 + 30w_3 + 40w_4 + 40w_5 \\
x &= 0w_1 + 2w_2 + 4w_3 + 6w_4 + 8w_5 \\
w_1 + w_2 + w_3 + w_4 + w_5 &= 1 \\
y_1 + y_2 + y_3 + y_4 &= 1 \\
w_1 &\leq y_1 \\
w_2 &\leq y_1 + y_2 \\
w_3 &\leq y_2 + y_3 \\
w_4 &\leq y_3 + y_4 \\
w_5 &\leq y_4 \\
w_1, \ldots, w_5 &\geq 0 \\
y_1, \ldots, y_4 &\in \{0, 1\}
\end{aligned}
$$

Convex hull



$$X_{HP} = \begin{cases} -x_1 + 2x_2 & \leq & 5 \\ 9x_1 + 5x_2 & \leq & 33 \\ x_1 \geq 0, x_2 \geq 0, & \text{integer} \end{cases}$$

$$\begin{aligned} -x_1 + x_2 &\leq 2 \\ x_2 &\leq 3 \\ 2x_1 + x_2 &\leq 7 \\ x_1 &\leq 3 \\ x_1,\ x_2 &\geq 0 \end{aligned}$$

Example strong formulation

$$\begin{aligned} \min\ z &= 2x_1 + 3x_2 + 6y_1 + 3y_2 \\ \text{s.t.}\quad & x_1 + x_2 \geq 5 \\ & x_1 \leq My_1 \\ & x_2 \leq My_2 \\ & x_1, x_2 \geq 0 \\ & y_1, y_2 \in \{0, 1\} \end{aligned}$$

Optimum:

$$x_1 = 5,\ \ x_2 = 0$$
$$y_1 = 1,\ \ y_2 = 0$$
$$z^* = 16$$

LP-relaxation

$$M = 100 \Rightarrow x_1 = 5,\ \ y_1 = 5/100,\ \ z^*_{LP} = 10.3$$
$$M = 10 \Rightarrow x_1 = 5,\ \ y_1 = 1/2,\ \ z^*_{LP} = 13$$
$$M = 5 \Rightarrow x_1 = 5,\ \ y_1 = 1,\ \ z^*_{LP} = 16$$

Choose M as small as possible

Note! M must be large enough. Otherwise, the Optimum is cut away.

Standard models

Standard IP models

- Knapsack problem
- Generalized Assignment Problem (GAP)
- Facility location problem
- Mixed Integer Programming (MIP)
- Set partitioning problem (SPP)

Knapsack problem

To state the general model we introduce the variables

$$x_j = \begin{cases} 1, \text{ if object } j \text{ is chosen, } j = 1, \ldots, n \\ 0, \text{ otherwise} \end{cases}$$

and the parameters

$$\begin{aligned} c_j &= \text{value if object } j \text{ is chosen} \\ a_j &= \text{resource usage if object } j \text{ is chosen} \\ b &= \text{resource limitation} \end{aligned}$$

The general model for a 0/1 knapsack problem is

$$\begin{aligned} \max \quad & z = \sum_{j=1}^{n} c_j x_j \\ \text{s.t.} \quad & \sum_{j=1}^{n} a_j x_j \leq b \\ & x_j \in \{0,1\}, \ j = 1, \ldots, n \end{aligned}$$

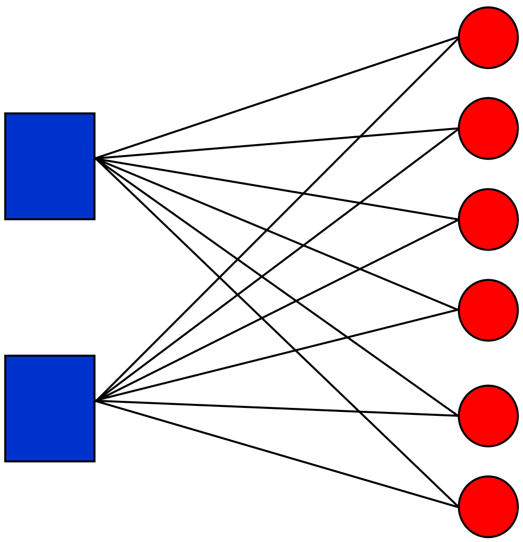
An alternative to define the variables is to use general integer variables. The only difference is that we swap $x_j \in \{0,1\}$ against $x_j \geq 0$, integer

Suppose we want to solve a 0/1 knapsack problem with 10 variables. Furthermore, suppose that it takes a computer 10^{-6} seconds to state a solution, check feasibility and then compute the objective function value. There are 2^{10} potential solutions (combinations of 0 and 1 for each variable) and the total time to test all alternatives and select the best is approximately 0.001 seconds.

If we instead study a problem with 50 variables the total approximate solution time is 37 years ($\frac{2^{50}}{365\times 24\times 3600}$) for the same computer. The solution time increases exponentially which also explains the complexity with IP problems. If we add one more variable to 51 the total solution time will double i.e. 74 years ....

... and we should have in mind that practical IP problems may have thousands even millions of integer variables!

Generalized assignment problem



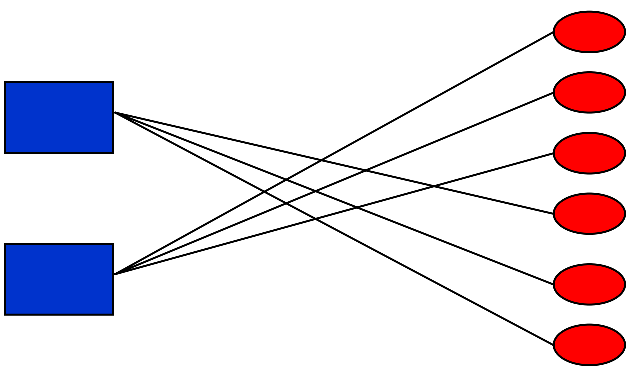
$$
\begin{aligned}
a_{ij} &= \text{usage of resource if machine } i \text{ is allocated job } j \\
c_{ij} &= \text{cost if machine } i \text{ is allocated job } j \\
b_i &= \text{capacity of machine } i
\end{aligned}
$$

$$
x_{ij} = \begin{cases} 1, & \text{if machine } i \text{ is allocated job } j \\ 0, & \text{otherwise}, \; i \in I, \;\; j \in J \end{cases}
$$

$$
\begin{aligned}
\min \quad z = & \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{j \in J} a_{ij} x_{ij} \leq b_i, && i \in I \\
& \sum_{i \in I} x_{ij} = 1, && j \in J \\
& x_{ij} \in \{0,1\}, && i \in I, \; j \in J
\end{aligned}
$$

Job	Time-M1	Time-M2	Cost-M1	Cost-M2
1	18	20	24	18
2	21	16	16	21
3	14	9	18	14
4	19	17	10	12
5	17	12	17	26
6	10	19	21	18

$$
\begin{array}{lrcl}
\min\ z = & 24x_{11} + 16x_{12} + 18x_{13} + 10x_{14} + 17x_{15} + 21x_{16} + & & \\
& 18x_{21} + 21x_{22} + 14x_{23} + 12x_{24} + 26x_{25} + 18x_{26} & & \\
\text{s.t.} & 18x_{11} + 21x_{12} + 14x_{13} + 19x_{14} + 17x_{15} + 10x_{16} & \leq & 55 \\
& 20x_{21} + 16x_{22} + 9x_{23} + 17x_{24} + 12x_{25} + 19x_{26} & \leq & 45 \\
& x_{11} + x_{21} & = & 1 \\
& x_{12} + x_{22} & = & 1 \\
& x_{13} + x_{23} & = & 1 \\
& x_{14} + x_{24} & = & 1 \\
& x_{15} + x_{25} & = & 1 \\
& x_{16} + x_{26} & = & 1 \\
& x_{ij} \in \{0,1\}, \quad i = 1,2; j = 1,\ldots,6 & &
\end{array}
$$



The optimal solution is $x^*_{14} = x^*_{15} = x^*_{16} = x^*_{21} = x^*_{22} = x^*_{23} = 1$, other variables are $x^*_{ij} = 0$, with objective function value $z^* = 101$. This means that jobs 1, 2 and 3 is done on machine 1 and jobs 4, 5 and 6 on machine 2.

In the following simplified example of GAP we assume that we just have a fxed cost f_i for using the machines that we want to minimize.

Model 1

$$
\begin{aligned}
\min\ z_1 = \ & \sum_{i=1}^{m} f_i y_i \\
& \sum_{j=1}^{n} a_{ij} x_{ij} \leq b_i y_i, \quad i = 1, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} = 1, \qquad\qquad j = 1, \ldots, n \\
& x_{ij},\ y_i \in \{0, 1\}, \quad i = 1, \ldots, m;\ j = 1, \ldots, n
\end{aligned}
$$

Model 2

$$
\begin{aligned}
\min\ z_2 = \quad & \sum_{i=1}^{m} f_i y_i \\
& \sum_{j=1}^{n} a_{ij} x_{ij} \le b_i, && i = 1, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} = 1, && j = 1, \ldots, n \\
& x_{ij} - y_i \le 0, && i = 1, \ldots, m;\ j = 1, \ldots, n \\
& x_{ij},\ y_i \in \{0, 1\}, && i = 1, \ldots, m;\ j = 1, \ldots, n
\end{aligned}
$$

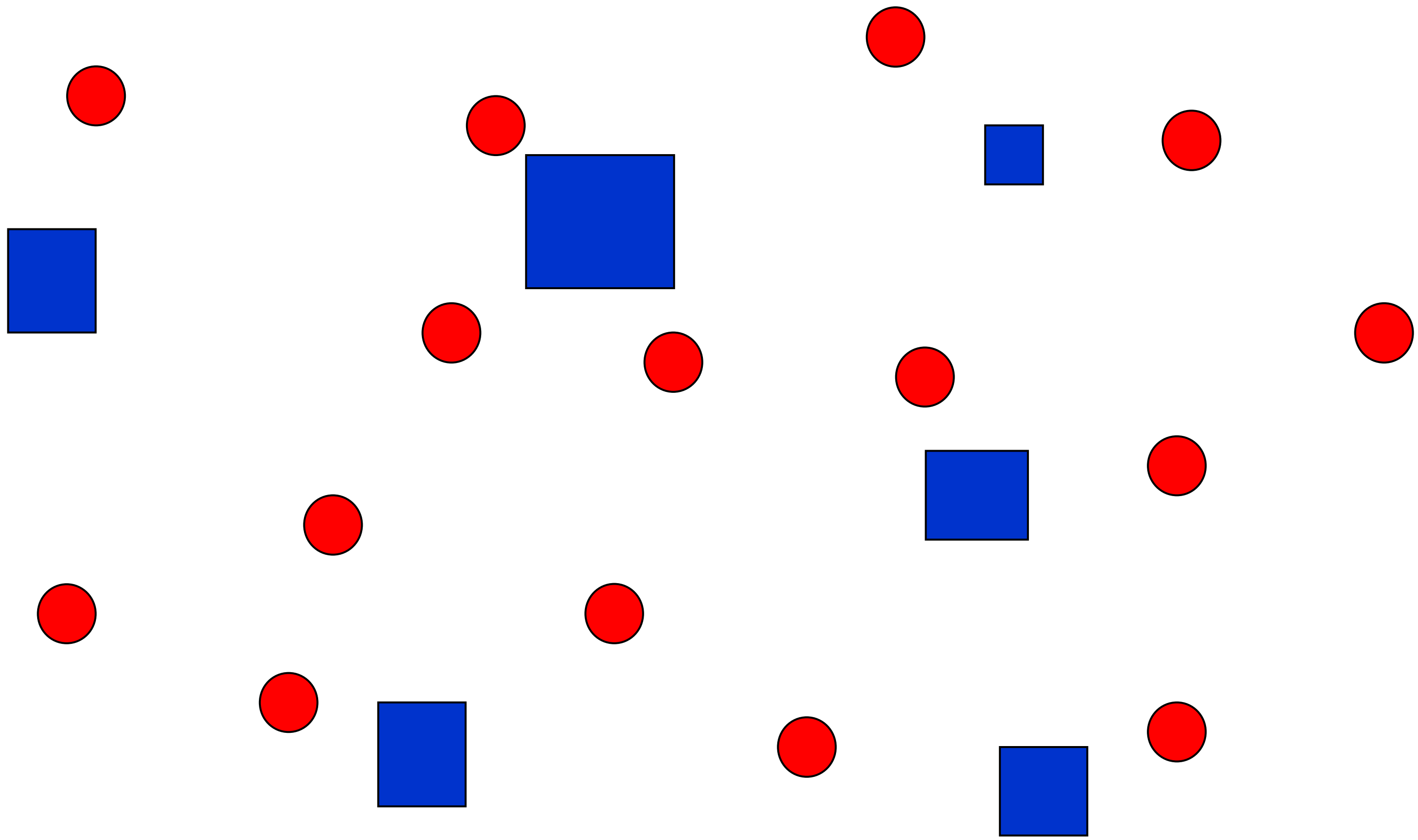
Model 3

$$
\begin{aligned}
\min\ z_3 = \quad & \sum_{i=1}^{m} f_i y_i \\
& \sum_{j=1}^{n} a_{ij} x_{ij} \leq b_i y_i, \quad && i = 1, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} = 1, && j = 1, \ldots, n \\
& x_{ij} - y_i \leq 0, && i = 1, \ldots, m;\ j = 1, \ldots, n \\
& x_{ij},\ y_i \in \{0, 1\}, && i = 1, \ldots, m;\ j = 1, \ldots, n
\end{aligned}
$$

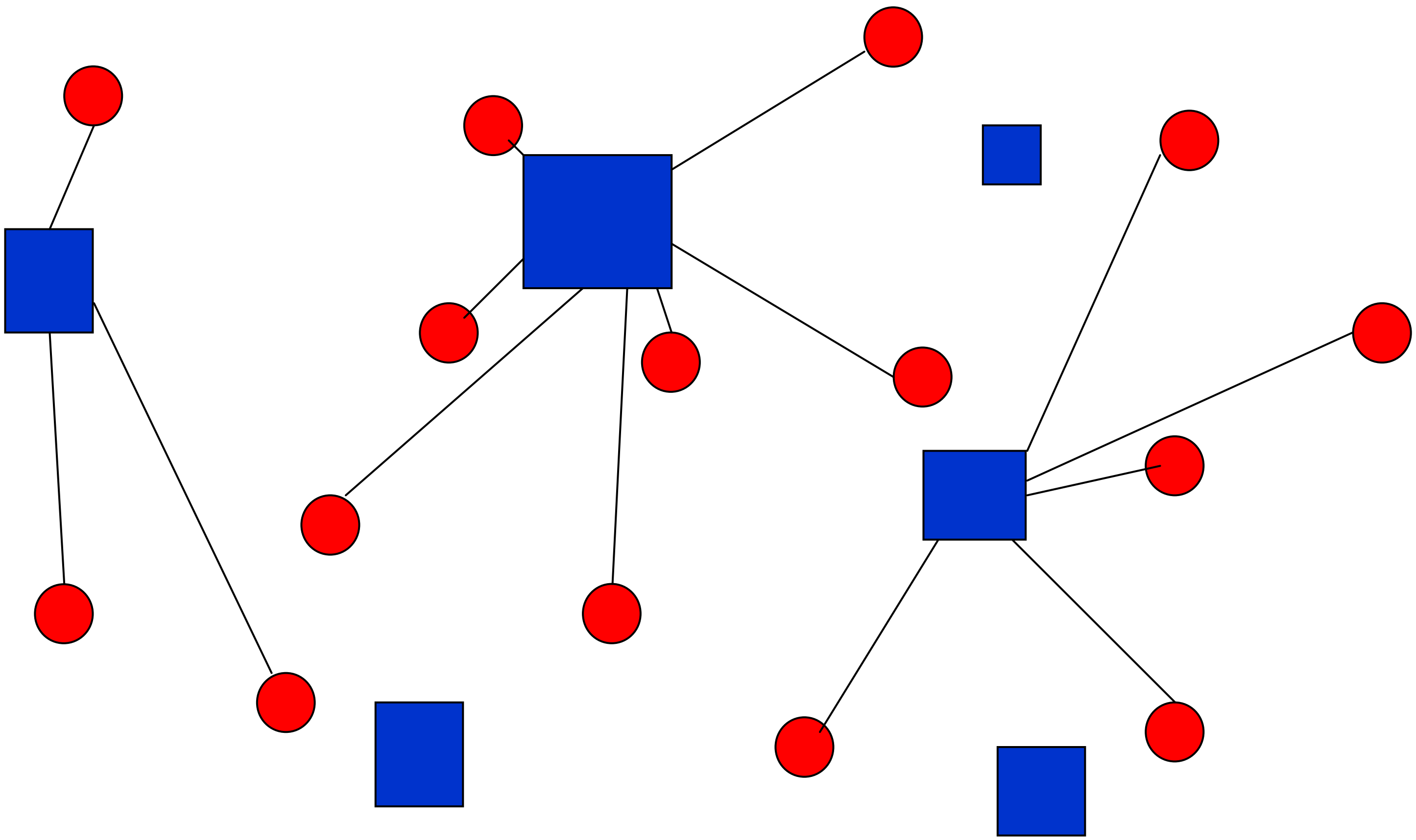
10 instances, 8 machines, 25 jobs, m=8, n=25

Instans	z_1^{LP}	z_2^{LP}	z_3^{LP}	$z_1^* = z_2^* = z_3^*$
1	241	127	293	333
2	132	68	150	192
3	174	154	271	275
4	117	83	163	207
5	221	158	310	269
6	104	89	157	177
7	160	188	388	409
8	156	129	219	291
9	123	96	171	189
10	40	41	77	102

Facility location



Facility location



We define the variables as

$$y_i = \begin{cases} 1, \text{ if facility } i \text{ is open} \\ 0, \text{ otherwise} \end{cases}$$

$$x_{ij} = \text{flow from facility } i \text{ to customer } j$$

and the model can be formulated as

$$\begin{aligned}
\min \quad z = \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i y_i \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq s_i y_i, \quad i = 1, \ldots, m \quad \text{(Supply)} \\
& \sum_{i=1}^{m} x_{ij} = d_j, \quad j = 1, \ldots, n \quad \text{(Demand)} \\
& x_{ij} \geq 0, \quad i = 1, \ldots, m; \; j = 1, \ldots, n \\
& y_i \in \{0, 1\}, \quad i = 1, \ldots, m
\end{aligned}$$

$$\begin{aligned}
\min\ z = \ & \sum_{i=1}^{m} f_i y_i + \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \\
\text{då} \quad & \sum_{i=1}^{m} x_{ij} = d_j, && j = 1,\ldots,n \\
& x_{ij} \leq M y_i, && i = 1,\ldots,m;\ j = 1,\ldots,n \\
& x_{ij} \geq 0,\ y_i \in \{0,1\}, && i = 1,\ldots,m;\ j = 1,\ldots,n
\end{aligned}$$

$$\mathbf{f} = (200\ 300\ 400)^T, \mathbf{d} = (15\ 40\ 35\ 25)^T \qquad \mathbf{c} = \begin{pmatrix} 13 & 22 & 9 & 14 \\ 12 & 10 & 11 & 13 \\ 24 & 22 & 17 & 11 \\ 13 & 14 & 25 & 17 \end{pmatrix}.$$

Different values of M gives the following results
From solving the LP-relaxation.

M	z^*_{LP}
1000	1199
500	1228
200	1315
100	1410
50	1530
40	1590
30	Infeas

Two routing models

- Network formulation
- Route based formulation with set partitioning model

Suppose we have K vehicles and that the capacity for each is b. We let d_i denote demand at customer i and c_{ij} the cost to travel between customer i and customer j ($i = 0$ is the depot).

$$
\begin{array}{llll}
\min \quad z = & \displaystyle\sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij}x_{ijk} & & \\
\text{s.t.} & \displaystyle\sum_{i=1}^{m} a_i y_{ik} \;\le\; b, & & k=1,\ldots,K \\
& \displaystyle\sum_{k=1}^{K} y_{ik} \;=\; K, & & i=0 \\
& \displaystyle\sum_{k=1}^{K} y_{ik} \;=\; 1, & & i=1,\ldots,m \\
& \displaystyle\sum_{i=1}^{m} x_{ijk} \;=\; y_{jk}, & & j=1,\ldots,m;\; k=1,\ldots,K \\
& \displaystyle\sum_{j=1}^{m} x_{ijk} \;=\; y_{jk}, & & i=1,\ldots,m;\; k=1,\ldots,K \\
& \displaystyle\sum_{(i,j)\in S}^{m} x_{ijk} \;\le\; \mid S \mid -1, & & k, 2 \le\mid S \mid\le m \\
& x_{ijk} \;\in\; \{0,1\}, & & i=1,\ldots,m;\; j=1,\ldots,m; k=1,\ldots,K \\
& y_{ik} \;\in\; \{0,1\}, & & i=1,\ldots,m; k=1,\ldots,K
\end{array}
$$

Set partitioning problem

We introduce notation

$$a_{ij} = \begin{cases} 1, & \text{if element } i \text{ is inluded in alternative } j \\ 0, & \text{otherwise}, \quad i = 1, \ldots, m, \; j = 1, \ldots, n \end{cases}$$

$$c_j = \text{cost for alternative } j, j = 1, \ldots, n$$

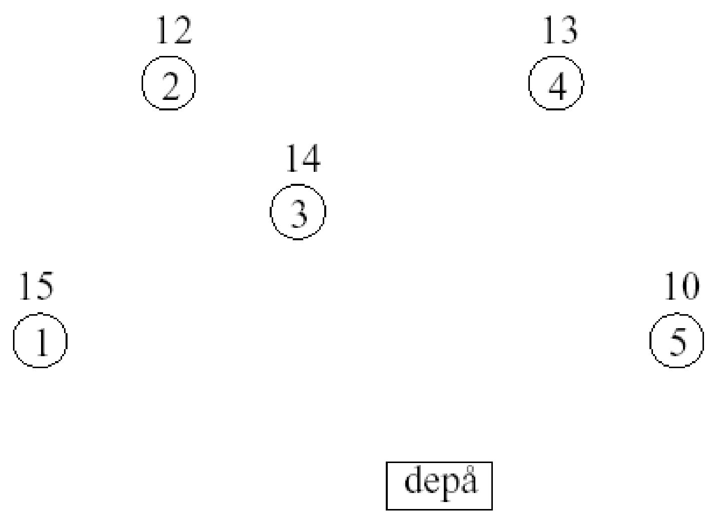
and the definition of variables becomes

$$x_j = \begin{cases} 1, & \text{if alternative } j \text{ is used} \\ 0, & \text{otherwise}, \quad j = 1, \ldots, n \end{cases}$$

The model can be formulated as

$$\begin{aligned} \min \quad & z = \sum_{j=1}^{n} c_j x_j \\ \text{då} \quad & \sum_{j=1}^{n} a_{ij} x_j = 1, \quad i = 1, \ldots, m \\ & x_j \in \{0, 1\}, \quad j = 1, \ldots, n \end{aligned}$$

A transporter has two vehicles and should deliver two five customers. The capacities are 40 and 35. The planner has decided that one vehicle will deliver to three customers and one to two customers. The customer demand is 15, 12, 14, 13 and 10.



$$\min \quad z = [117\ 123\ 112\ 119\ 98\ 104\ 87\ 90\ 76\ 110\ 104\ 71\ 99\ 105\ 78\ 78\ 74]^T \mathbf{x}$$

$$\text{s.t.} \quad \begin{bmatrix} 1&1&1&1&1&1&1&&&&&&&&&& \\ &&&&&&&1&1&1&1&1&1&1&1&1&1 \\ 1&1&1&1&&&&1&1&1&1&&&&&& \\ 1&1&&&1&1&&1&&&&1&1&1&&& \\ &&1&&1&1&1&&1&&&1&&&1&1& \\ 1&&&1&1&&1&&&1&&&1&&1&&1 \\ &1&1&1&&1&1&&&&1&&&1&&1&1 \end{bmatrix} \mathbf{x} \;=\; \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{matrix} \text{(Vehicle 1)} \\ \text{(Vehicle 2)} \\ \text{(Customer 1)} \\ \text{(Customer 2)} \\ \text{(Customer 3)} \\ \text{(Customer 4)} \\ \text{(Customer 5)} \end{matrix}$$

Optimal solution: $x_7 = x_8 = 1$, others $x_j = 0$ and $z^* = 177$.

Comparison (10 vehicles, 50 customer)

- Network
  - 25,500 variables
  - 1070 constraints
  - ? constraints (subtours)
- + and -:
  + simple formulation
  + all routes included
- - solvable?
  - difficult with capacity constraints

- Set-partitioning
  - ? Variables
  - 60 constraints
- + and -:
  + simple to generate routes (with capacity)
- + solvable
- - not all routes

Methods

Implicit enumeration In this class of methods we find Branch and Bound (B&B) methods. Here enumeration with optimistic and pessimistic bounds are used to limit the search.

Relaxation- and decomposition These methods use relaxations of the model. frequently used relaxations are LP-relaxation and Lagrangian relaxation.

Decomposition methods splits the master problem into a number of sub-problems. Each subproblem is relatively easy to solve as compared to the original problem. Generation of variables are often called column generation.

Cutting plane methods These methods are based on solving a sequence of LP problems where additional constraints are added. This means that the convex hull of the integer points are generated.

Heuristics Heuristics are methods that are based on simple rules and/or optimization methodology. There is no guarantee of optimal solutions. However, the solutions are often found in a short solution time.

It is possible to state optimality conditions (Karush-Kuhn-Tucker conditions) for LP problems and nonlinear problems. These conditions, together with convexity analysis, can be used to identify and prove that a solution is a local or global optimal solution. For IP problems there are no corresponding optimality conditions. we need to use other theories and techniques to show that a solution is optimal or not.

Usually, iterative methods are developed that finds optimistic bounds on the optimal objective function value z^* and pessimistic estimations. For minimization problems the optimistic bounds are lower bounds, $\underline{z}$ (LBD), and the pessimistic bounds an upper bound $\bar{z}$ (UBD).

If we assume a minimization problem and generate a sequence of optimistic bounds

$$\underline{z}^1 < \underline{z}^2 < \underline{z}^3 < \ldots < \underline{z} \leq z^*$$

and a sequence of pessimistic bounds

$$\bar{z}^1 > \bar{z}^2 > \bar{z}^3 > \ldots > \bar{z} \geq z^*$$

we can stop the solution approach when the optimal objective function value is within a small interval, i.e. $\bar{z} - \underline{z} < \epsilon$.

Consider the IP problem [IP]

$$
\begin{array}{llrcll}
\min & z_{IP} = & \mathbf{c}^T\mathbf{x} & & & \\
\text{s.t.} & & \mathbf{A}\mathbf{x} & \leq & \mathbf{b} & (1) \\
 & & \mathbf{x} & \in & X & (2) \\
 & & \mathbf{x} & \multicolumn{2}{l}{\text{integer}} & (3)
\end{array}
$$

where we have divided the constraints into two sets. One set is the complicating constraints (1), and the second (2) are simple constraints. Examples on simple constraints are for example lower and upper bounds on variables. Constraints (3) are the integer requirements on the variables.

Pessimistic bounds

Each feasible solution $\bar{\mathbf{x}}$ to the problem IP provides an upper bound and is called pessimistic estimation of z^*_{IP}, i.e. $\mathbf{c}^T\bar{\mathbf{x}} \geq z^*_{IP}$. These bounds are also called *primal bounds*.

Optimistic bounds

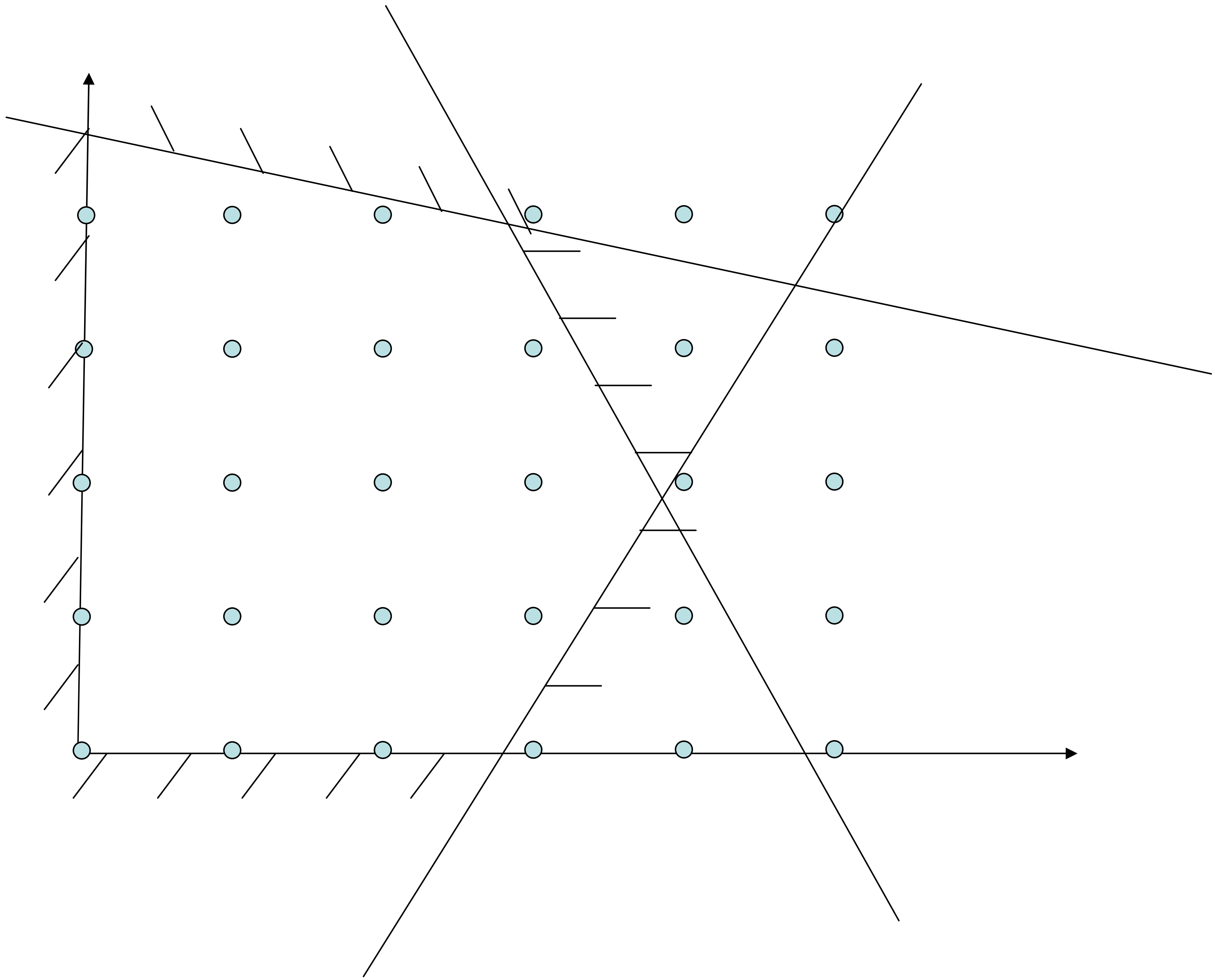
To generate lower bounds and optimistic bounds to z^*_{IP} is not equally standard. Instead there is a need to use different methods. Each is based on some form of *relaxation* which is some kind of simplification of the original problem. It is possible to for example

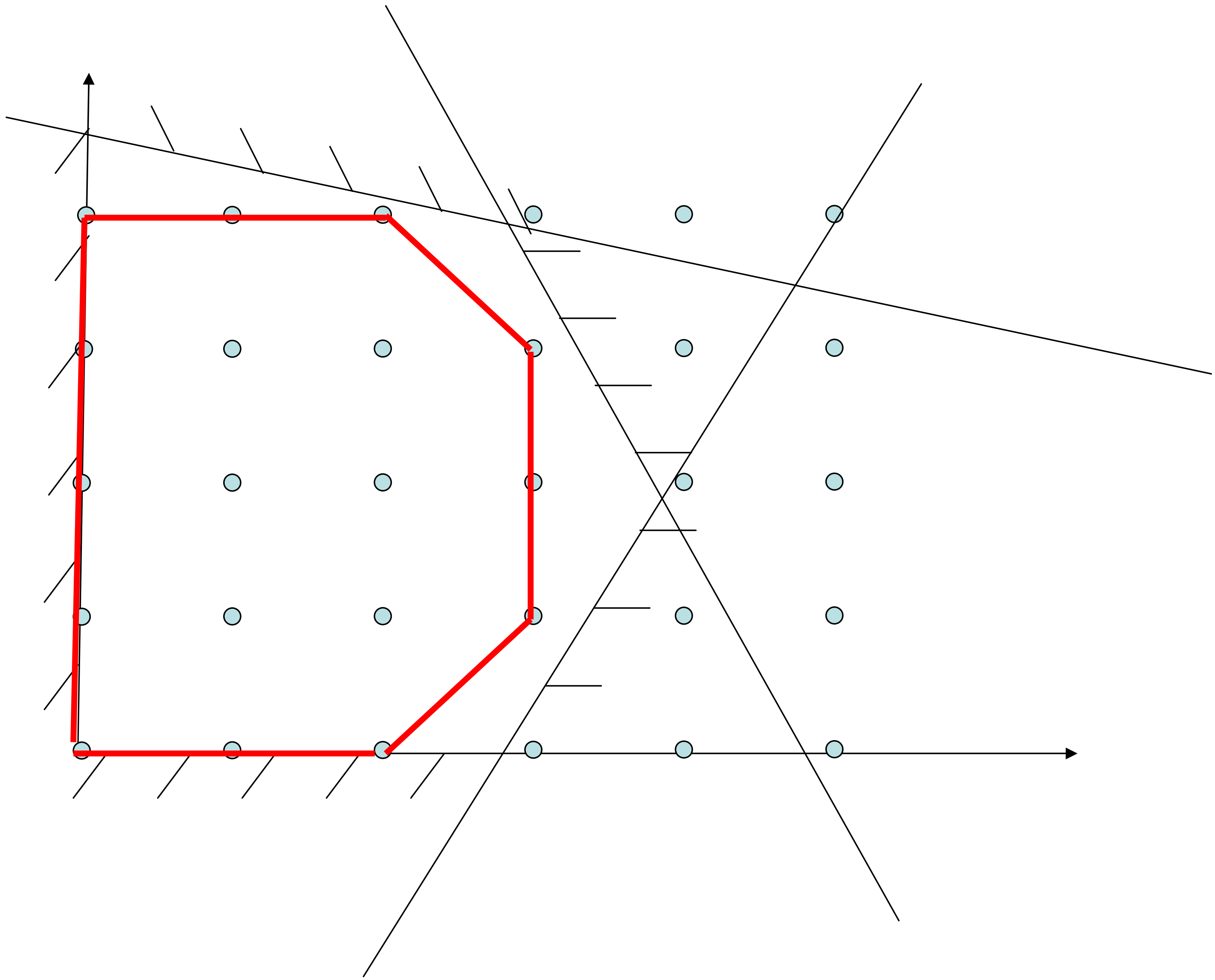
- Remove the integer constraints (3) and solve the LP relaxation.
- Make the feasible area larger by removing one or several of the constraints. We can, for example, solve problem IP without the complicating constraints (1)
- Make a *Lagrangian relaxation*, which remove a set of constraints, for example, the constraint set (1) but at the same time change the objective function buy introducing terms based on Lagrangian multipliers.

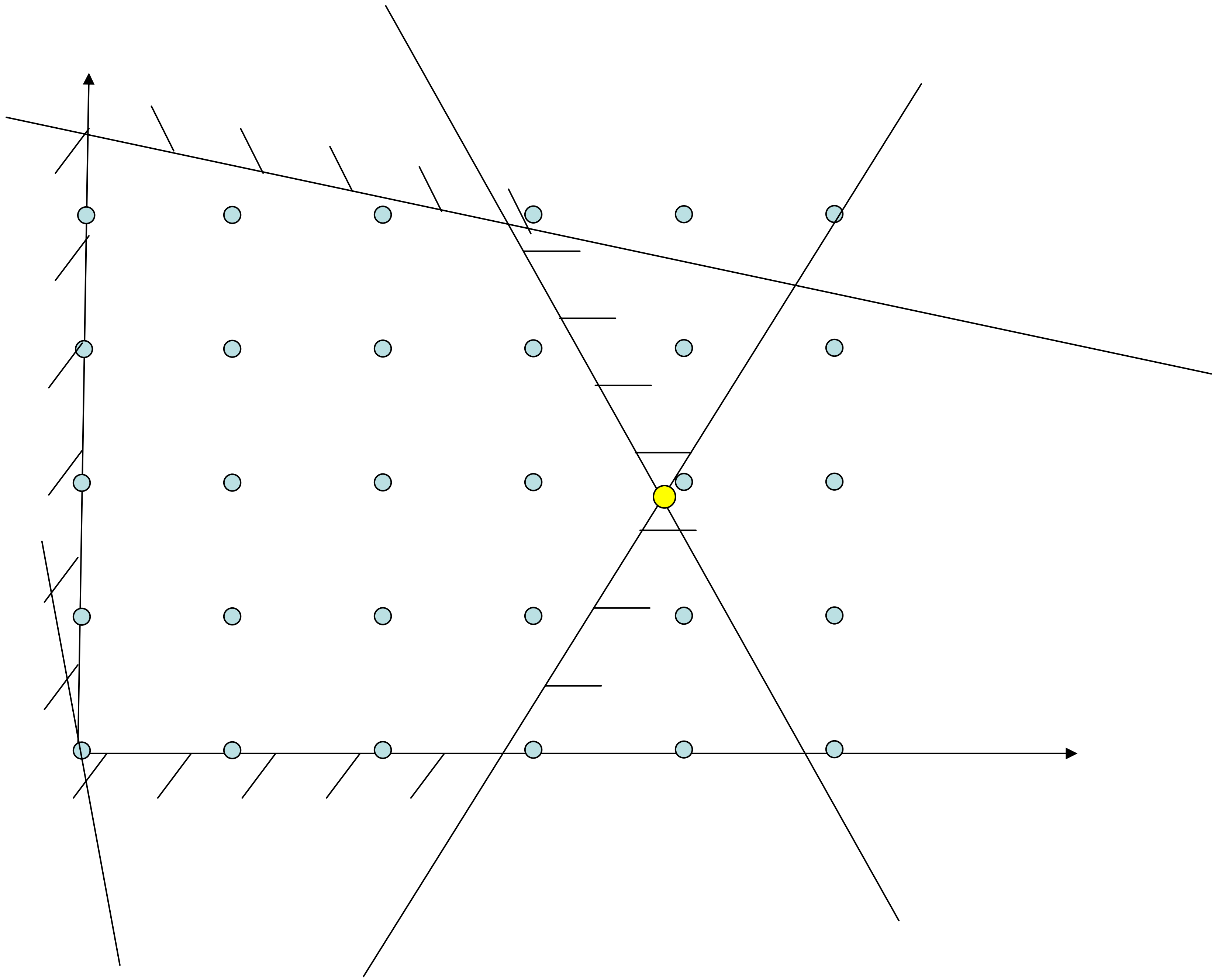
The solution to a relaxation provides important information about the IP problem. Let R denote a relaxation of IP. Moreover, let $\mathbf{x}_{IP}^*$ and $\mathbf{x}_R^*$ denote the optimal solutions to the IP problem and the relaxation , respectively. Also let z_{IP}^* and z_R^* denote the corresponding optimal objective function values.

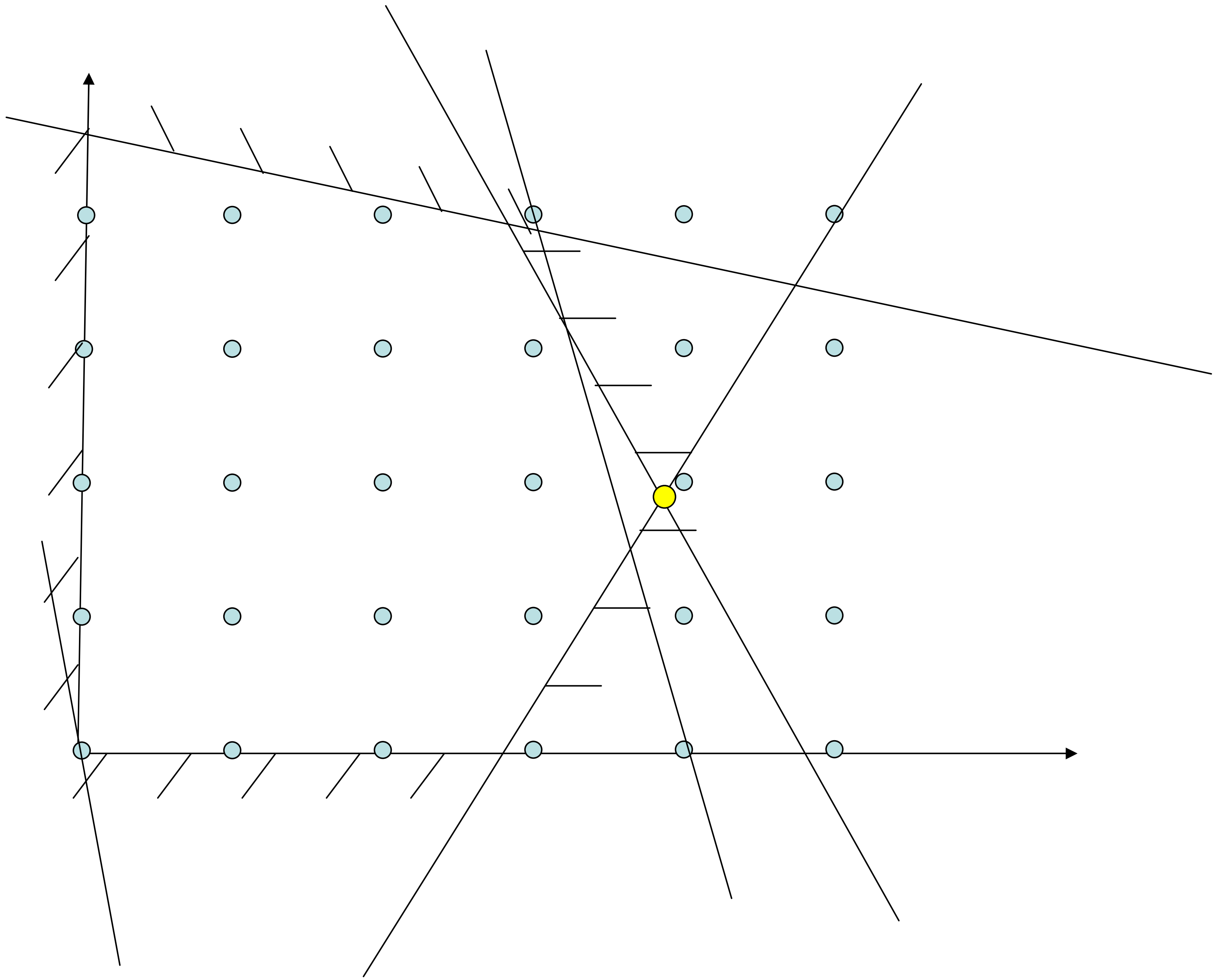
- A relaxation always provide a better (or equally good) objective function value i.e. $z_R^* \leq z_{HP}^*$.
- If R has no feasible solution, then problem IP has no feasible solution.
- If x_R^* is a feasible solution to IP, we have $\mathbf{x}_{IP}^* = \mathbf{x}_R^*$ and $z_{IP}^* = z_R^*$.
- Each feasible solution $\bar{\mathbf{x}}$ to IP found, for example, by modifying $\mathbf{x}_R^*$, provides a pessimistic bound of z_{IP}^*.

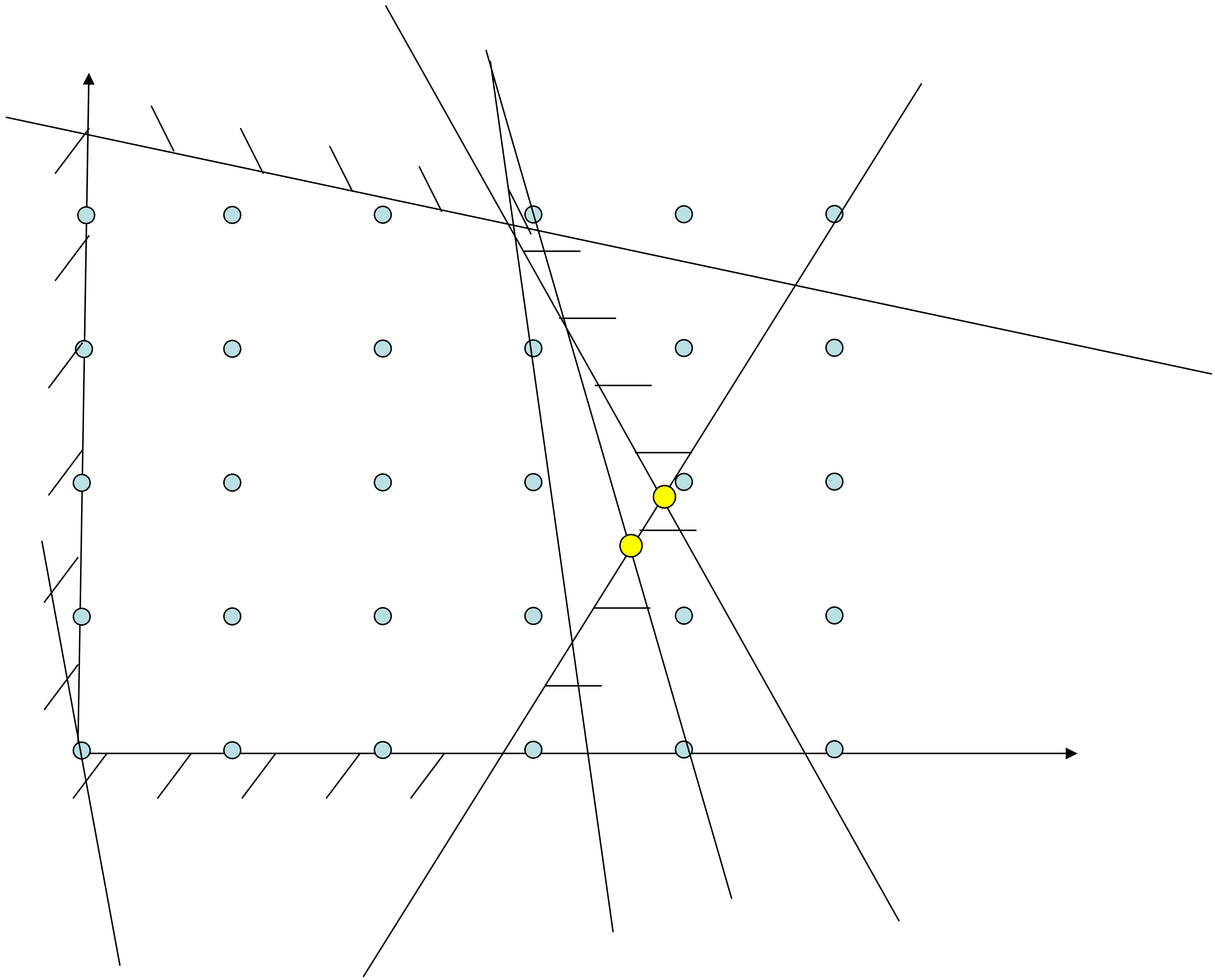
Cutting planes

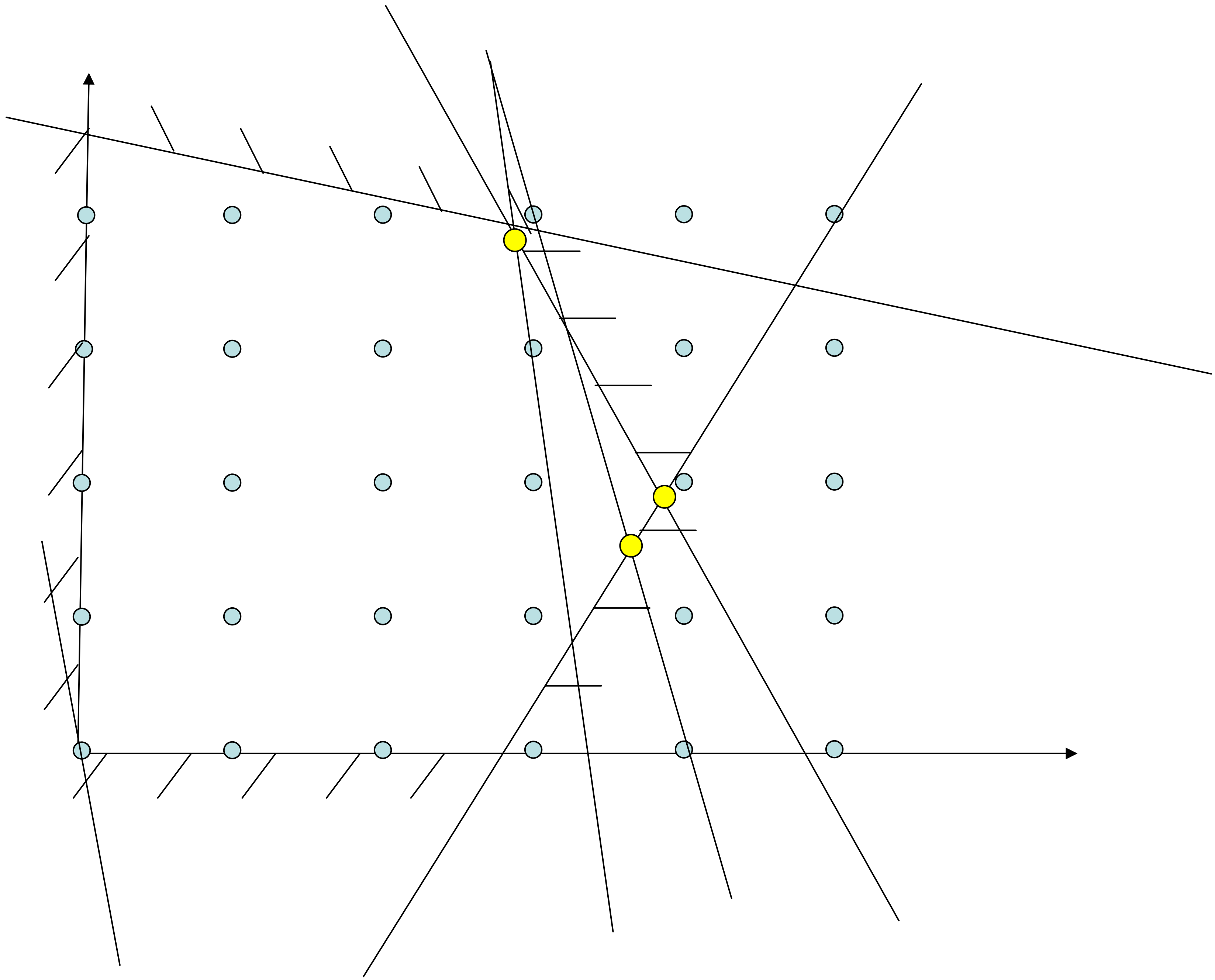


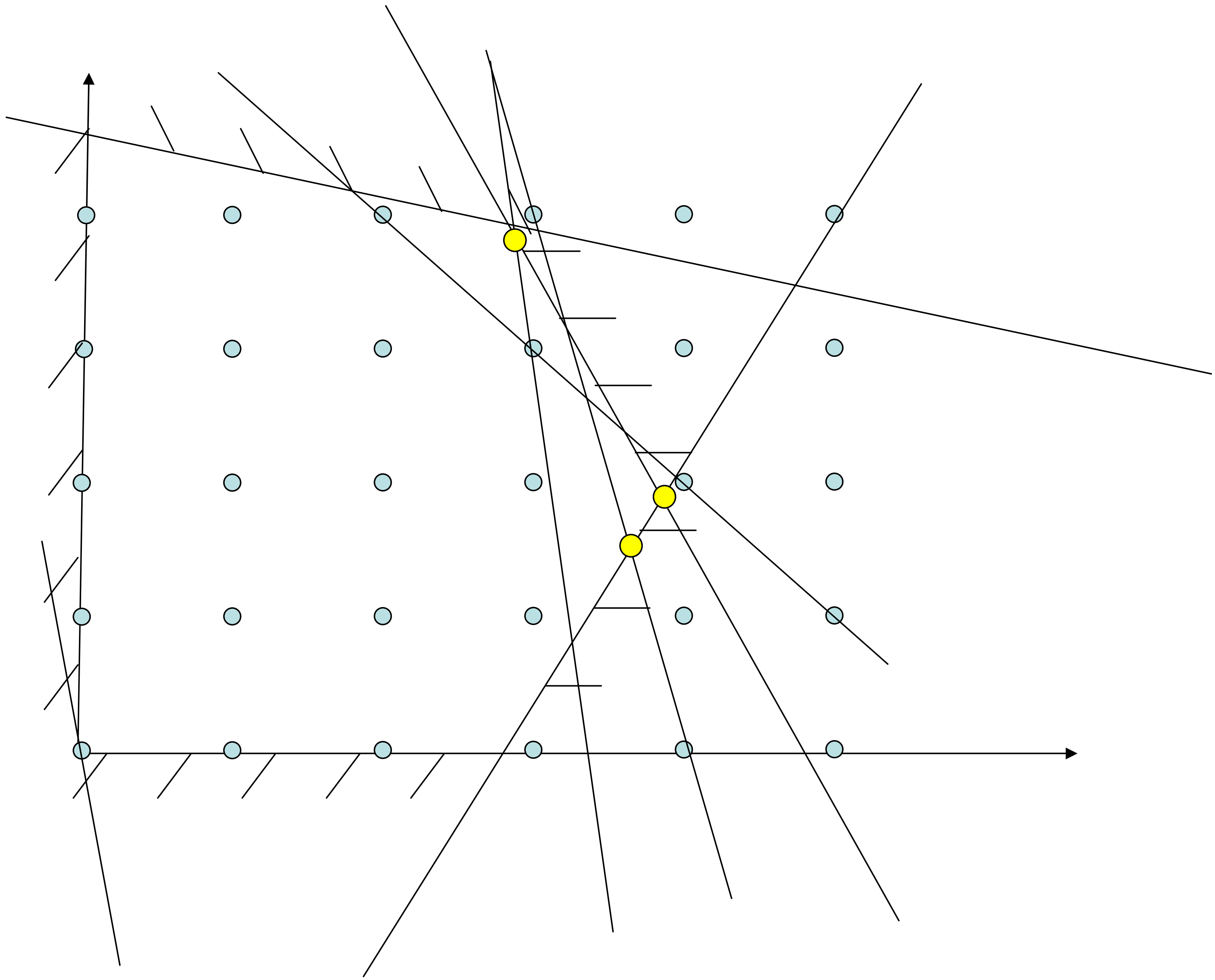


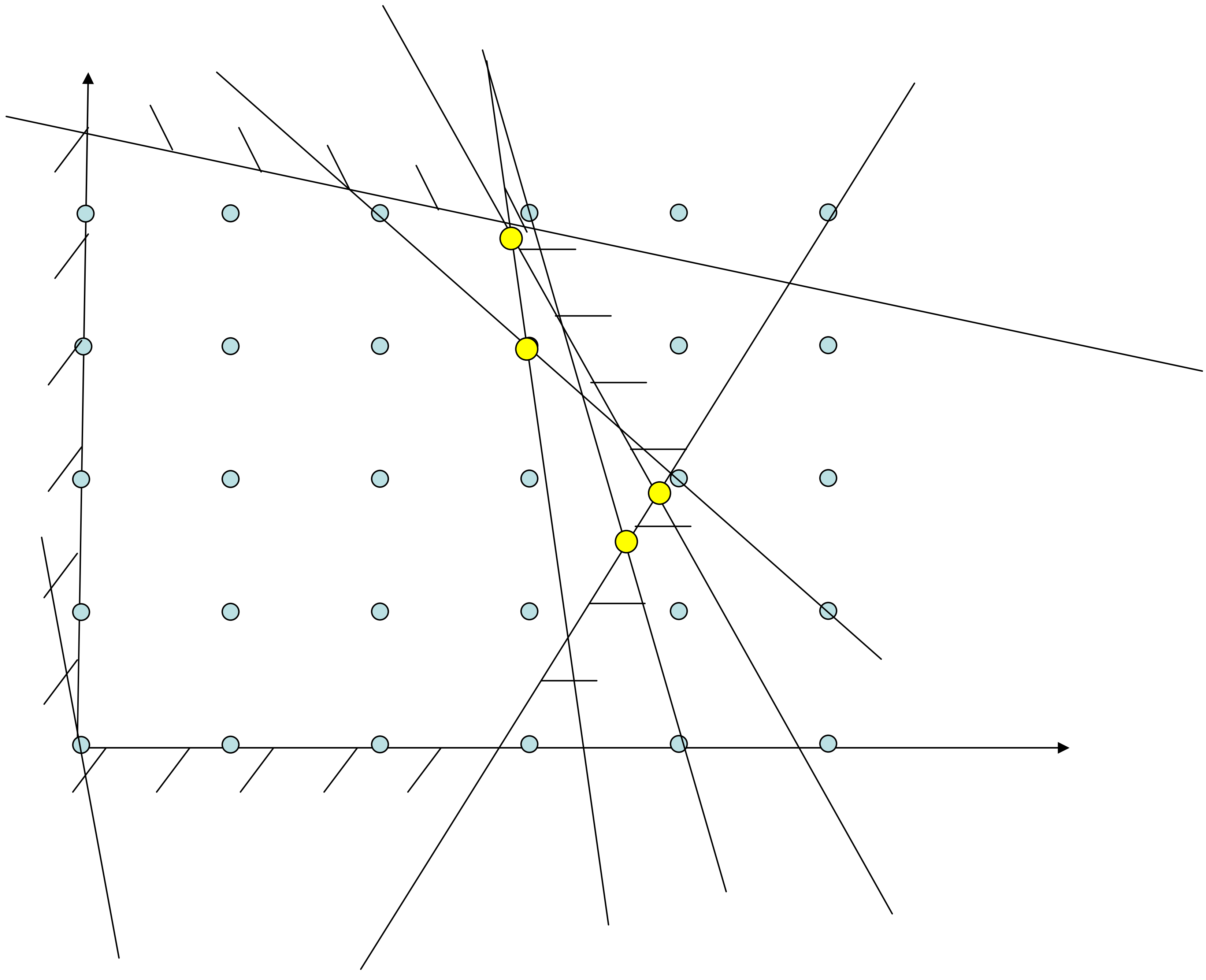


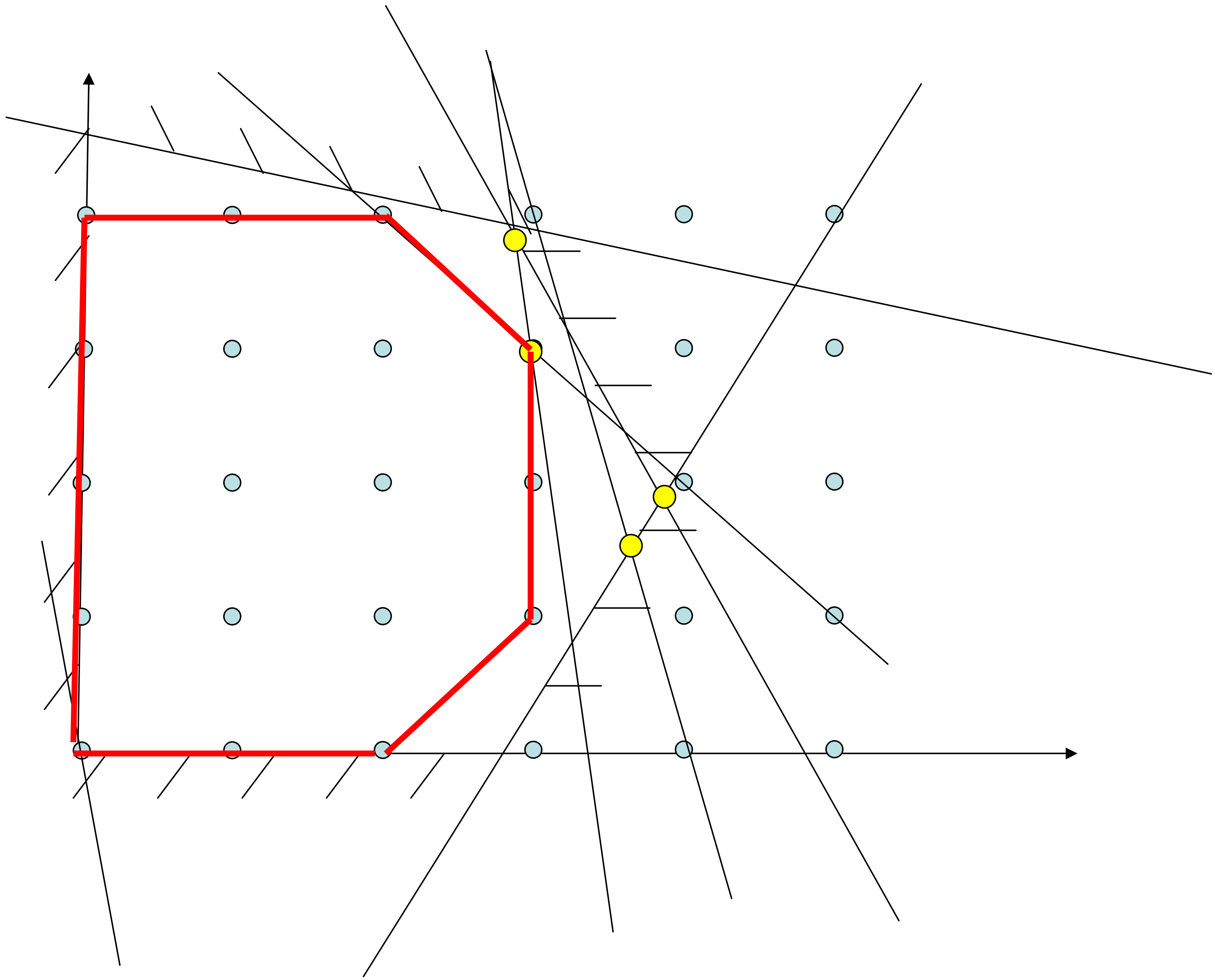












Step 1 Choose a strong formulation of the problem. Add initially generated valid inequalities if possible.

Step 2 Solve the LP-relaxation.

Step 3 If an integer solution is found $\rightarrow$ Stop, we have found the optimum slution.

Step 4 Add one or several valid inequaliteis that cut away the solution to the LP-relaxation.

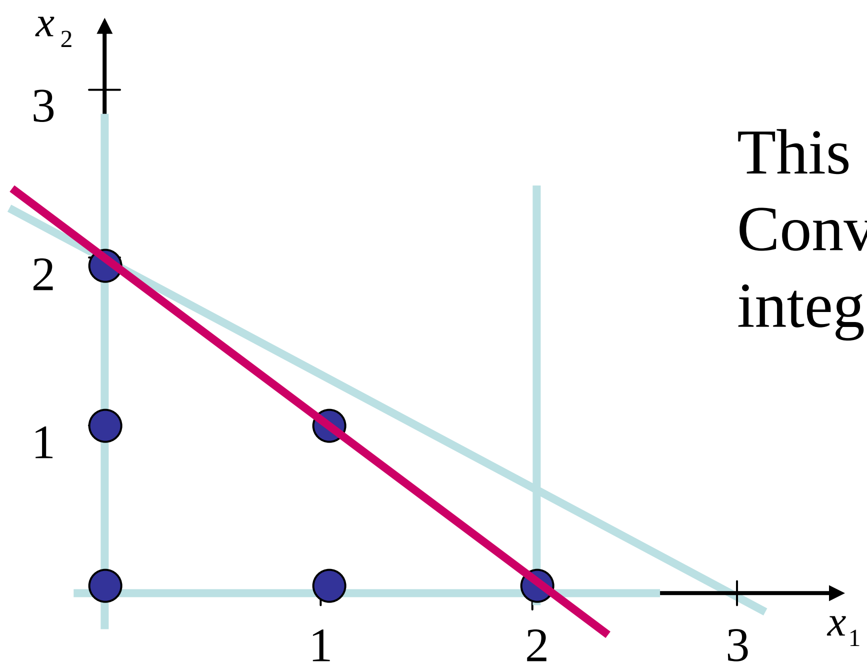
(a) based on problem specific inequalities, or

(b) based on a general cutting plane method(e.g. Gomory's method).

Step 5 Reoptimize and go to Step 3.

Example

Gomory cut: $x_1 + x_2 \leq 2$



This cut provides the Convex hull to the integer points

Example

$$\max\ z = 11x_1 + 10x_2 + 3x_3 + 4x_4 + x_5$$

$$\text{st} \qquad 5x_1 + 3x_2 + 2x_3 + 2x_4 + x_5 \leq 6$$

$$x_j \in \{0,1\},\ \forall\ j$$

Possible valid inequalities

$$x_1 + x_2 \leq 1 \quad (1) \qquad x_1 + x_3 \leq 1 \quad (2)$$

$$x_1 + x_4 \leq 1 \quad (3) \qquad x_2 + x_3 + x_4 \leq 2 \quad (4)$$

Solve LP relaxation $\Rightarrow$ $x_{LP}^{*} = \left(3/5, 1, 0, 0, 0\right)^{T},\ z = 16\,3/5$

Add (1) $\Rightarrow$ $x_{LP}^{*} = \left(0, 1, 1/2, 1, 0\right)^{T},\ z = 15\,1/2$

Add (4) $\Rightarrow$ $x_{LP}^{*} = \left(0, 1, 0, 1, 1\right)^{T},\ z = 15$

Optimum!

Branch & bound methods

Algorithm – Land-Doig-Dakins:

Step 0 Initialize the pessimistic bound $\bar{z} = +\infty$. If a feasible solution is known, $\hat{\mathbf{x}}$, update the bound with $\bar{z} = \mathbf{c}^T\hat{\mathbf{x}}$. Set $n = 0$ and $k = 0$.

Step 1 Solve The LP-relaxation of subproblem Pk. We get a solution $\mathbf{x}^{Pk}$ and objective function value z_{Pk} which provides an optimistic bound in that part of the search tree.

Step 2 If no solution is found in Pk, stop the solution and back track. Go to Step 6.

Step 3 If $z_{Pk} > \overline{z}$, then we can not find a better solution and we can terminate the search in this region and back track. Go to Step 6.

Step 4 If $\mathbf{x}^{Pk}$ satisfy the integer requirements then we can not find better solutions and we can break and back track. If $z_{Pk} < \overline{z}$ then we update the best pessimistic bound by $\overline{z} = z_{Pk}$ and let $\hat{\mathbf{x}} = \mathbf{x}^{Pk}$. Go to Step 6.

Step 5 Choose a fractional variable x_j with value $\bar{b}_j$ and create two subproblems as

$Pn+1$: Pk + constraint $x_j \leq \lfloor \bar{b}_j \rfloor$
$Pn+2$: Pk + constraint $x_j \geq \lfloor \bar{b}_j \rfloor + 1$

Let $n := n + 2$. Note that node Pk is searched.

Step 6 If all nodes are searched or if the convergence crteria are satisfied, stop. optimal solution is $\hat{\mathbf{x}}$ with objective function value $\bar{z}$.

Otherwise, choose a not searched node Pk and go to Step 1.

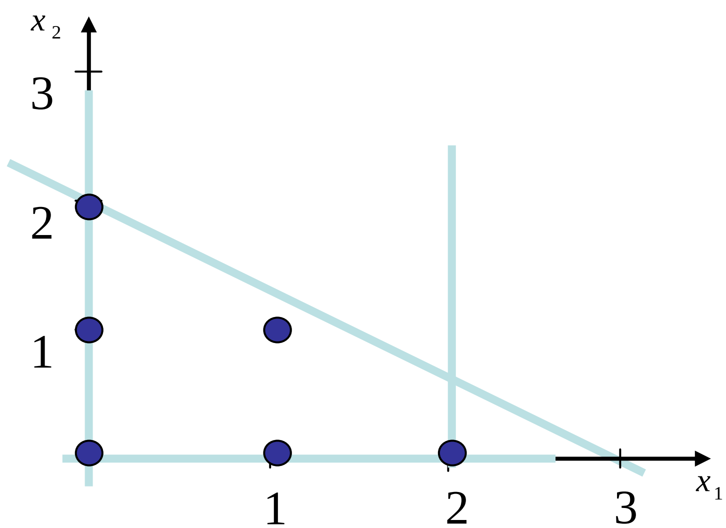
Land-Doig-Dakins

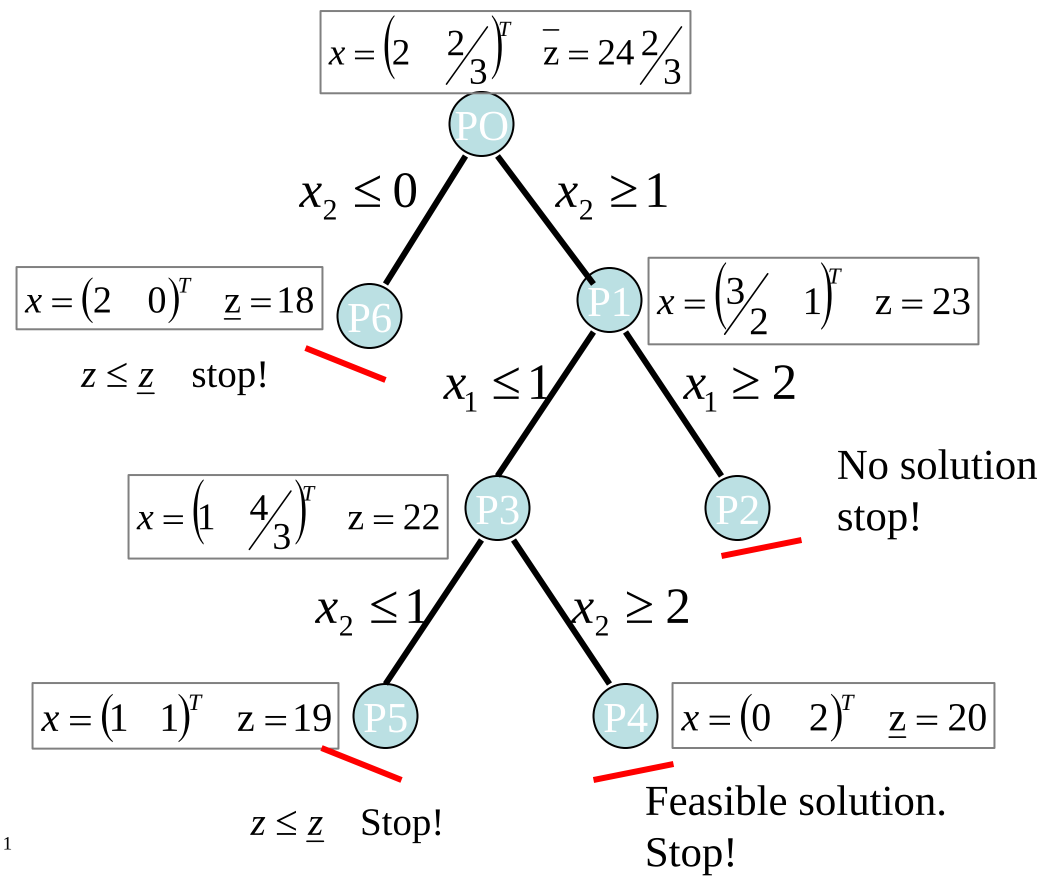
$$\max \quad z = 9x_1 + 10x_2$$

$$\text{st} \quad 2x_1 + 3x_2 \le 6$$

$$x_1 \le 2$$

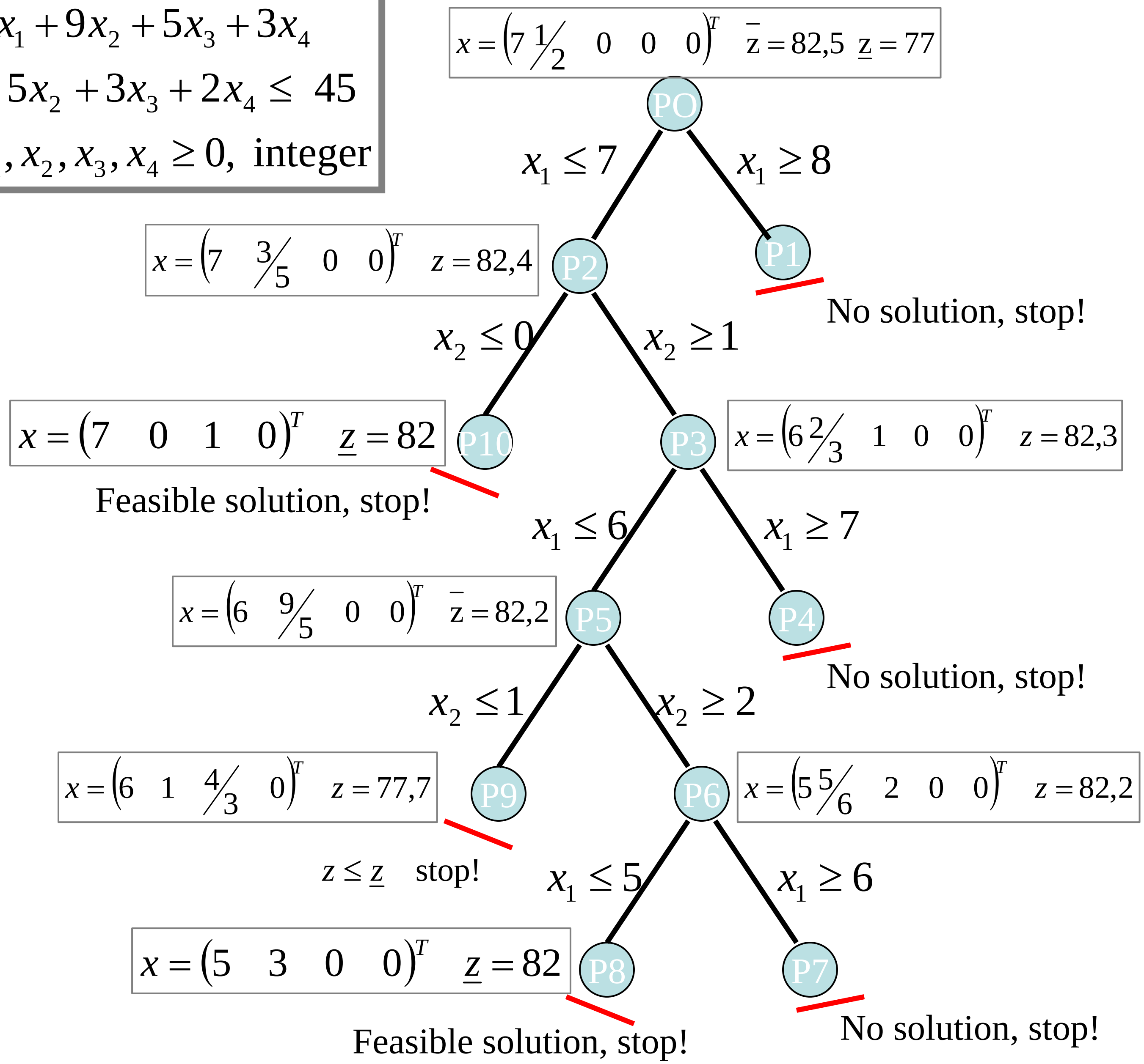
$$x_1, x_2 \ge 0, \text{ integer}$$





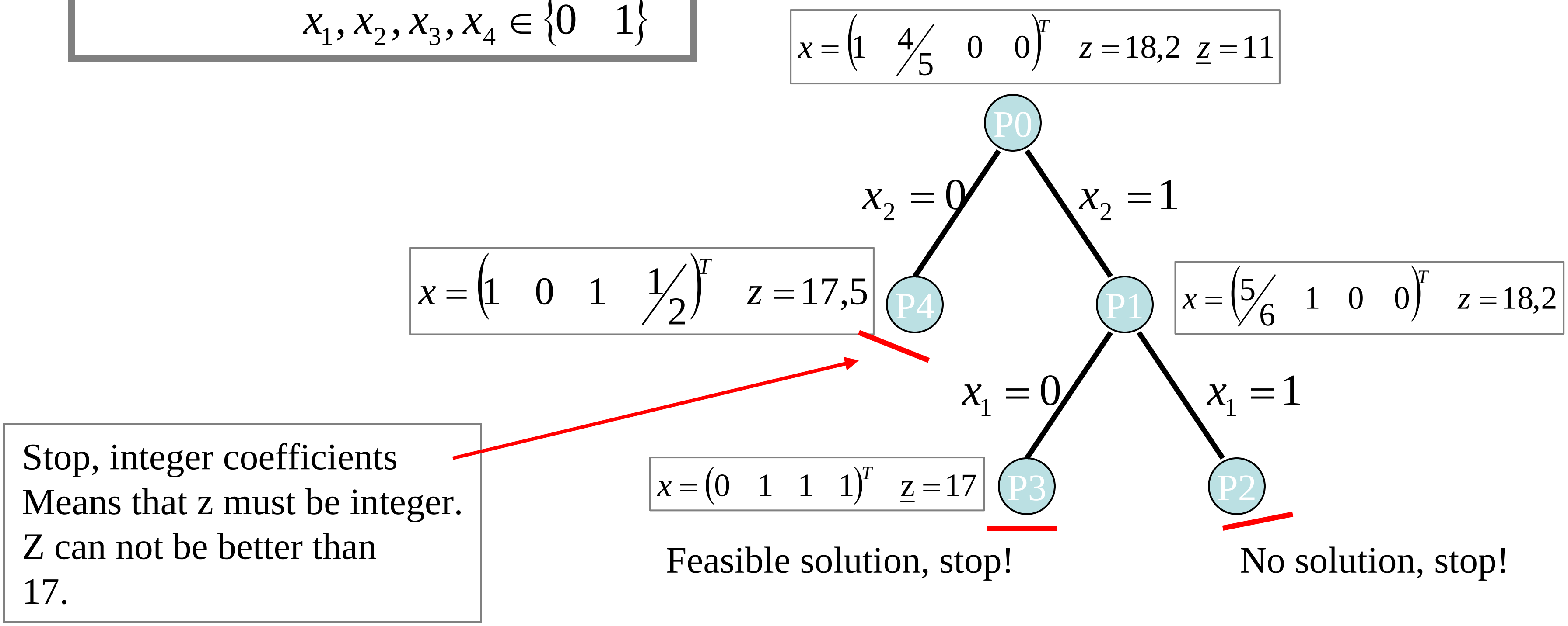
Optimum: $x^* = (0 \;\; 2)^T, \;\; z^* = 20$

$$
\begin{aligned}
\max \quad & z = 11x_1 + 9x_2 + 5x_3 + 3x_4 \\
\text{st} \quad & 6x_1 + 5x_2 + 3x_3 + 2x_4 \le 45 \\
& x_1, x_2, x_3, x_4 \ge 0, \text{ integer}
\end{aligned}
$$



Knapsack 0/1

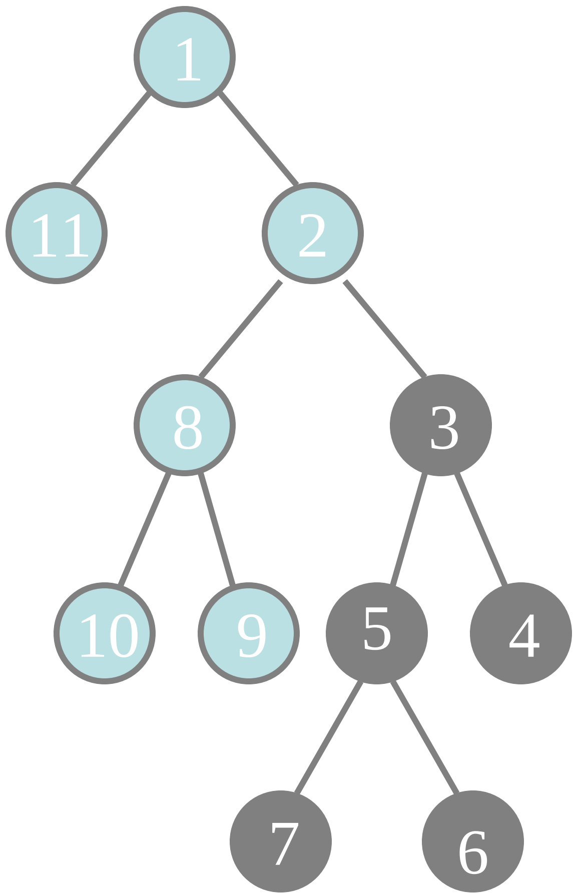
$$
\begin{aligned}
\max \quad z &= 11x_1 + 9x_2 + 5x_3 + 3x_4 \\
\text{st} \quad & 6x_1 + 5x_2 + 3x_3 + 2x_4 \leq 10 \\
& x_1, x_2, x_3, x_4 \in \{0 \quad 1\}
\end{aligned}
$$



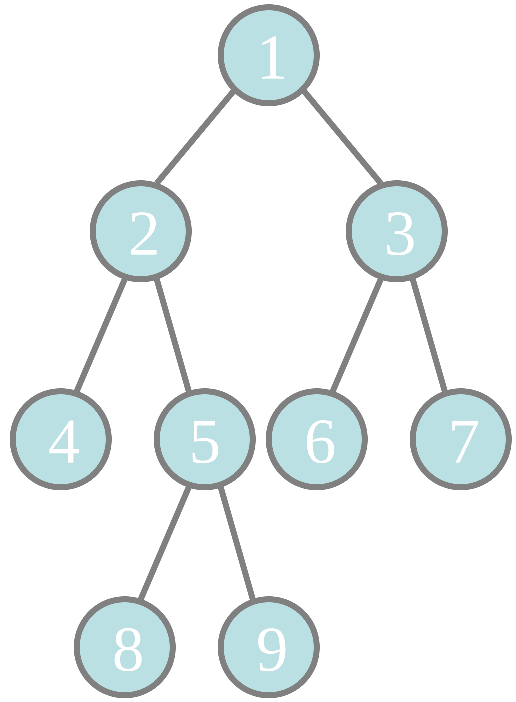
Many knapsack problems can be formulated and solved efficiently by dynamic programming.

Search strategies

Depth-first



Breadth first



Example B&B

Five jobs on one machine. Set up time dependent on sequence, se table. Decide sequence of jobs in order to minimize total time.

Variables:

$$x_{ij} = \begin{cases} 1, \text{if job } i \text{ is done as number } j \\ 0, \text{otherwise} \end{cases}$$

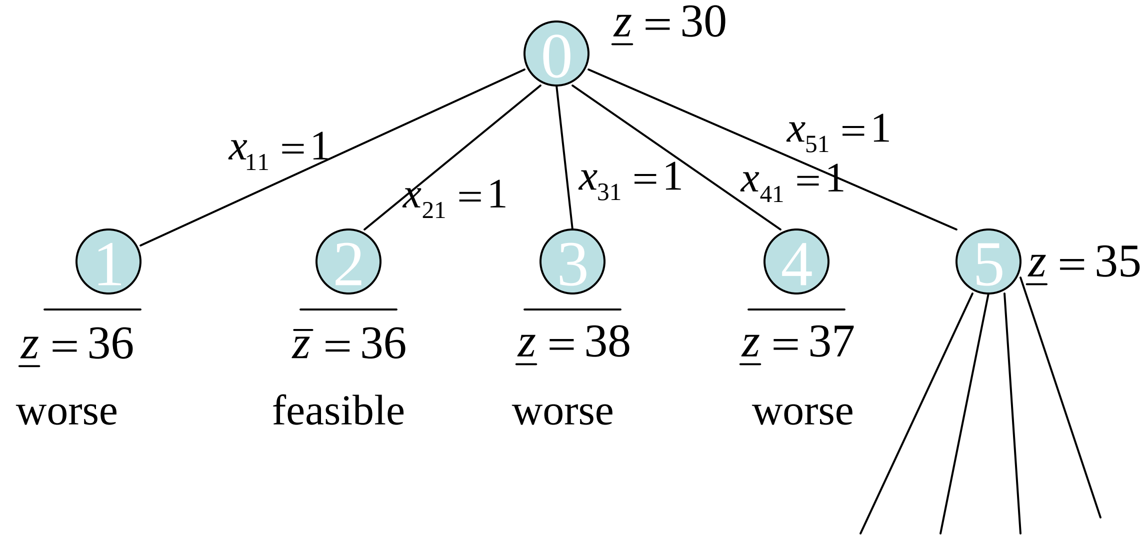
Previous	Job 1	2	3	4	5
none	4	5	8	9	4
1	–	7	12	10	9
2	6	–	10	14	11
3	10	11	–	12	10
4	7	8	15	–	7
5	12	9	8	16	–

B&B: relax the ordering. Solve by chosing cheapest cost in column. Fix one job and resolve.

Example, B&B

$P_0, \underline{z} = 30$

Prev	1	2	3	4	5
None	**4**	**5**	**8**	**9**	**4**
1	–	7	12	10	9
2	6	–	10	14	11
3	10	11	–	12	10
4	7	8	15	–	7
5	12	9	8	16	–



Further search

$P_2, \overline{z} = 36$

Prev	1	2	3	4	5
None	4	**5**	8	9	4
1	–	7	12	**10**	9
2	**6**	–	10	14	11
3	10	11	–	12	10
4	7	8	15	–	**7**
5	12	9	**8**	16	–

Feasible solution
2-1-4-5-3

$P_5, \underline{z} = 35$

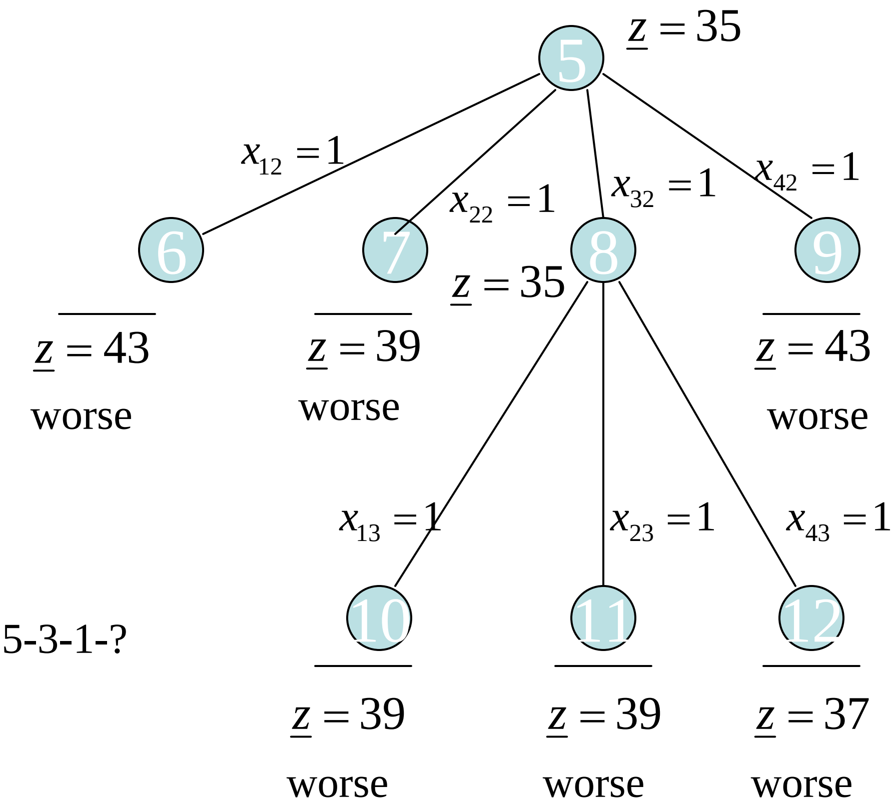
prev	1	2	3	4	5
none	4	5	8	9	**4**
1	–	**7**	12	**10**	9
2	**6**	–	10	14	11
3	10	11	–	12	10
4	7	8	15	–	7
5	12	9	**8**	16	–

$P_8, \underline{z} = 35$ sequence: 5-3-?

prev	1	2	3	4	5
none	4	5	8	9	**4**
1	–	**7**	12	**10**	9
2	**6**	–	10	14	11
3	10	11	–	12	10
4	7	8	15	–	7
5	12	9	**8**	16	–

$P_{10}, \underline{z} = 39$ sequence: 5-3-1-?

prev	1	2	3	4	5
none	4	5	8	9	**4**
1	–	**7**	12	**10**	9
2	6	–	10	14	11
3	**10**	11	–	12	10
4	7	8	15	–	7
5	12	9	**8**	16	–



Optimum in node 2

$z^* = 36$, sequence: 2-1-4-5-3

We were lucky to find a feasible solution fast.

In order

$\underline{z} = 40$

prev	1	2	3	4	5
none	**4**	5	8	9	4
1	–	**7**	12	10	9
2	6	–	**10**	14	11
3	10	11	–	**12**	10
4	7	8	15	–	**7**
5	12	9	8	16	–

cheapest

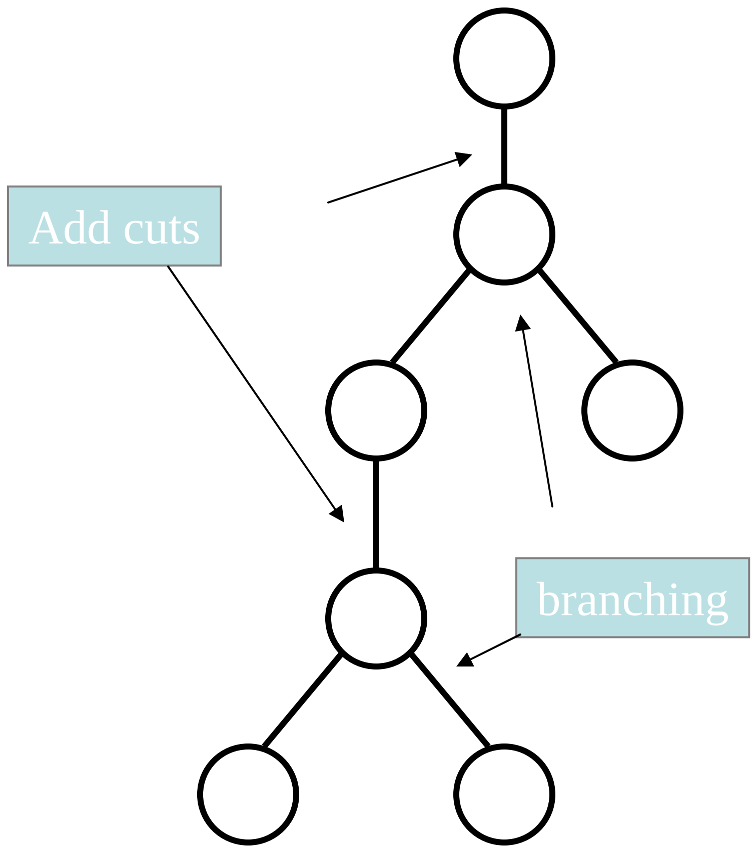
$\underline{z} = 43$

prev	1	2	3	4	5
none	4	5	8	9	**4**
1	–	**7**	12	10	9
2	6	–	10	**14**	11
3	**10**	11	–	12	10
4	7	8	15	–	7
5	12	9	**8**	16	–

Branch & Cut

Combination of B&B and cutting planes

Idea: Add cuts so that the LP-relaxation becomes stronger before branching

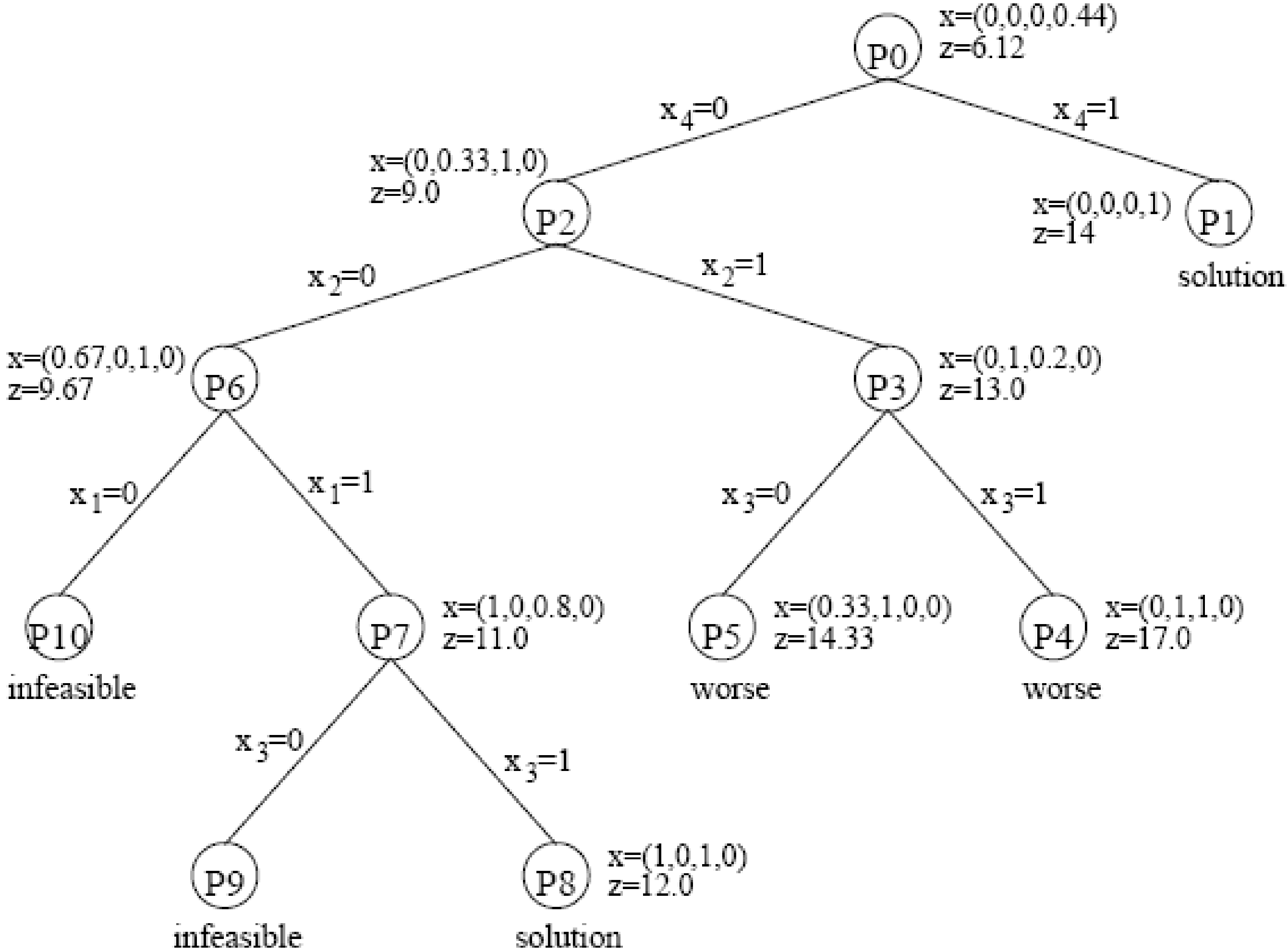


Branch & Cut

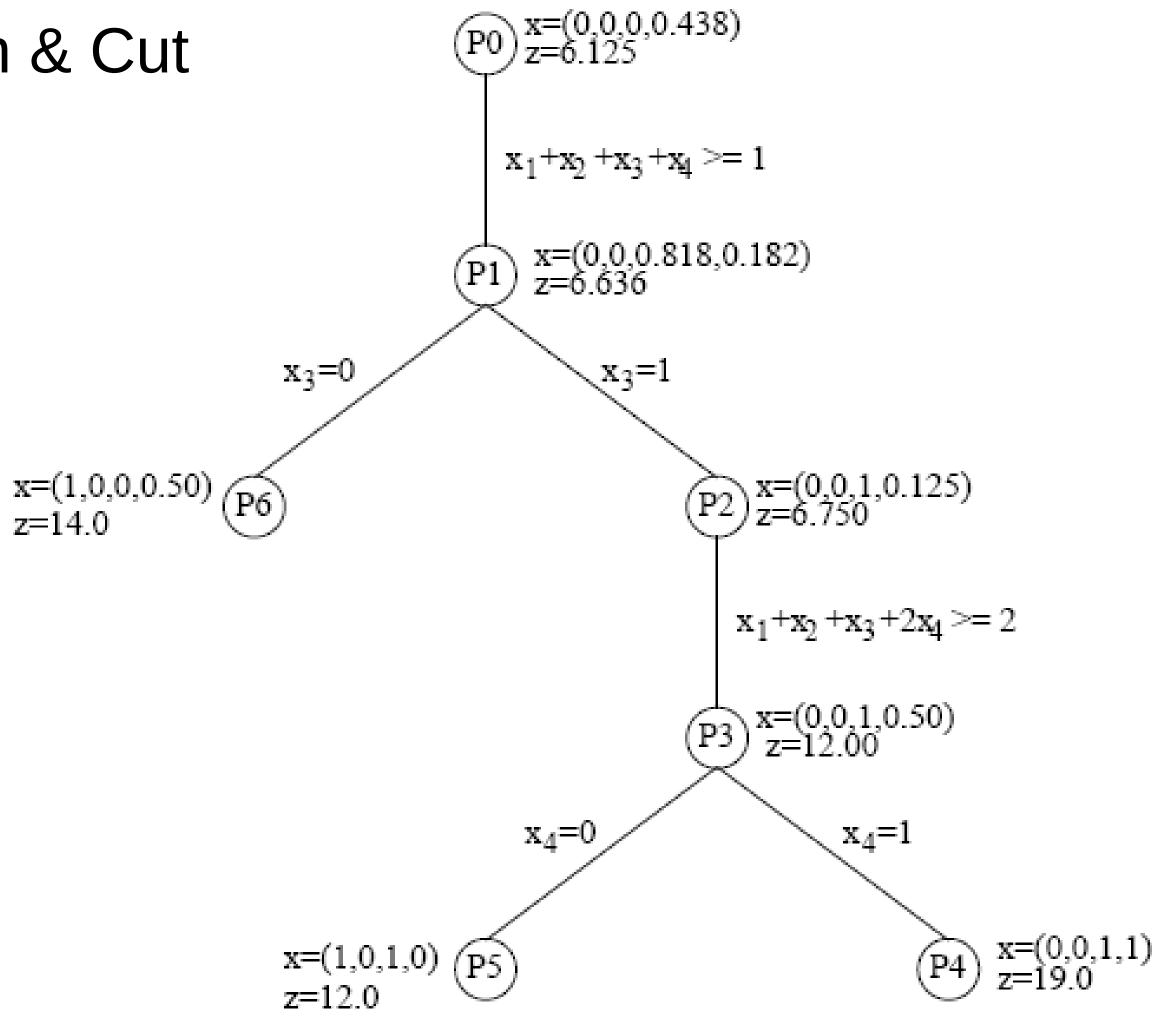
An important extension is to make the LP-relaxation stronger by adding valid inequalities. This is the idea behind *Branch & Cut*.
In each node in the search tree we add, if possible, one or several valid inequalities. We study a simple knapsack problem to illustrate the idea.

$$
\begin{array}{lrcl}
\min \quad z = & 7x_1 + 12x_2 + 5x_3 + 14x_4 & & \\
\text{då} & 300x_1 + 600x_2 + 500x_3 + 1600x_4 & \geq & 700 \\
 & x_1, \ldots, x_4 & \in & \{0, 1\}
\end{array}
$$

Normal branching



Branch & Cut


P0
x=(0,0,0,0.438)
z=6.125
x1+x2 +x3 +x4 >= 1
P1
x=(0,0,0.818,0.182)
z=6.636
x3=0
x3=1
x=(1,0,0,0.50)
z=14.0
P6
P2
x=(0,0,1,0.125)
z=6.750
x1+x2 +x3 +2x4 >= 2
P3
x=(0,0,1,0.50)
z=12.00
x4=0
x4=1
x=(1,0,1,0)
z=12.0
P5
P4
x=(0,0,1,1)
z=19.0

Comments:

- The valid inequalities generated in one node can also be used in other nodes and hence strengthen the LP relaxation in other nodes.
- How to generate valid inequalities is very application dependent.

Constraint branching

The standard approach is to branch on variables with fractional values. For some applications (e.g. set-partitioning models) this is very weak and gives a large search tree. In some set-partitioning models where a variable is set to one, many variables are set to 0. If the variable is set to 0, essentially nothing happens. To get at better balance we can choose to do a so-called *constraint branching*.

We will illustrate the technique with a set-partitioning model. Constraint branching means that two new subproblem is created by requiring that a set of variables should sum to 0 or 1 instead of a single variable. In a LP-solution to a set-partitioning model each constraint or object (to be partitioned) will be covered by one or several variables/columns (representing alternatives).

For each pairwise constraints, p and q, we can define the variables $j \in J_{pq}$ that have a coefficient that is non-zero for both constraints p and q, i.e.

$$J_{pq} = \{j \mid a_{pj} = 1 \text{ and } a_{qj} = 1\}.$$

In an integer solution we must have

$$\sum_{j \in J_{pq}} x_j = 1 \text{ or } \sum_{j \in J_{pq}} x_j = 0,$$

at the same time as there always exists a fractional LP-solution where there are two constraints p, q with the property

$$0 < \sum_{j \in J_{pq}} x_j < 1.$$

From the rote node we can define two sub-problems

$$P1: \quad P0 + \text{constraint} \sum_{j \in J_{pq}} x_j = 1$$

$$P2: \quad P0 + \text{constraint} \sum_{j \in J_{pq}} x_j = 0$$

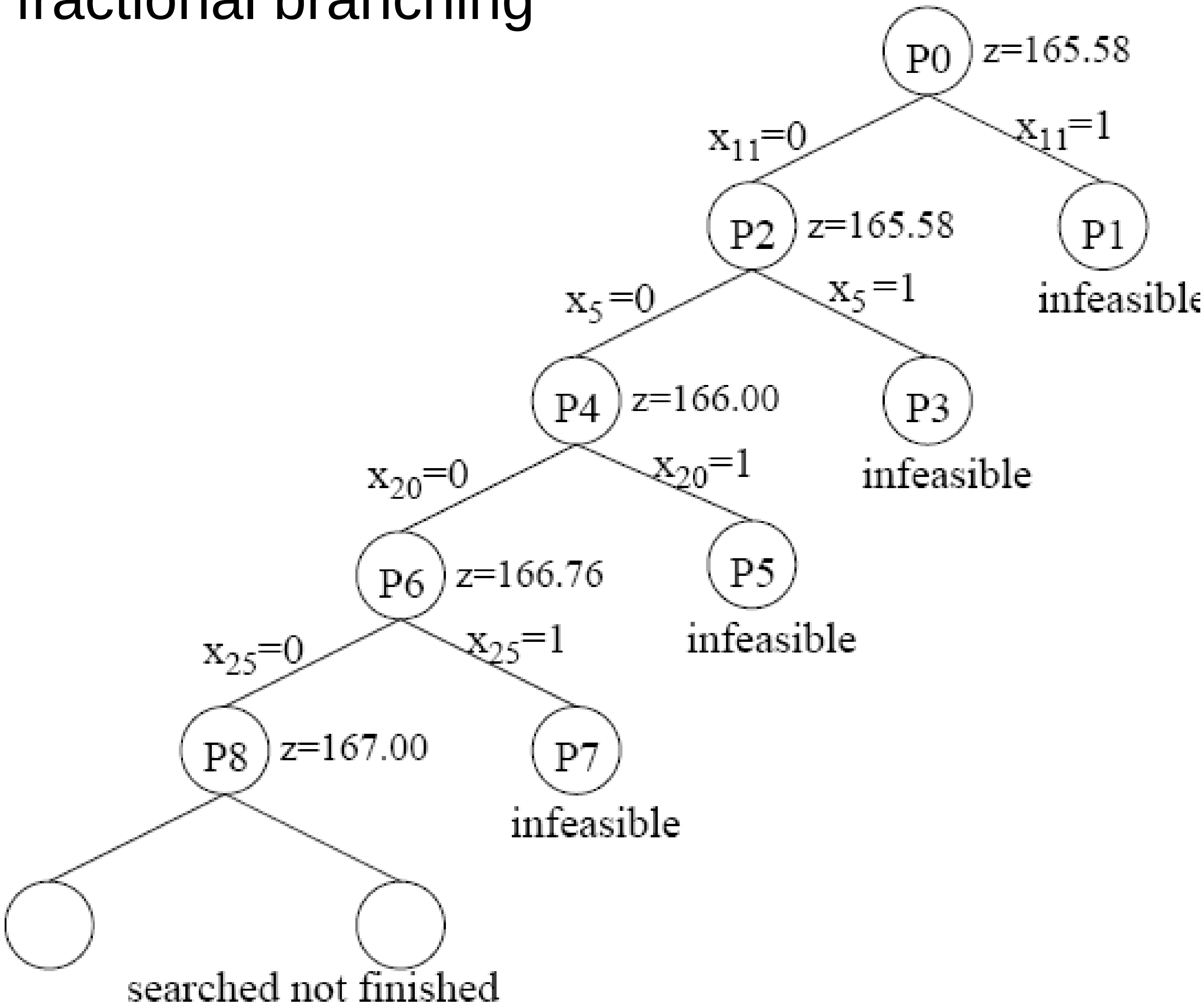
There are often several combinations of constraints to choose and normally the pair of constraints with highest value of $\sum_{j \in J_{pq}} x_j$ is chosen.

We study a routing problem with three vehicles. There are nine customers and 30 routes (ten for each vehicle). Each route visits 2,3 or 4 customers. The model is

$$
\begin{aligned}
\min \quad & z = (57\ 45\ 69\ 55\ 57\ 67\ 57\ 57\ 71\ 55\ 54\ 55\ 68\ 44\ 58\ 56\ 69 \\
& \qquad 55\ 56\ 54\ 55\ 56\ 71\ 56\ 53\ 43\ 69\ 55\ 71\ 43)^T \mathbf{x} \\
\text{s.t.} \quad &
\left[\begin{array}{cccccccccccccccccccccccccccccc}
1&1&1&1&1&1&1&1&1&1&&&&&&&&&&&&&&&&&&&&\\
&&&&&&&&&&1&1&1&1&1&1&1&1&1&1&&&&&&&&&&\\
&&&&&&&&&&&&&&&&&&&&1&1&1&1&1&1&1&1&1&1\\
1&&1&&&1&&&&&1&&1&&&&1&&&&1&1&&&&&&1&1&\\
&&&1&&&&1&&&&1&&1&1&1&&&1&&&1&1&1&&&&&&\\
1&&1&1&&1&&1&&&1&&&&&&1&1&&1&&&&&1&&&1&&\\
&1&&&1&&1&&1&&&&1&&1&&1&&&&&&1&1&&&&&1&1\\
1&&&&&&1&&&&&1&1&&1&&&&1&&&&&&&1&1&1&&1\\
&&1&&1&&&&1&1&1&&&&&&&1&&1&&&1&&1&&1&&&\\
&&1&1&&1&&&1&1&&1&&&&&&1&&&&&&1&&&1&&1&\\
&1&&&1&&&&&1&&&1&1&&1&1&&&1&1&&&&1&&&&1&\\
&&&&&1&1&1&&&&&&&&1&&&1&&1&1&1&&&1&1&&&
\end{array}\right] \mathbf{x} =
\left[\begin{array}{c}1\\1\\1\\1\\1\\1\\1\\1\\1\\1\\1\\1\end{array}\right]
\begin{array}{c}\text{(Vehicle 1)}\\\text{(Vehicle 2)}\\\text{(Vehicle 3)}\\\text{(Cust 1)}\\\text{(Cust 2)}\\\text{(Cust 3)}\\\text{(Cust 4)}\\\text{(Cust 5)}\\\text{(Cust 6)}\\\text{(Cust 7)}\\\text{(Cust 8)}\\\text{(Cust 9)}\end{array} \\
& \mathbf{x} \in \{0, 1\}
\end{aligned}
$$

if we solve the LP-relaxation we get $z = 165.58$ with $x_4 = 0.17, x_6 = 0.17, x_7 = 0.33, x_{10} = 0.33, x_{11} = 0.42, x_{12} = 0.17, x_{13} = 0.17, x_{16} = 0.25, x_{22} = 0.25, x_{24} = 0.17, x_{25} = 0.25, x_{30} = 0.33$ (other variables 0).

Normal fractional branching



In this case we combine vehicle constraints with customer constraints. For each combination (vehicle i and customer k) we compute the value $d_{ik} = \sum_{j \in J_i} a_{3+i,j} x_j$, where J_i are routes driven by vehicle i. These values are given in table 6. One interpretation is which proportion of routes that visit each customer. If we study vehicle 1 we can see that the LP-solution "visits" customer 1 at 17 % of the routes and customer 7 at 67 % of the routes.

	Vehicle 1	Vehicle 2	Vehicle 3
Customer 1	0.17	0.58	0.25
Customer 2	0.17	0.42	0.42
Customer 3	0.33	0.42	0.25
Customer 4	0.33	0.17	0.50
Customer 5	0.33	0.33	0.33
Customer 6	0.33	0.42	0.25
Customer 7	**0.67**	0.17	0.17
Customer 8	0.33	0.42	0.25
Customer 9	0.50	0.25	0.25

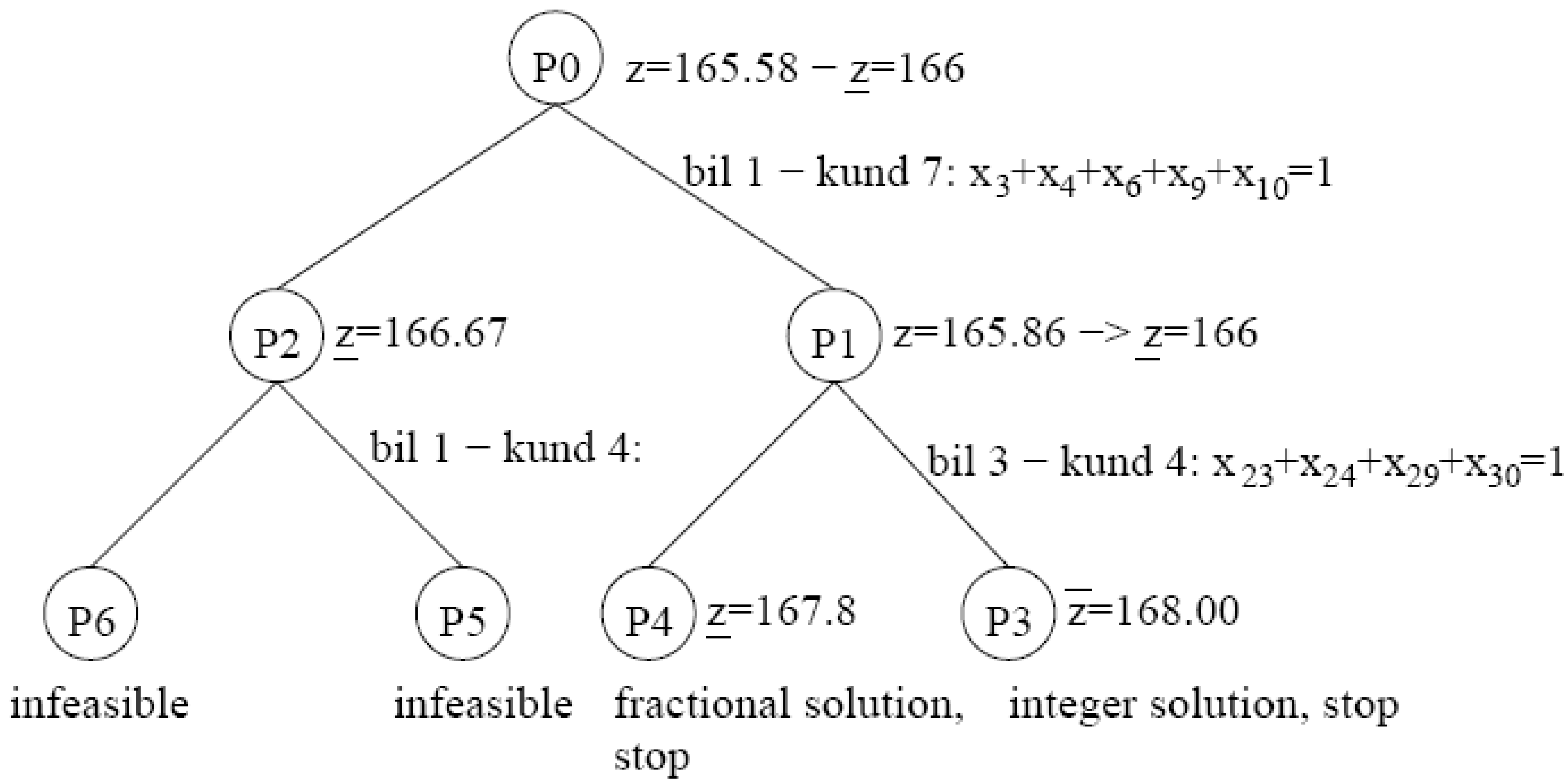
$$
\begin{aligned}
P1: &\quad \text{Vehicle 1 visit customer 7}: x_3 + x_4 + x_6 + x_9 + x_{10} = 1 \\
P2: &\quad \text{Vehicle 1 do not visit customer 7}: x_3 + x_4 + x_6 + x_9 + x_{10} = 0
\end{aligned}
$$

In this model it is easy to consider the branching decision. Instead of adding explicit constraints, we fix variables to 0. In $P1$ we know that $x_1 = x_2 = x_5 = x_7 = x_8 = 0$ since the other must sum to 1. In the same way we can in $P2$ require that $x_3 = x_4 = x_6 = x_9 = x_{10} = 0$. We select $P1$ since the solution suggest that vehicle 1 should visit customer 7.

The new solution from $P1$ gives $z = 165.86$ with $x_4 = 0.29, x_6 = 0.29, x_{10} = 0.43, x_{11} = 0.36, x_{13} = 0.21, x_{16} = 0.29, x_{19} = 0.14, x_{22} = 0.14, x_{23} = 0.14, x_{25} = 0.07, x_{30} = 0.64$ (other variables 0). Corresponding values of vehicle-customer are given in table 7. Note that vehicle 1 - customer 7 has value 1.

	Vehicle 1	Vehicle 2	Vehicle 3
Customer 1	0.29	0.57	0.14
Customer 2	0.29	0.43	0.29
Customer 3	0.57	0.36	0.07
Customer 4	0.00	0.21	**0.79**
Customer 5	0.00	0.36	0.64
Customer 6	0.43	0.36	0.21
Customer 7	1.00	0.00	0.00
Customer 8	0.43	0.50	0.07
Customer 9	0.29	0.43	0.29

Next branching is done for vehicle - Customer 4 which has a value of 0.79. The entire search tree is given in figure 6. In subproblem $P4$ we get $\underline{z} = 167.8$ but since the coefficients are integer we get $\underline{z} = 168 \geq \overline{z}$. The optimal solution is $x_3^* = x_{16}^* = x_{30}^* = 1$ with $z^* = 168$.



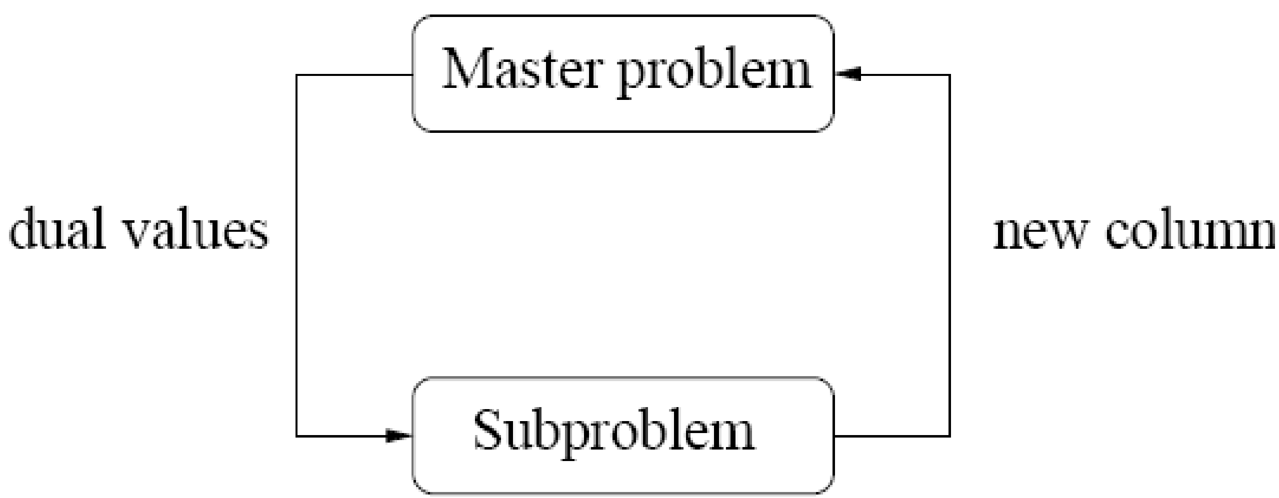
Branch & Price

- In many applications (e.g. Set partitioning type models) it is not possible to generate all variables explicit
- Column generation – a method to solve large scale LP problems can be used in each of the B&B nodes

Column generation

$$\begin{array}{lrcl} \min & z = \mathbf{c}^T\mathbf{x} & & \\ \text{då} & \mathbf{A}\mathbf{x} & = & \mathbf{b} \\ & \mathbf{x} & \geq & \mathbf{0} \end{array}$$

A has many columns as compared to the number of constraints



Algorithm – Column generation:

Step 0 Choose a subset $R^0 \subset R$ of columns with the property

$$\sum_{j \in R^0} a_{ij} x_j = b_i,\ i = 1 \ldots, m. \text{ Set } k = 0.$$

Step 1 Solve Master problem

$$\begin{aligned} \min \quad & z = \sum_{j \in R^k} c_j x_j \\ \text{s.t.} \quad & \sum_{j \in R^k} a_{ij} x_j \ = \ b_i, \quad i = 1, \ldots, m \\ & x_j \ \geq \ 0, \quad j \ \in \ R^k \end{aligned}$$

Let $\mathbf{x}^{(k)}$ and $\mathbf{v}^{(k)}$ denote the primal and dual solution.

Step 2 Solve subproblem

$$w^* = \min_{j \in R \setminus R^k} \left\{ c_j - \sum_{i=1}^{m} v_i^{(k)} a_{ij} \right\}$$

Application dependent

and let the optimal solution be a new column (variable) s.

Step 3 Check convergence. The point $\mathbf{x}^{(k)}$ is an optimal solution if $w^* \geq 0$.

Step 4 Add column s till R^k, i.e. update $R^{k+1} = R^k \cup \{s\}$.

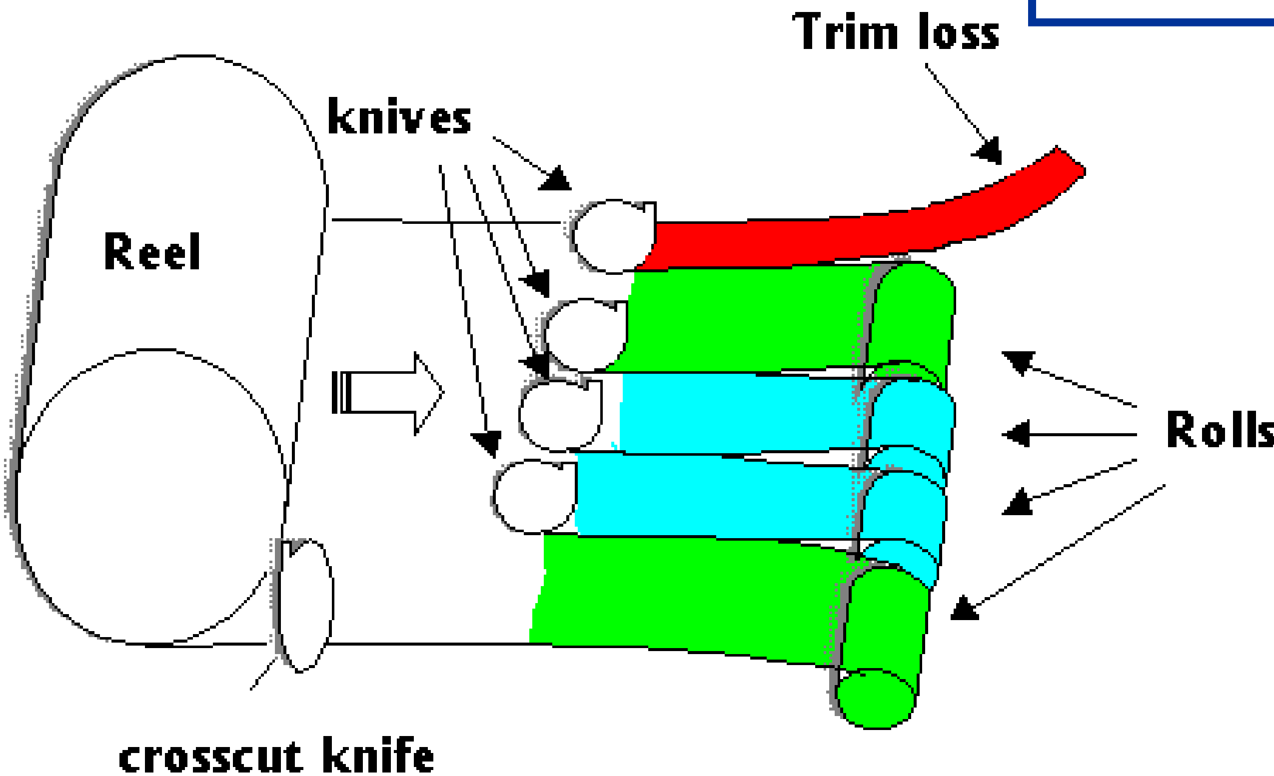
Step 5 Update $k := k + 1$ and go to Step 1.

Cases & Results

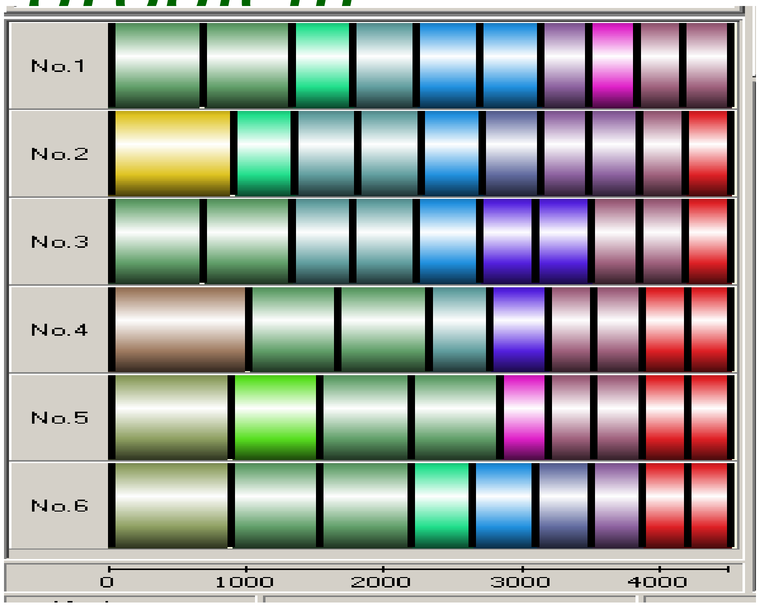
Paper roll cutting

Roll cutting at paper mills

Length: 30,000 meters
Width: 5-10 meters
Products: 0.3-1.0 m
Roll: 5,000 meters
Fixed demand



Tactical problem: *Cutting stock problem*



$$[\mathrm{P1}] \quad w = \min \sum_{j=1}^{n} x_j$$

$$\text{s.t.} \ \sum_{j=1}^{n} a_{ij} x_j \geq b_i, \quad i = 1,\dots,m$$

$$x_j \geq 0, \ \text{integer}, \quad j = 1,\dots,n$$

Coefficients:

b_i = Demand of roll i .

a_{ij} = Number of times roll i appears in pattern j.

Variables:

x_j = Number of times pattern j is used.

Standard solution approach

- Classical Gilmore & Gomory: Column generation and B&B

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

$$s.t. \sum_{j=1}^{n} a_{ij} x_j - s_i = b_i, \quad i = 1, \dots, m$$

$$x_j \geq 0, \text{ integer}$$

duals →

← *new pattern*

$$\min \text{rc} = 1 - \sum_{j=1}^{n} v_i w_i$$

$$s.t. \sum_{i=1}^{m} d_i w_i \leq L$$

$$w_i \geq 0, \text{ integer}, i = 1, \dots, m$$

Data:

d_i = width of product *i*.

L = width of main roll.

v_i = dual variable for constraint *i*

Variables:

w_i =number of product *i* in new pattern.

Practical considerations

- The number of rolls in a pattern is limited.
- The number of different rolls in a pattern is limited.
- Some rolls are not allowed to be in the same pattern.
- Some rolls must be included in the same pattern.
- There is a maximum allowed trim loss.
- Demand is given as a target value with bounds on under- and overproduction.

OnTop Trim - [Case A]

File Settings Window Help

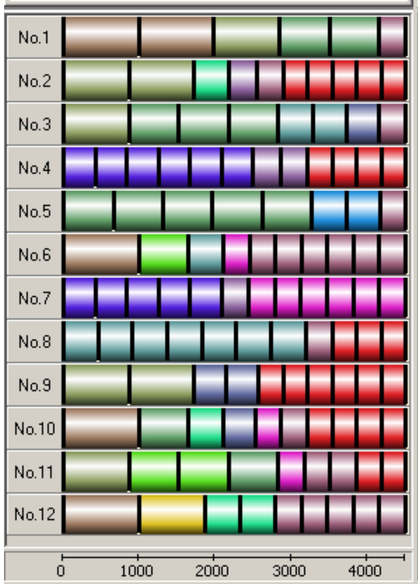
Saved Solutions | Solve Settings

Trim Loss	321
Reel Width [mm]	4500
Max Product Count	14
Max Unit Count	13
Max solutions:	0
Max time [min]:	0

Sorting: <No sorting>

Products | Calculation Results

Name:	Length:	Width:	Core	Demand:	MaxNrInP...	LowerBou...	UpperBou...	Shipping date	Price	Weight
P933	8903	993	none	20	100	20	21	2001-01-24	0	0
P879	8903	879	76mm	6	100	6	6	2001-01-24	0	0
P864	8903	864	76mm	25	100	25	26	2001-01-24	0	0
P655	8903	655	none	20	100	20	21	2001-01-24	0	0
P654	8903	654	70mm	150	100	149	151	2001-01-24	0	0
P459	8903	459	none	19	100	19	20	2001-01-24	0	0
P456	8093	456	70mm	84	100	84	85	2001-01-24	0	0
P444	8903	444	none	49	100	49	51	2001-01-24	0	0
P420	8093	420	78mm	11	100	10	12	2001-01-24	0	0
P415	8903	415	none	64	100	63	65	2001-01-24	0	0
P361	8903	361	none	25	100	25	24	2001-01-24	0	0
P344	8903	344	70mm	28	100	27	29	2001-01-24	0	0
P342	8903	342	none	146	100	146	150	2001-01-24	0	0
P322	8903	322	none	118	100	117	119	2001-01-24	0	0



	P933	P879	P864	P655	P654	P459	P456	P444	P420	P415	P361	P344	P342	P322	Count	Trim loss [%]
Pattern 1	2	0	1	0	2	0	0	0	0	0	0	0	1	0	1	0.000
Pattern 2	0	0	2	0	0	1	0	0	0	0	1	0	1	5	5	0.000
Pattern 3	0	0	1	0	3	0	2	0	1	0	0	0	1	0	6	0.000
Pattern 4	0	0	0	0	0	0	0	0	0	6	2	0	0	4	9	0.000
Pattern 5	0	0	0	0	5	0	0	2	0	0	0	0	1	0	25	0.000
Pattern 6	1	0	0	1	0	0	1	0	0	0	0	1	6	0	10	0.000
Pattern 7	0	0	0	0	0	0	0	0	0	5	1	6	0	0	2	0.000
Pattern 8	0	0	0	0	0	0	7	0	0	0	0	0	1	3	9	0.000
Pattern 9	0	0	2	0	0	0	0	0	2	0	0	0	0	6	2	0.000
Pattern 10	1	0	0	0	1	1	0	0	1	0	0	1	1	4	2	0.000
Pattern 11	0	0	1	2	1	0	0	0	0	0	0	1	2	2	5	0.000
Pattern 12	1	1	0	0	0	2	0	0	0	0	0	0	5	0	6	0.000
Demand Diff:	**0**	**0**	**1**	**0**	**2**	**0**	**1**	**1**	**1**	**0**	**0**	**1**	**2**	**0**		

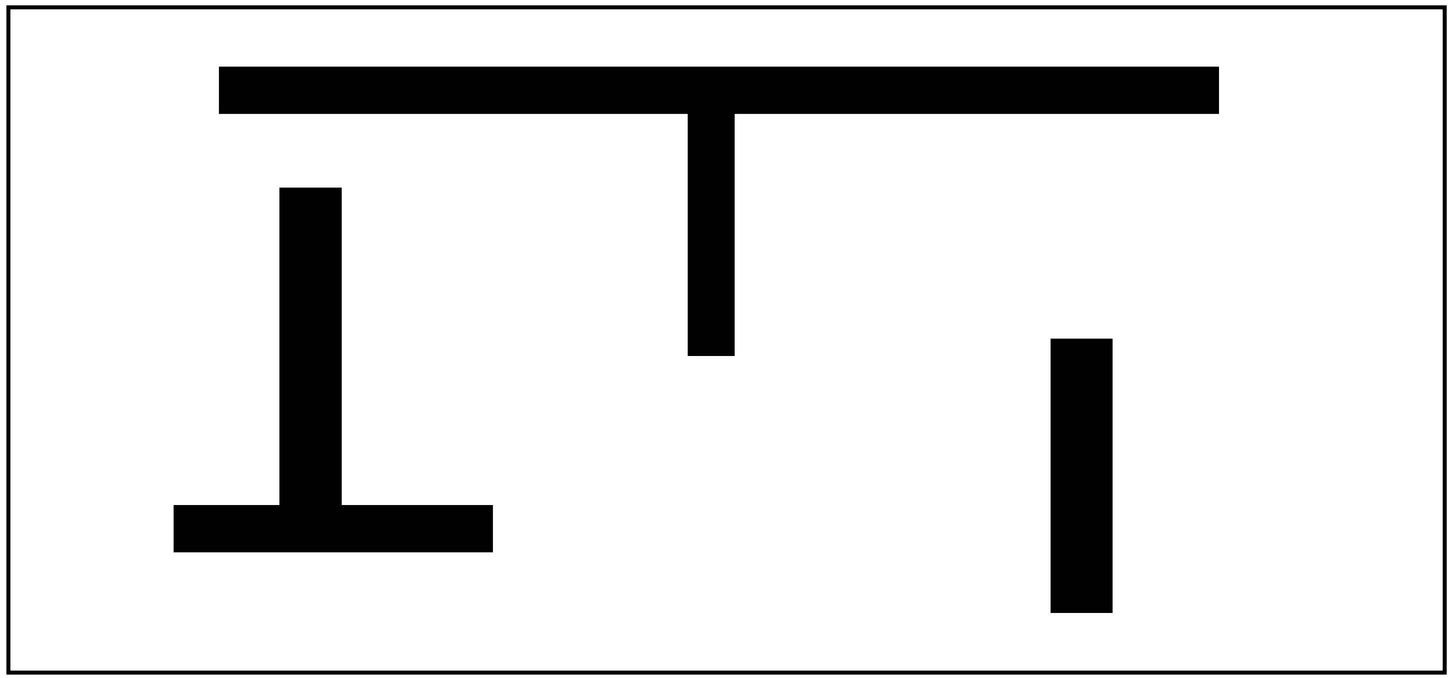
Summary

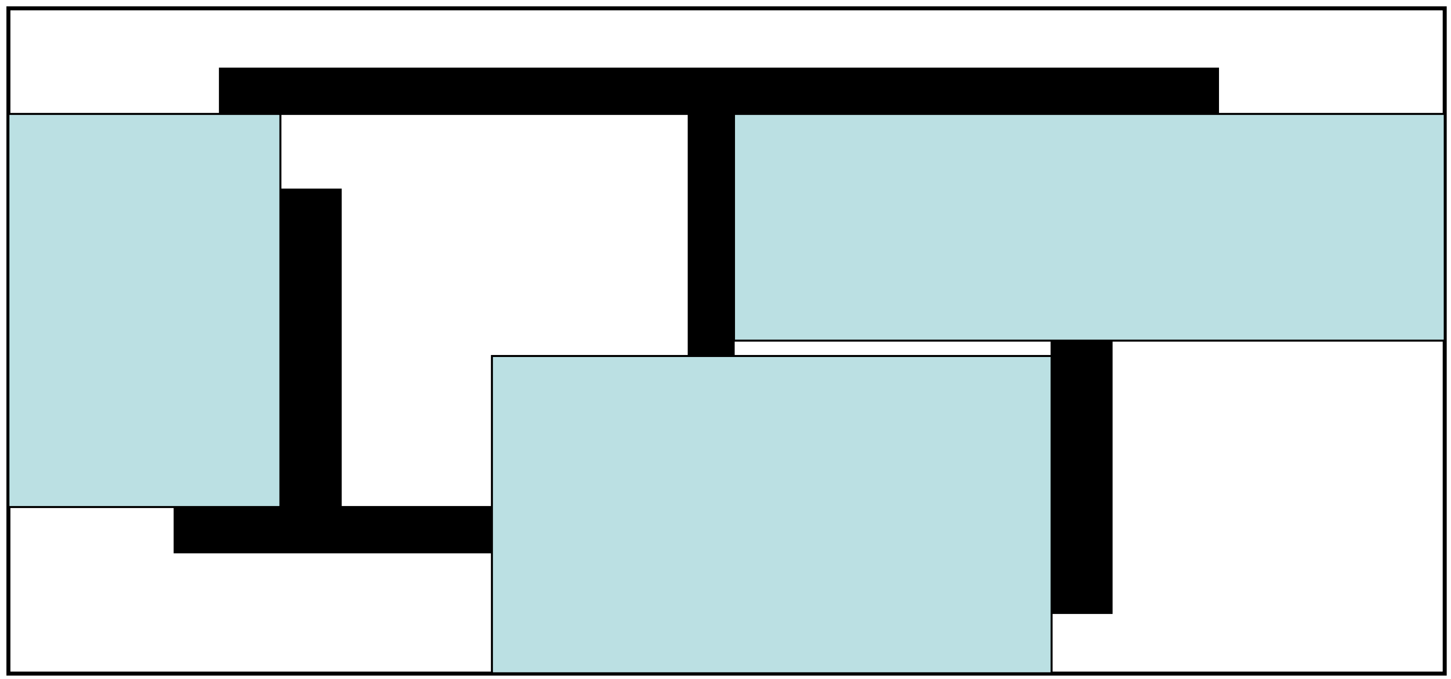
Set Count:	**82**
Under production:	**0**
Over production:	**9**
Total Trimloss [mm]:	**0**
Total Trimloss [%]:	**0.000**
Total date faults:	**0**
Total core faults:	**24**

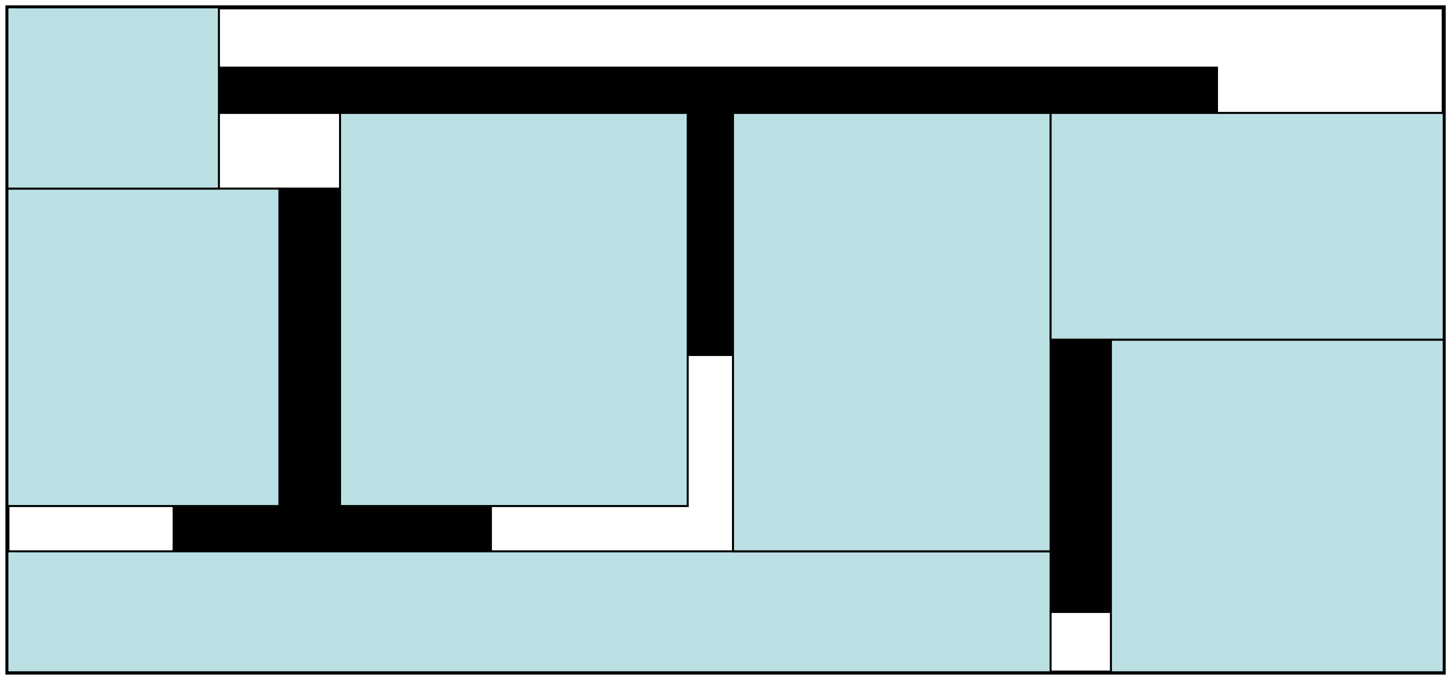
Modified

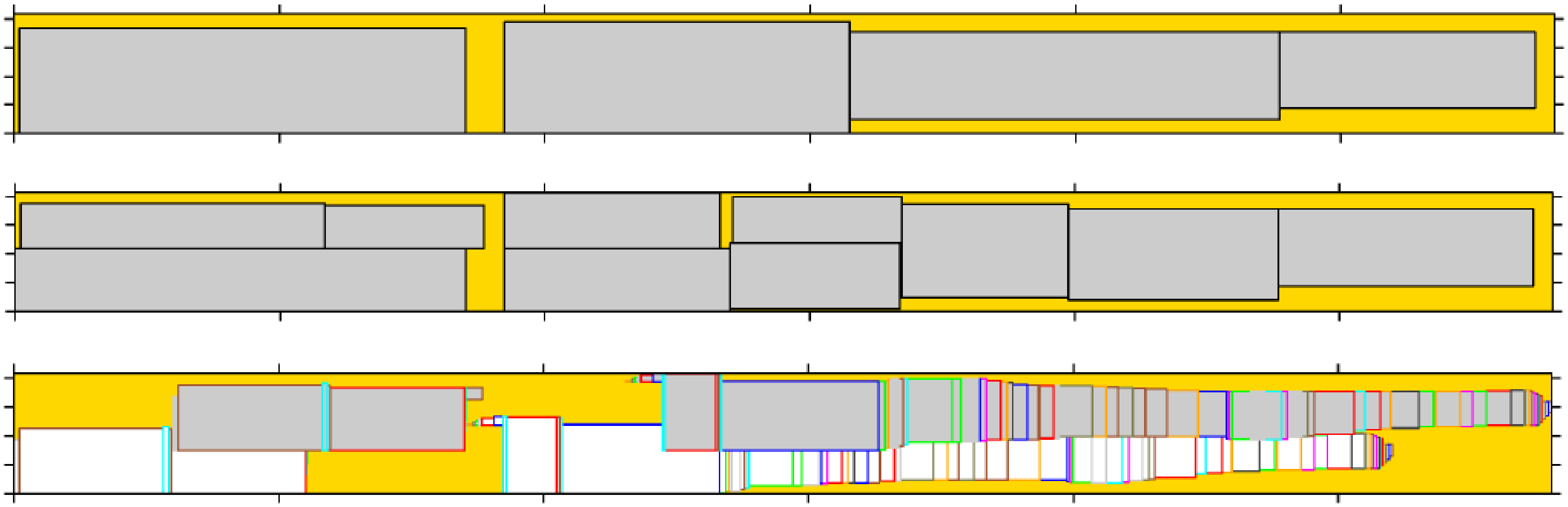
Start | Ma... | Inb... | My... | Vis... | Del... | Do... | On... | Mina doc | 14:40

2D Board cutting

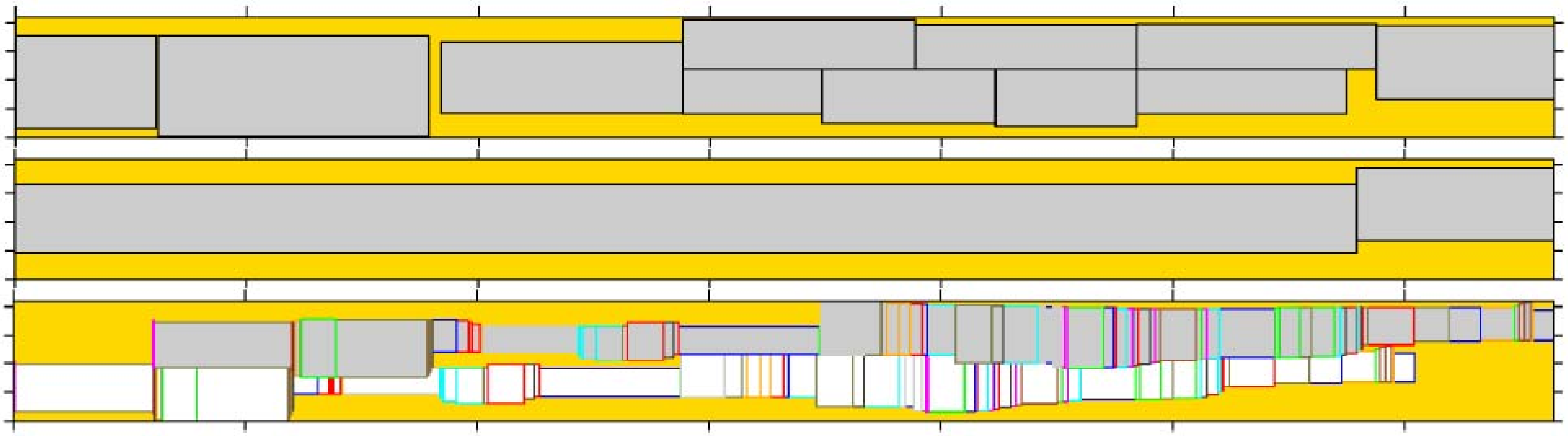






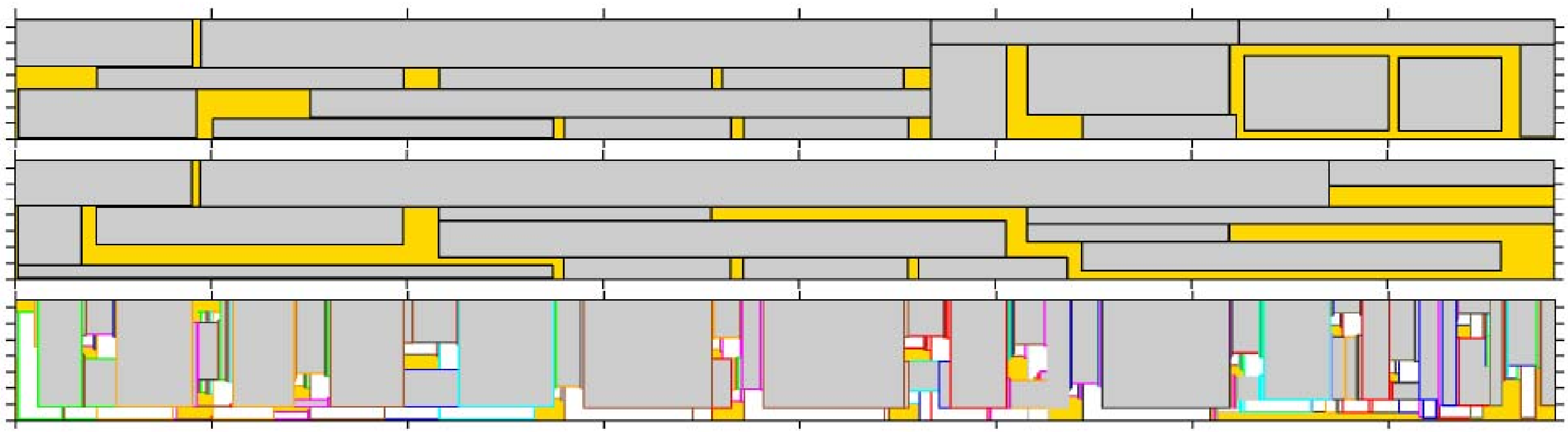


Upper part: optimal placement of max 4 products.
Middle part: optimal placement of max 4 products.
Lower part: Allocation areas (defined through defects)

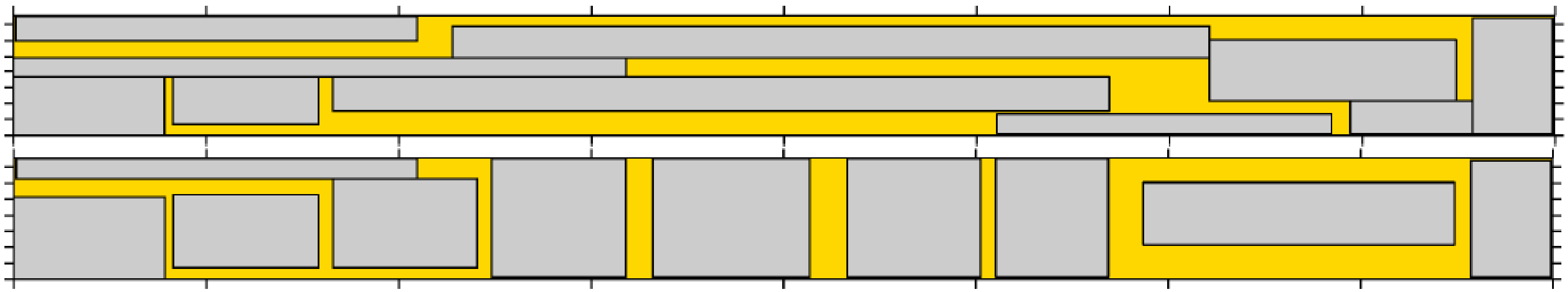


Upper part: optimal placement (max 100 products)
Middle part: current heuristic solution
Lower part: Allocation areas (defined through defects)

ca. 2



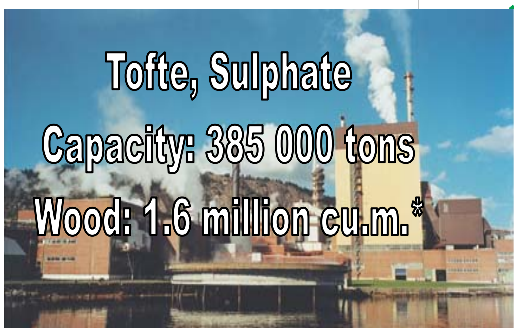
Upper part: optimal placement (max 100 products)
Middle part: current heuristic solution
Lower part: Allocation areas (defined through defects)

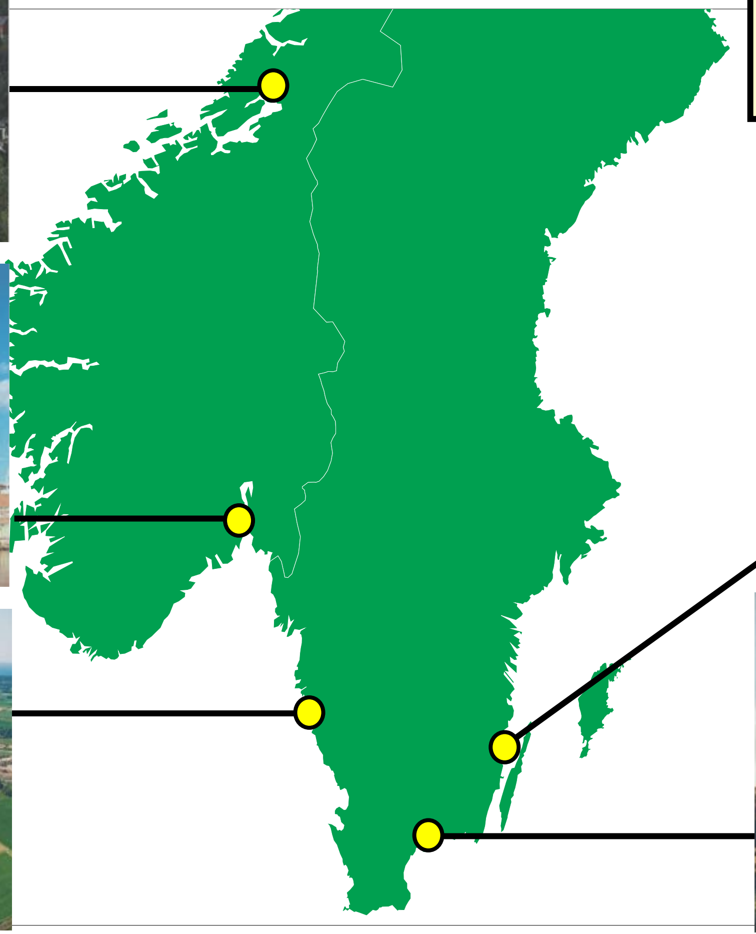


Upper part: heuristic solution (note non-guilliotine solution)
Middle part: optimal placement

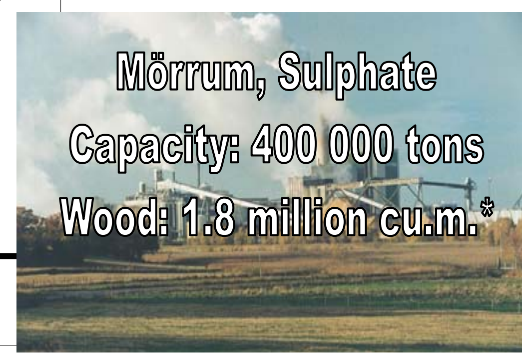
Terminal location

Case: Södra Cell - mills

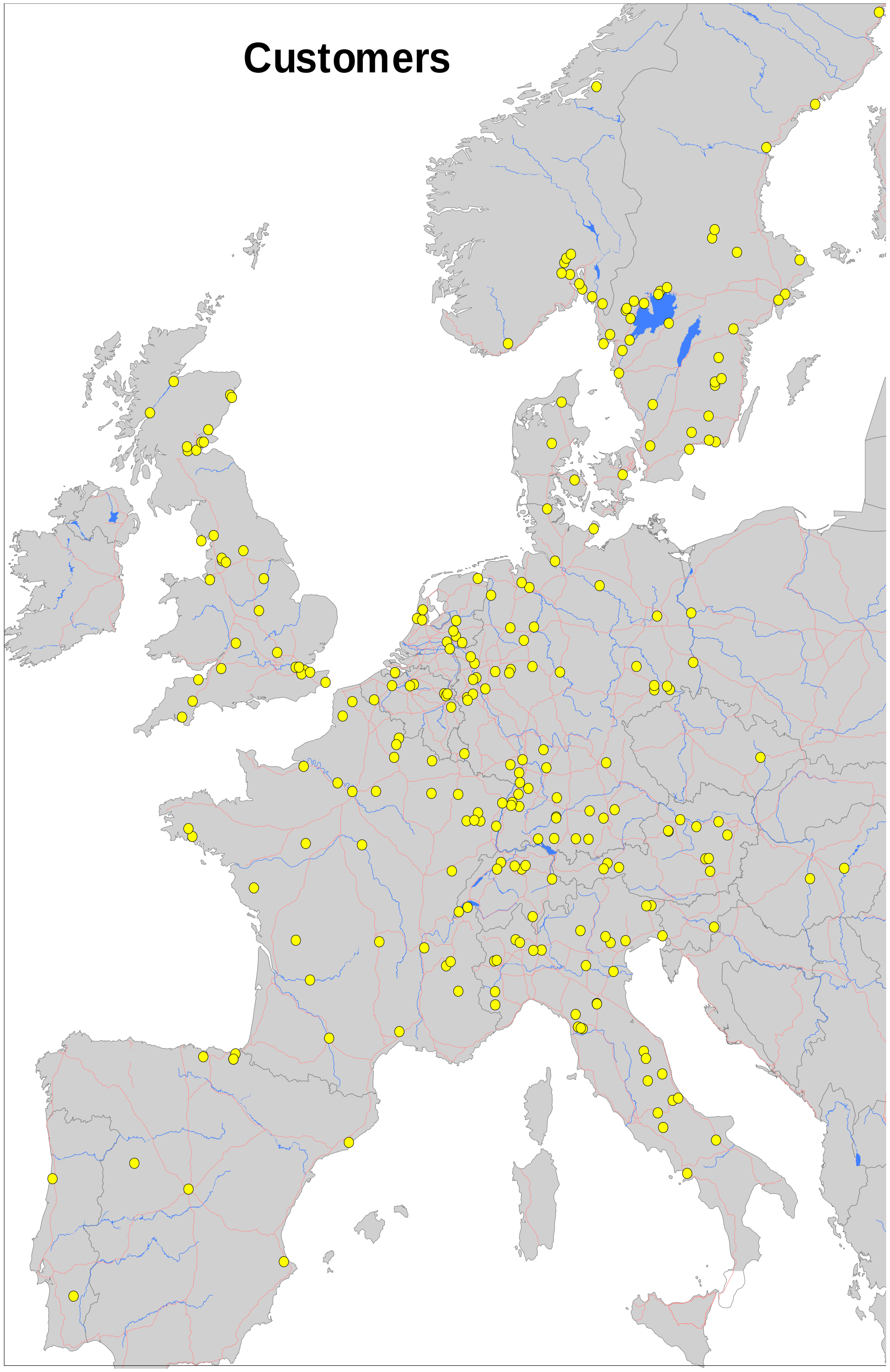




Södra Cell in total
Capacity: ~2 million tons
Wood: ~9 million cu.m.*

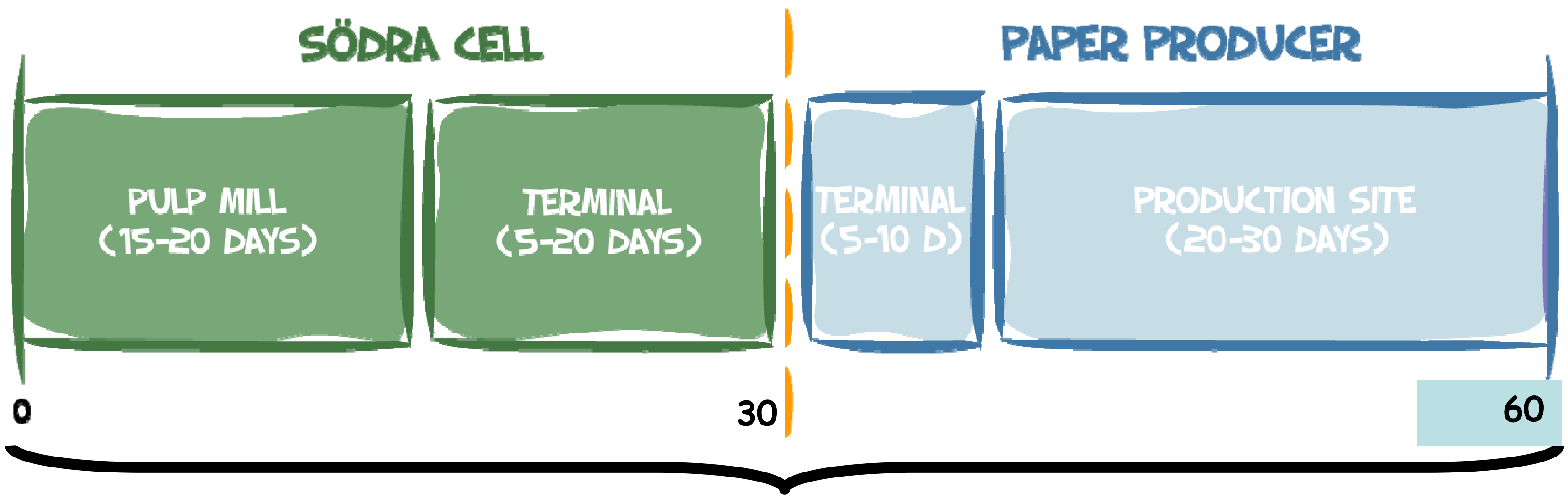


**cu.m. = cubic meters, solid under bark*


Customers


Terminals
Terminaler
Folla (4)
Svenska (17)
Svenska, Tofte (2)
Tofte (7)
SKOGN
HONEFOSS
HALDEN
INVERNESS
ABERDEEN
CORPACH
DUNDEE
GRANGEMOUTH
SUNDERLAND
KIEL
HULL
GRIMSBY
EMDEN
STETTIN
BREMEN
LOWESTOFT
NORTHFLEET
TERNEUZEN
WESEL
CHATHAM
ANTWERPEN
GHENT
BOULOGNE
BASEL
LA PALLICE
GENUA
PASAJES
AVEIRO
GANDIA

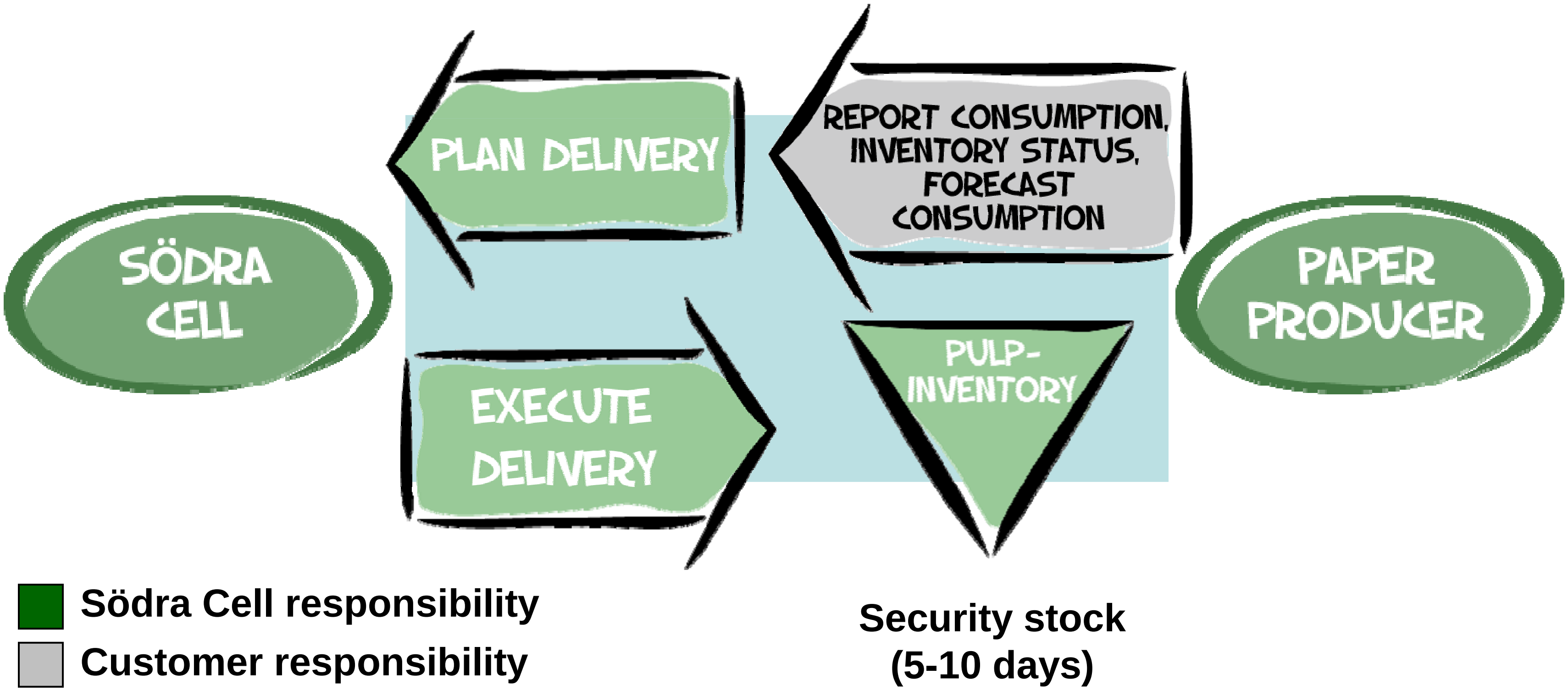
Inventories / Lead times ...



>15% of volume => 320 000 tonnes
€ 150 million

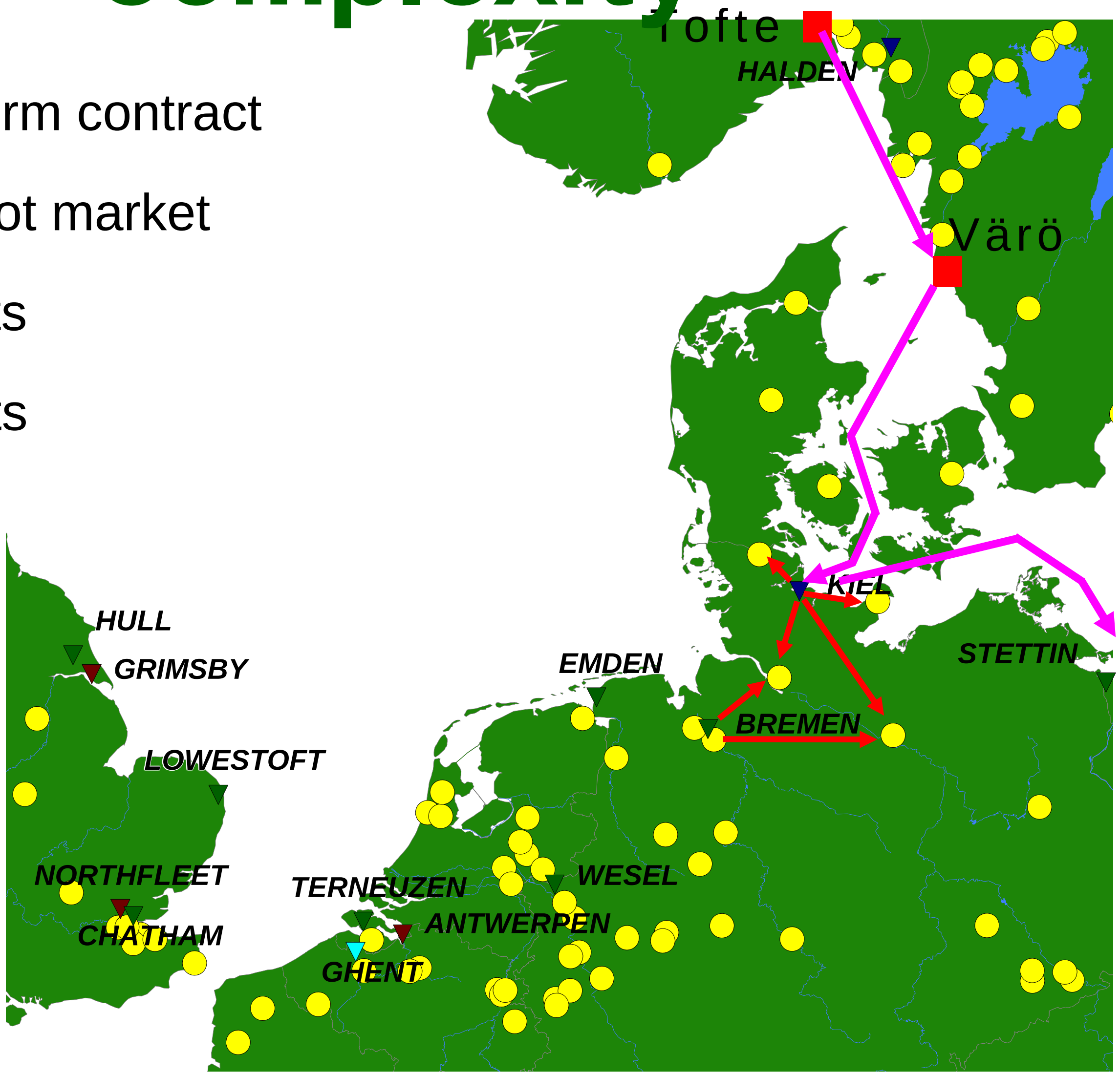
Storage cost (20%)
€ 30 million

PulpSMI:
Supplier Managed Inventory



Distribution - Complexity

- Three vessels on long term contract
- Additional vessels on spot market
- Train and truck transports
- Several different products
- Supply ≈ Demand
- Alternative shipment ports
- Alternative terminals



Mathematical model: objective

$$\min \sum_{k\in R_A}\sum_{i\in I}\sum_{j\in J_H}\sum_{p\in P}(c_k^A + c_i^{MP})x_{kijp}^A +$$ *A-routes*

$$\sum_{k\in R_B}\sum_{i\in I}\sum_{j\in J_H}\sum_{p\in P}(c_k^B + c_i^{MP})x_{kijp}^B +$$ *B-routes*

$$\sum_{i\in I}\sum_{j\in J_H}\sum_{p\in P}(c_{ij}^S + c_i^{MP})x_{ijp}^S + \sum_{j\in J}\sum_{i\in I} c_{ji}^R x_{ji}^R +$$ *Spot + return trips*

$$\sum_{j\in J} c_j^T y_j^{tot} + \sum_{j\in J} f_j z_j +$$ *Terminal costs*

$$\sum_{h\in J_H}\sum_{l\in J_L}\sum_{p\in P} c_{hl}^T y_{hlp}^T + \sum_{j\in J}\sum_{q\in Q}\sum_{p\in P} c_{jq}^Q y_{jqp}^Q +$$ *Distribution from terminals*

$$\sum_{i\in I}\sum_{q\in Q}\sum_{p\in P} c_{iq}^{train} y_{iqp}^{train} + \sum_{i\in I}\sum_{q\in Q}\sum_{p\in P} c_{iq}^{truck} y_{iqp}^{truck}$$ *Train + truck*

Mathematical model: constraints

$$\sum_{k \in R_A} \sum_{i \in I} \sum_{p \in P} x_{kijp}^{A} \leq \sum_{l \in L} \alpha_l \, p_l \, w_{jl}^{T} \quad \forall j \in J_H$$

Proportion A-routes

$$\sum_{k \in R_A} \sum_{i \in I} x_{kijp}^{A} + \sum_{k \in R_B} \sum_{i \in I} x_{kijp}^{B} + \sum_{i \in I} x_{ijp}^{S} = \sum_{l \in L} y_{jlp}^{T} + \sum_{q \in Q} y_{jqp}^{Q} \quad \forall j \in J_H, p \in P$$

Flow conservation

$$\sum_{h \in J_H} y_{hjp}^{T} = \sum_{q \in Q} y_{jqp}^{Q} \quad \forall j \in J_L, p \in P$$

Flow conservation

$$\sum_{j \in J} y_{jqp}^{Q} + \sum_{i \in I} y_{iqp}^{train} + \sum_{i \in I} y_{iqp}^{truck} = D_{qp} \quad \forall q \in Q, p \in P$$

Customer demand

$$\sum_{k \in R_A} \sum_{i \in I} \sum_{j \in J_H} \sum_{p \in P} t_k^{A} x_{kijp}^{A} + \sum_{k \in R_B} \sum_{i \in I} \sum_{j \in J_H} \sum_{p \in P} t_k^{B} x_{kijp}^{B} + \sum_{j \in J_H} \sum_{i \in I} x_{ji}^{R} \leq r$$

Ship capacity

$$\sum_{k \in R_A} \sum_{j \in J_H} \sum_{p \in P} x_{kijp}^{A} + \sum_{k \in R_B} \sum_{j \in J_H} \sum_{p \in P} x_{kijp}^{B} = \sum_{j \in J_H} x_{ji}^{R} \quad \forall i \in I$$

Return flow

$$\sum_{k \in R_A} \sum_{i \in I} \sum_{p \in P} x_{kijp}^{A} + \sum_{k \in R_B} \sum_{i \in I} \sum_{p \in P} x_{kijp}^{B} = \sum_{i \in I} x_{ji}^{R} \quad \forall j \in J$$

Return flow

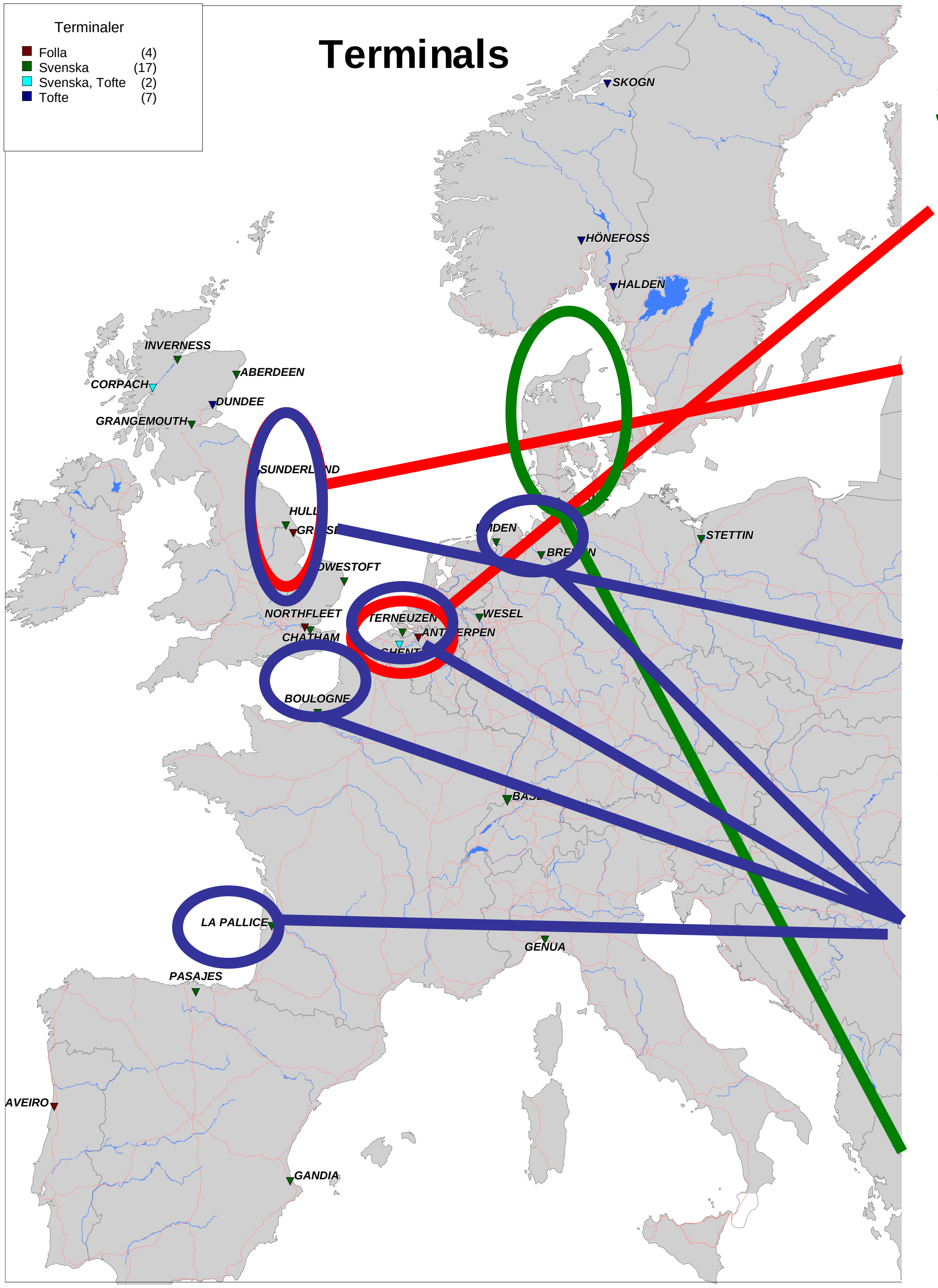
$$z_j \geq \sum_{l \in L} w_{jl} \quad \forall j \in J$$

Return flow

Problem size

Number of:

▪ Customers	262
▪ Pulpmills	4
▪ Harbor-terminals	21
▪ Land-terminals	3
▪ Ships	3
▪ Products	30
▪ A-routes	84
▪ B-routes	1,873
▪ Spot-routes	84



Scenarios

Problem P2: Terminal i Terneuzen is accessible for all customers in Italy.

Problem P3: Terminals in Sunderland and Grimsby become available for all customers in UK

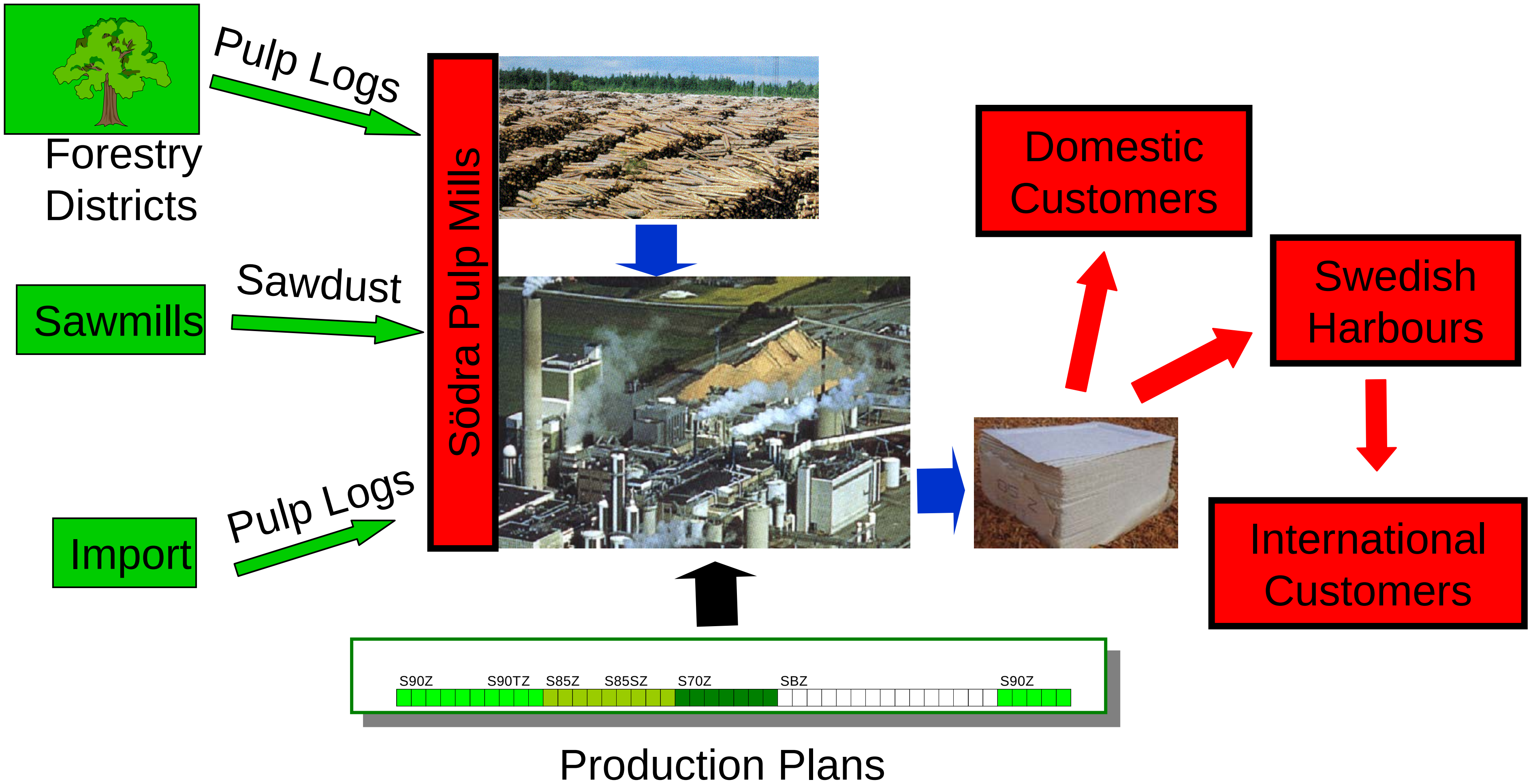
Problem P4: Problem P3 + only one of the terminals in Sunderland and Grimsby can be used

Problem P5: Terminals in Kiel Ghent, Boulogne and La Pallice are removed.

Problem P6: Alternative to the Kiel canal is tested.

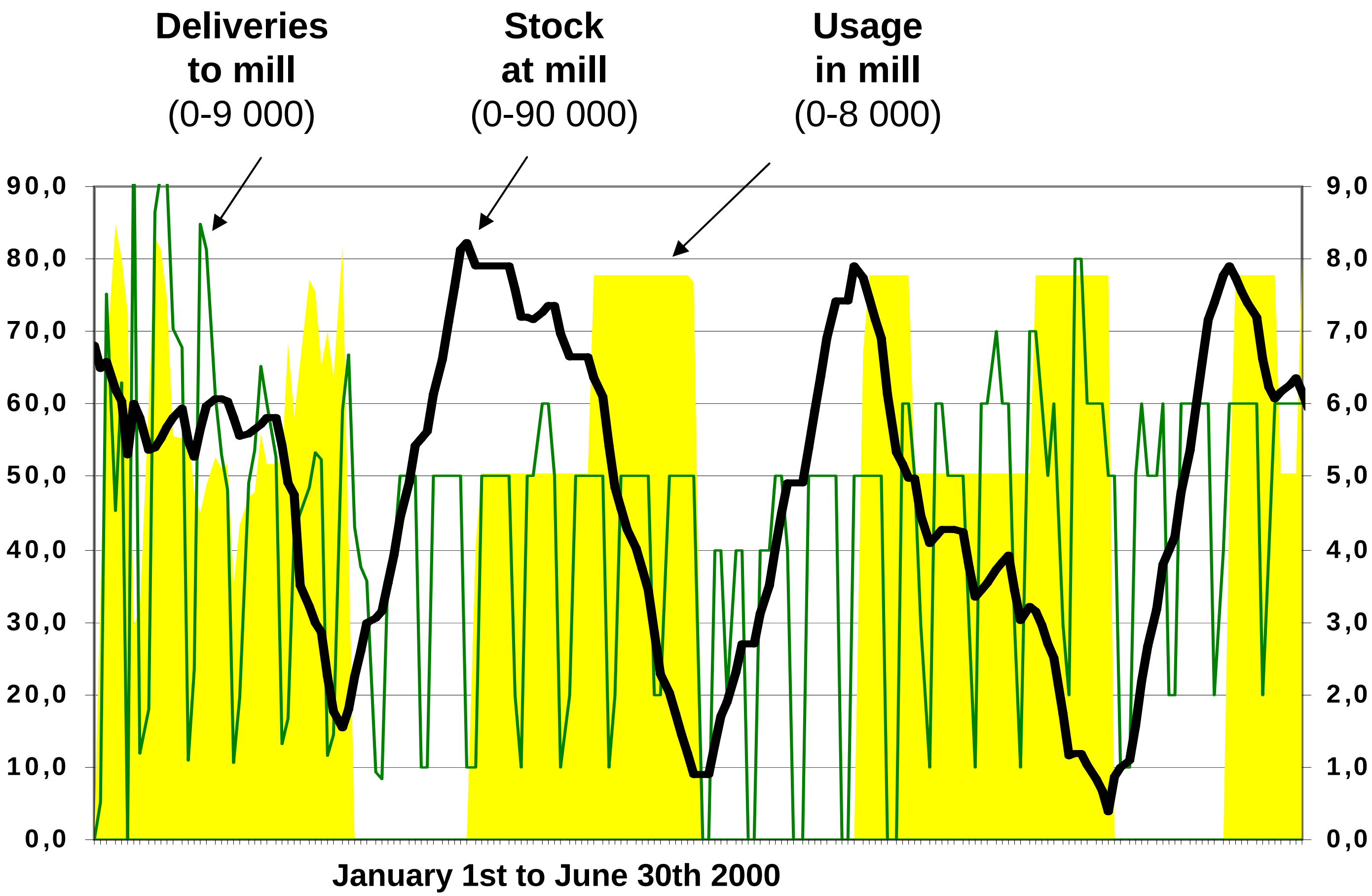
Production planning

Supply chain structure

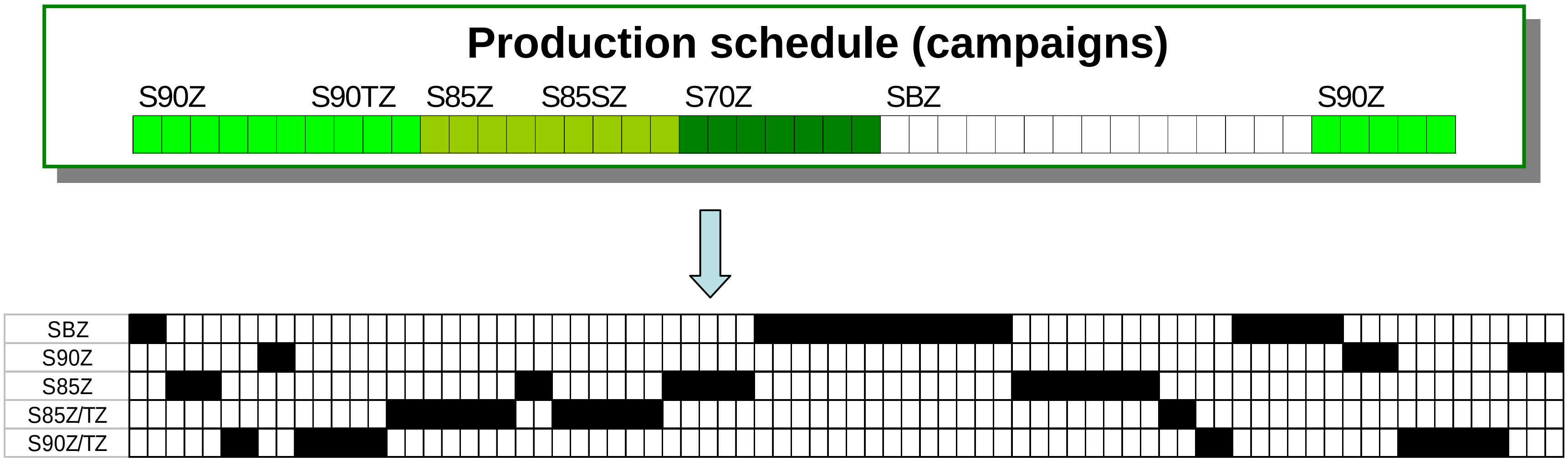


Södra Cell, Supply Chain - conditions

Daily variation due to campaign production



Solution method - column generation



Each campaign has a minimum and maximum length in time

There is a fixed cost to change campaigns i.e. recipies

Model: Constraints

$$l^F_{ia,t-1} + H_{iat} - \sum_{j \in M} x_{ijat} = l^F_{iat} \quad \forall i,a,t$$

Storage in forests

$$l^H_{jp,t-1} + w_{jpt} - v_{jpt} = l^H_{jpt} \quad \forall j,p,t$$

Storage at domestic harbours

$$\sum_{a \in A} f_{ja} \le T^M_j \quad \forall j$$

Inflow levels

$$0.9 f_{ja} \ge \sum_{i \in F} x_{ijat} \le 1.1 f_{ja} \quad \forall j,a,t$$

Inflow levels

$$\sum_{j \in M} \sum_{a \in A} x_{ijat} \le T^D_i \quad \forall i,t$$

Capacity levels in forests

$$\sum_{j \in M} y_{jdpt} = D^D_{dpt} \quad \forall d,p,t$$

Domestic demand

$$\sum_{j \in M} v_{jpt} = D^E_{pt} \quad \forall p,t$$

International demand

Model: constraints cont.

Storage of pulp logs

$$l^A_{ja,t-1} + \sum_{i \in F} x_{ijat} - \sum_{q \in Q_j} \sum_{r \in R_j} R^{in}_{jra} \delta_{jqrt} z_{jq} - l^A_{jat} = 0 \quad \forall j,a,t \qquad \text{dual: } \alpha_{jat}$$

$$l^P_{jp,t-1} + \sum_{q \in Q_j} \sum_{r \in R_j} R^{out}_{jrp} \delta_{jqrt} z_{jq} - w_{jpt} - \sum_{d \in D} y_{jdpt} - l^P_{jpt} = 0 \quad \forall j,p,t \qquad \text{dual: } \beta_{jpt}$$

Storage of pulp products

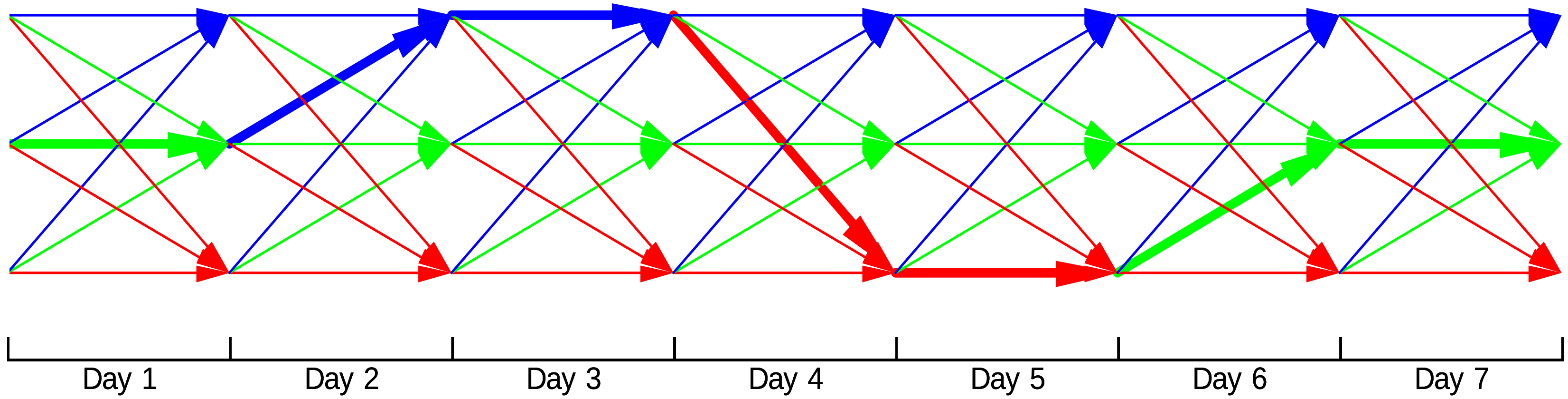
$$\sum_{q \in Q_j} z_{jq} = 1 \quad \forall j \qquad \text{dual: } \gamma_j$$

R^{in}_{jra} = amount of assortment a used in one time period when running recipe r at pulp mill j

R^{out}_{jrp} = amount of product p produced in one time period when running recipe r at pulp mill j

δ_{jqrt} = 1 if recipe r runs in production plan q during time period t at pulp mill j, 0 otherwise

Solution method - Subproblem



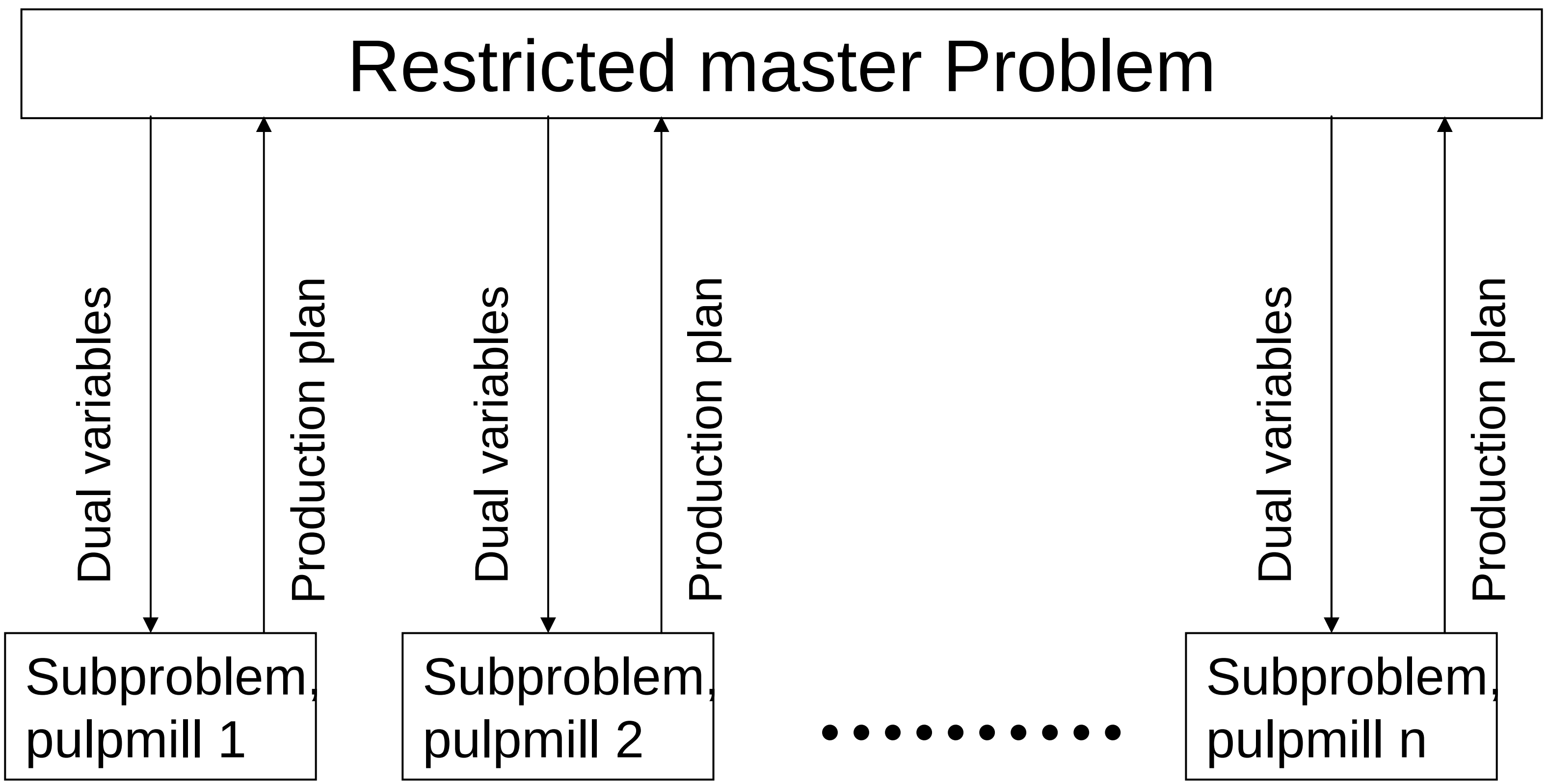
Find best reduced cost production plan

$$\overline{c_{jq}} = C_{jq}^{Z} + \sum_{a,t,r} \delta_{jqrt} R_{jra}^{in} \alpha_{jat} - \sum_{p,t,r} \delta_{jqrt} R_{jrp}^{out} \beta_{jpt} - \gamma_{j}$$

Subproblem: a shortest path problem

- Arcs represent campaigns
- Arc costs reflect dual prices on log / pulp inventory constraint and production / changeover costs

Solution method - column generation



Branch and Bound procedures

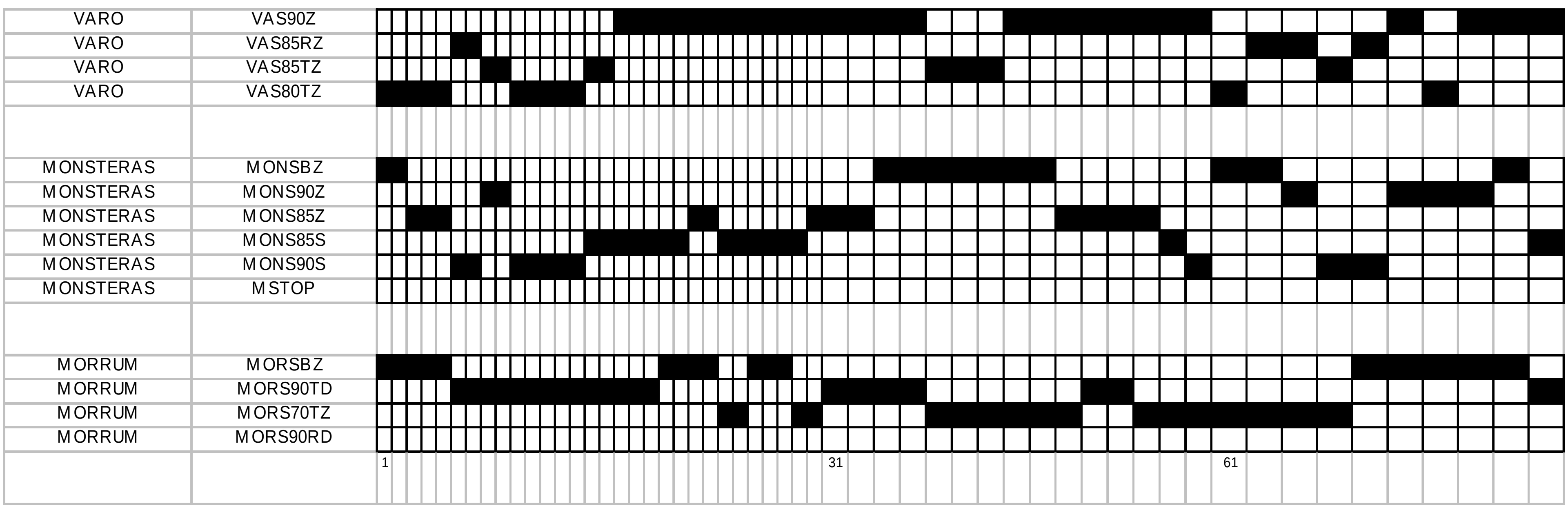
- Normal variable branching is not effective
  - There are very few 0/1-variables with value 1 (out of a very large number of possible)
  - The 1-branch is too strong (most often creating infeasible solutions)
  - The 0-branch is too weak (there are many similar production plans)
  - Procedure creates a huge Branch and Bound tree

Constraint branching heuristic

- Constraint branching enables a more efficient strategy
- Given a fractional LP-solution:
  - Sum the fractional usage of each receipe for each time period and pulp mill
  - choose the usage closest to 1.0 and branch on this
    - for example: use receipe S90Z at day 14 in mill Mönsterås
  - the branch is easy to implement in the subproblem i.e. simply remove certain arcs

Production plans

- Comparison with manual plans (campaign changes 3→5 million SEK, total 120 → 110 million SEK)
- Strategic implications regarding campaign scheduling



Home care operations

Daily planning problem

- *Address (location)*
- *Gender*
- *Language*

- *Service (medical etc..)*
- *Care time*
- *Time windows*

- *Availability*
- *Working hours*
- *Competence/ skills*

Problem in OR terms

- Decisions
  - Allocation of visits to home care workers
  - Routing of workers
- Constraints
  - Skills
  - Time windows (short and wide time windows)
  - Time relations (precedence and synchronization)
  - Working hours, travel time/ breaks
- Objective
  - Short and long term continuity
  - Route cost/ time
  - Fairness
  - Preferences

OR challenges

- Synchronisation of visits
  - Synchronized visits (double staffing)
  - Precedence relations of visits (at the same elderly)
- Translation between OR and practice
  - OR ←→ Real life (planners have no OR background)
- Multiple transport modes: Car, bike, walking
- Last minute planning
  - Many simulations require short solution times
  - Sometimes cannot assign all tasks
  - May require re-prioritization
- Evaluation of quality

OR Modeling

- Set partitioning model, extended with
  - Synchronization and precedence constraints
  - Fairness measurement

$$f^*_{SCSP} = \min \sum_{k \in K} \sum_{j \in J_k} c_{kj} z_{kj}$$

$$s.t.$$

$$\sum_{j \in J_k} z_{kj} = 1 \quad \forall k \in K$$

$$\sum_{k \in K} \sum_{j \in J_k} A_{ij} z_{kj} = 1 \quad \forall i \in N$$

$$\sum_{k \in K} \sum_{j \in J_k} (R_{i_1 j} - R_{i_2 j}) z_{kj} = 0 \quad (i_1, i_2) \in P^{sync}$$

$$z_{kj} \in \{0, 1\} \quad \forall k \in K \; \forall j \in J_k$$

THE CITY OF STOCKHOLM

Approach

- In practice locally since 2003
- Full scale implementation 2008
  - 800 Planning Officers are involved
  - All Home Care Units, about 15000 workers participate
  - 40 000 Elderly Clients enjoy the benefit
- Large scale solutions
  - E-learning programs
  - Centralized database
  - Interconnected systems to ensure information flow

Quantifiable benefits (i)

- Increased Efficiency
  - The efficiency improvement, which is calculated as more contact hours to less cost and with increased service quality, was 12 percent. For example, the ordinary staff could make 12 percent more visits in the same working time while having a better competence match and visiting the same clients.
  - Morning meeting times have decreased by two-thirds;
  - In the City of Stockholm, the time spent for developing schedules for each of the 15,000 care workers in the city’s home-help units can be lowered by an average of 10-12 minutes every day. On an annual basis, this corresponds to 310-375 full-time annual equivalents.

Quantifiable benefits (ii)

- Transportation

  In Bengtsfors the driving distance is 20 percent lower than previously

- Sick Leave

  In Jakobsberg, the annual short-term sick leave fell from 563 days to 166.

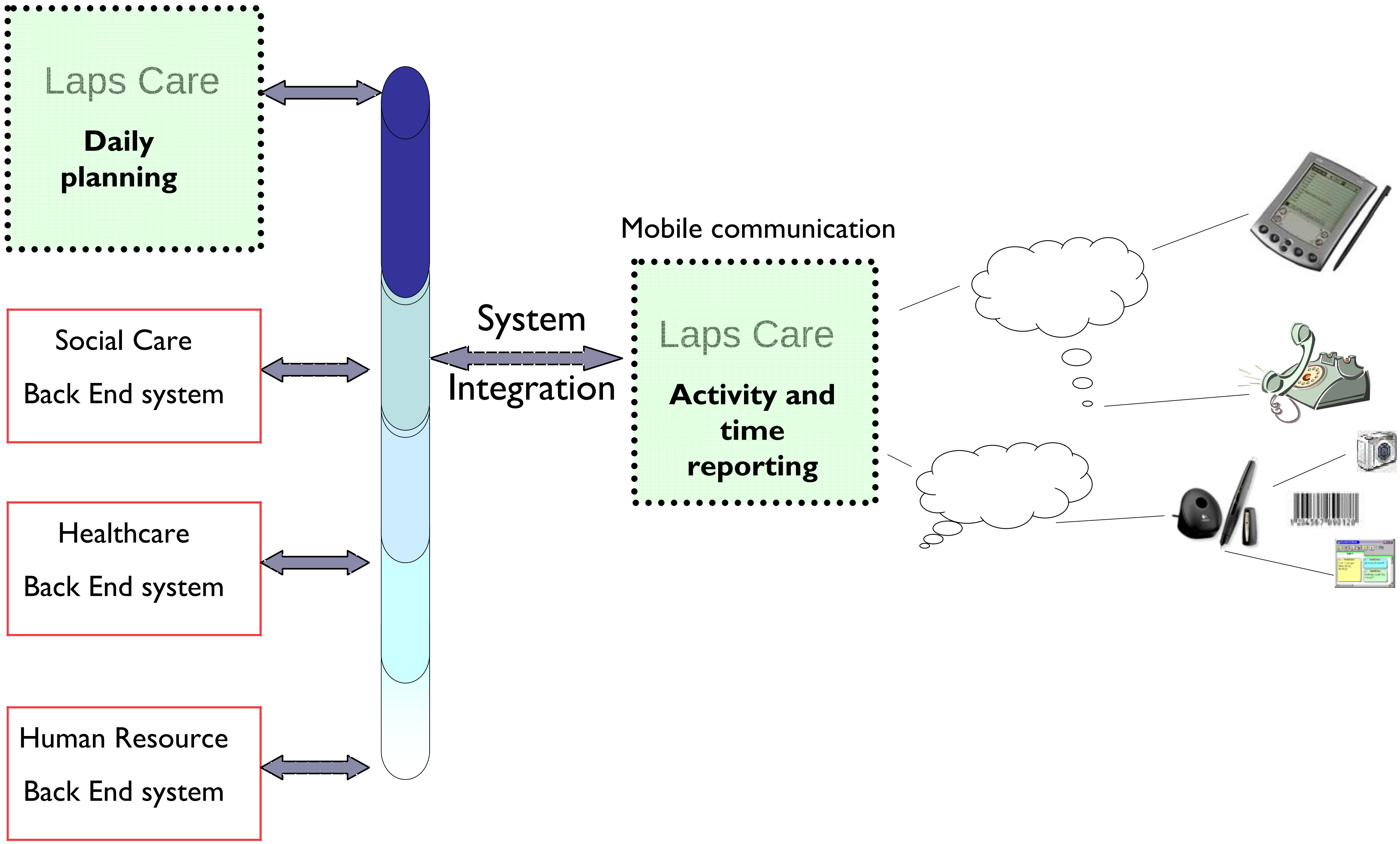
- Quality and Safety
  - In Jakobsberg, the number of missed visits (forgotten, delayed, or rebooked) fell from 91 to 4.
  - In Bengtsfors, where staff had previously had many discussions about which staff member should perform which visit, these discussions almost ceased because the system is totally objective.
  - Aurskog-Høland employs highly qualified nurses; better skill matching allows 22 percent of the nurses to be used for work requiring their specific skills.

Unquantifiable benefits (i)

- Clients
  - The risk of forgetting visits is reduced significantly.
  - Easier to control the continuity among member of staff visiting a client.
  - The reporting makes it possible to follow up, e.g., to review the overall care hours.

- Staff
  - Work in the home-care sector can be stressful. The removal of the often-turbulent last-minute morning meetings has removed one stress factor.
  - Creating routes that such that work is distributed fairly among the staff members, and that allow for realistic travel times between visits has also alleviated stress.
  - The system can also schedule work tasks, such as meetings, documentation, and administration.
  - Better usage of employee skills could also raise the status of the home-care profession.

System Overview



Laps Care: Usage

- 2007 Laps Care daily scheduled 4.000 staff in 200 units
- 2008 : 20.000 staff in 250+800 units
- 2009 : 30.000 staff in 1500 units
- Sweden has 120.000 employed in Home Care
- Northern Europe has one million Care workers

Laps Care: Business Case

- Case study proves 10% improved efficiency in cost savings and 5% in initial needed investment

  - Laps Care clients saved 2007: $US 30 million
  - Savings 2008: $US 75 million
  - Savings 2009: $US 125 million
  - Potential annual benefits in Sweden: $US 700 million
  - Potential annual benefits in N. Europe: $US 6 billion

Sudoku

Sudoku

- Given initial data
- Fill digits 1-9 into boxes such that
- Every digit 1-9 appears in every row, column, and 3x3 box

1					6	3		8
		2	3				9	
						7	1	6
7		8	9	4				2
		4				9		
9				2	5	1		4
6	2	9						
	4				7	6		
5		7	6					3

Solution

- Solver: CPLEX
- Time: 0
- Nodes: 0

… AMPL presolve

1	7	5	4	9	6	3	2	8
8	6	2	3	7	1	4	9	5
4	9	3	8	5	2	7	1	6
7	1	8	9	4	3	5	6	2
2	5	4	1	6	8	9	3	7
9	3	6	7	2	5	1	8	4
6	2	9	5	3	4	8	7	1
3	4	1	2	8	7	6	5	9
5	8	7	6	1	9	2	4	3

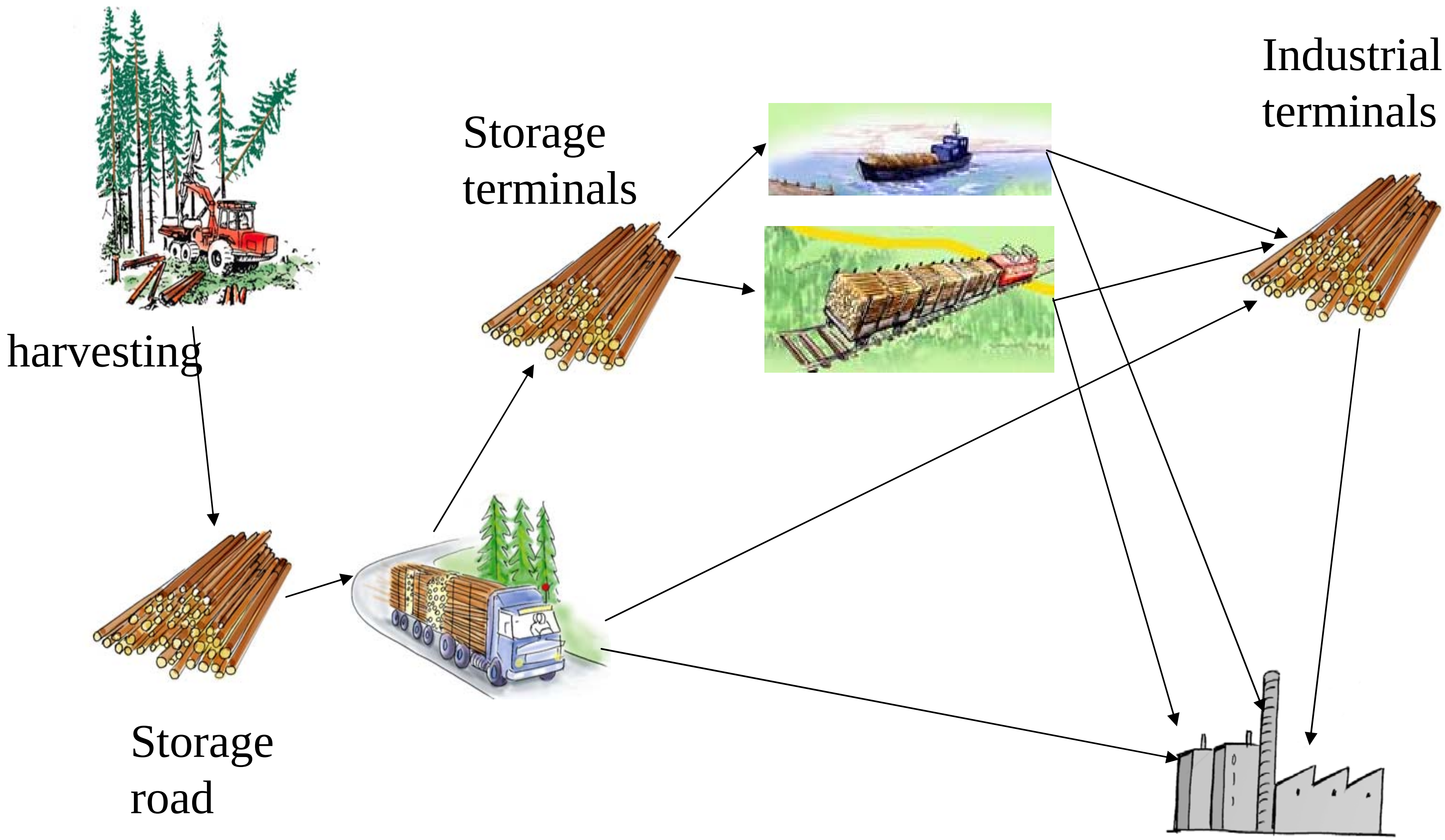
Case: Logistics planning after the storm Gudrun at Sveaskog

The storm Gudrun

- During the weekend of January 8-9 2005 the storm Gudrun (hurricane winds) hit southern part of Sweden.
- More than 70 million m^3 was blowned down; this corresponds to one full annual harvest for Sweden. Damage often at ”difficult” locations.
- Important function in the infrastructure was out of order: electricity, phones, transportation etc.
- The value of wood alone is 30 billion SEK approx. 3.2 billion Euros.
- Worst forest damages in Sweden for the last 100 years.


Storm-aid

Dangerous and slow operations


Industrial
terminals
Storage
terminals
harvesting
Storage
road

Sveaskog's logistics

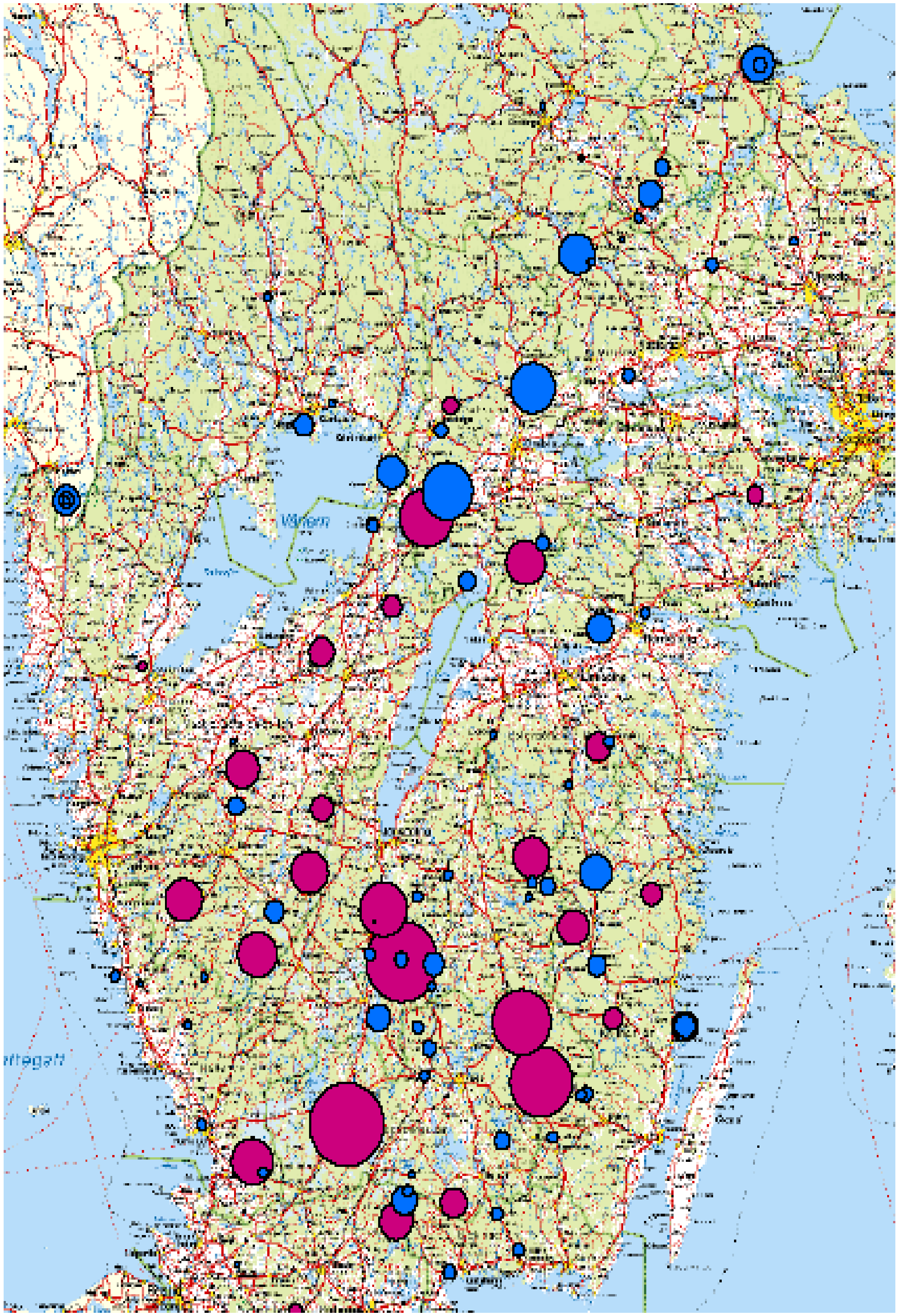
Supply

- 2,5 – 3 million m^3 own forest

  + external volumes
- not close to harbours

Demand

- Customer essentially north of the area
- Signed contracts to be followed



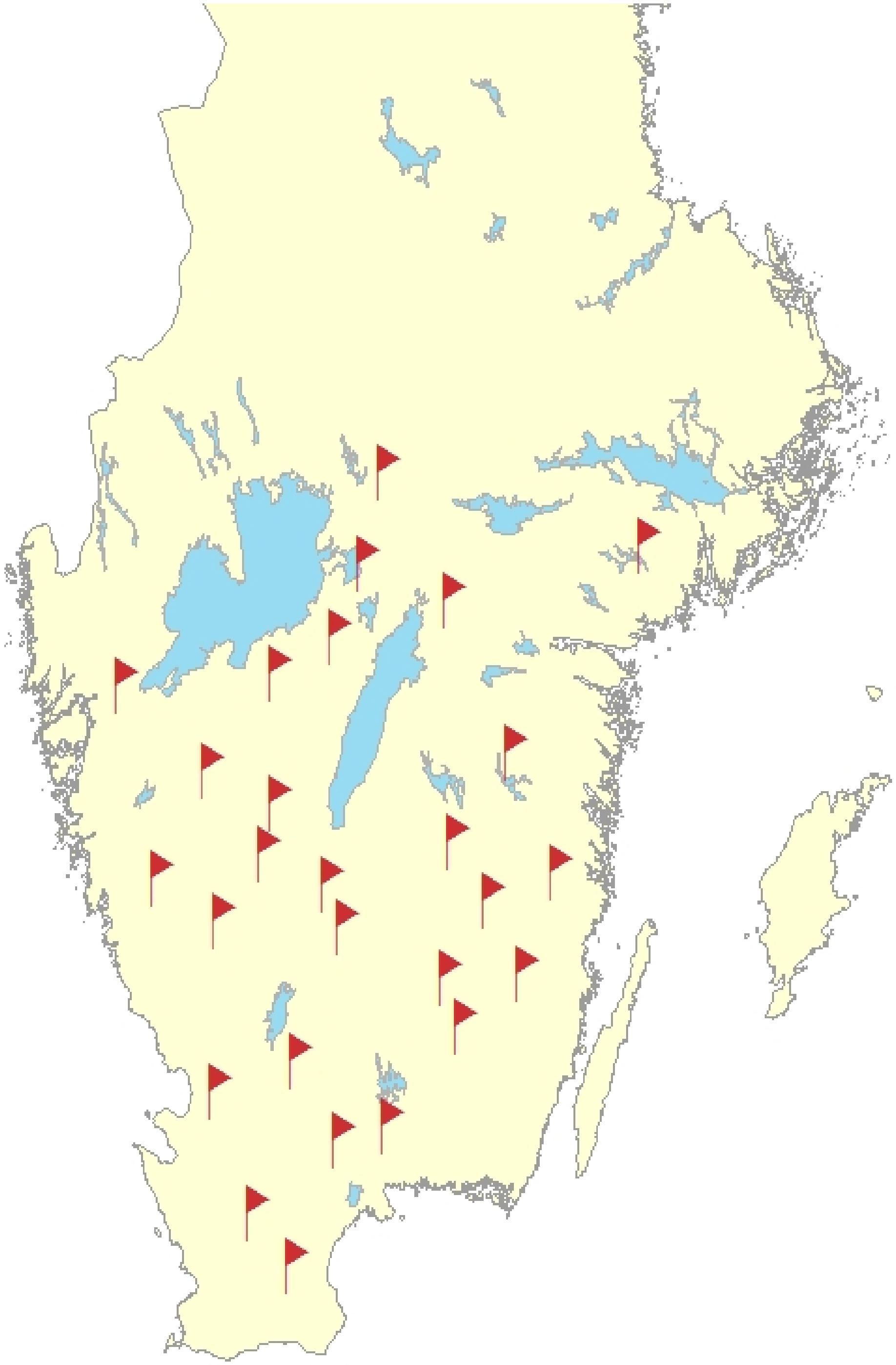
Transportation modes

Terminals


1 km
13 m

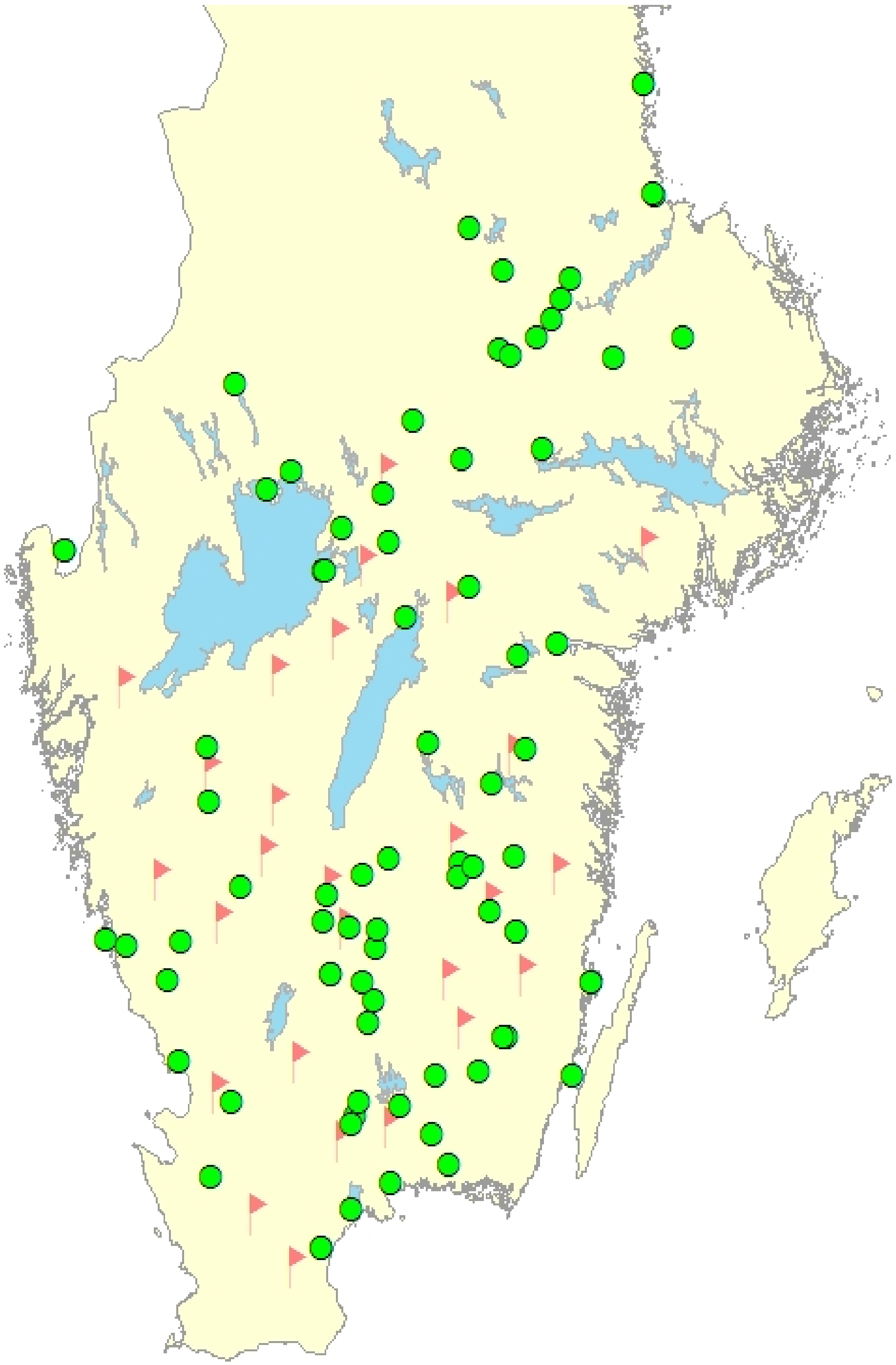
StormOpt – new decisions and restrictions

- Resource limitations
  - Trucks (ton*km)
  - Harvest capacity (different machine types)
- New decisions: harvesting, storage
  - binary variables needed
- Wood value
  - Current and new industrial orders
  - Roadside storage of logs
  - Terminal storage of logs
  - Trees not harvested
- Costs for harvesting and storage



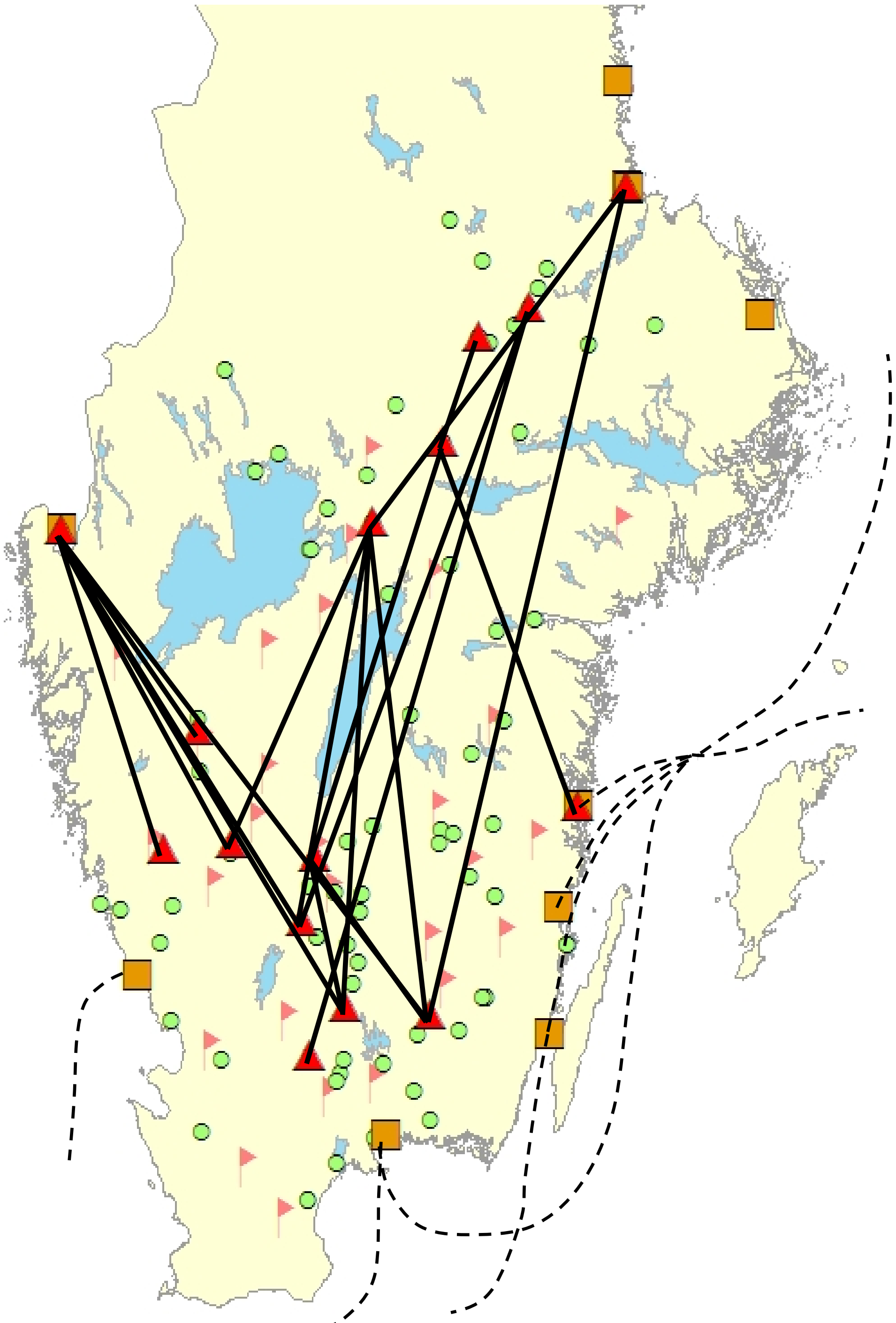
Situation

- 27 aggregated harvest areas
- 4 harvesting classes
- 5 assortment per class
- 2,5 - 3 million m^3



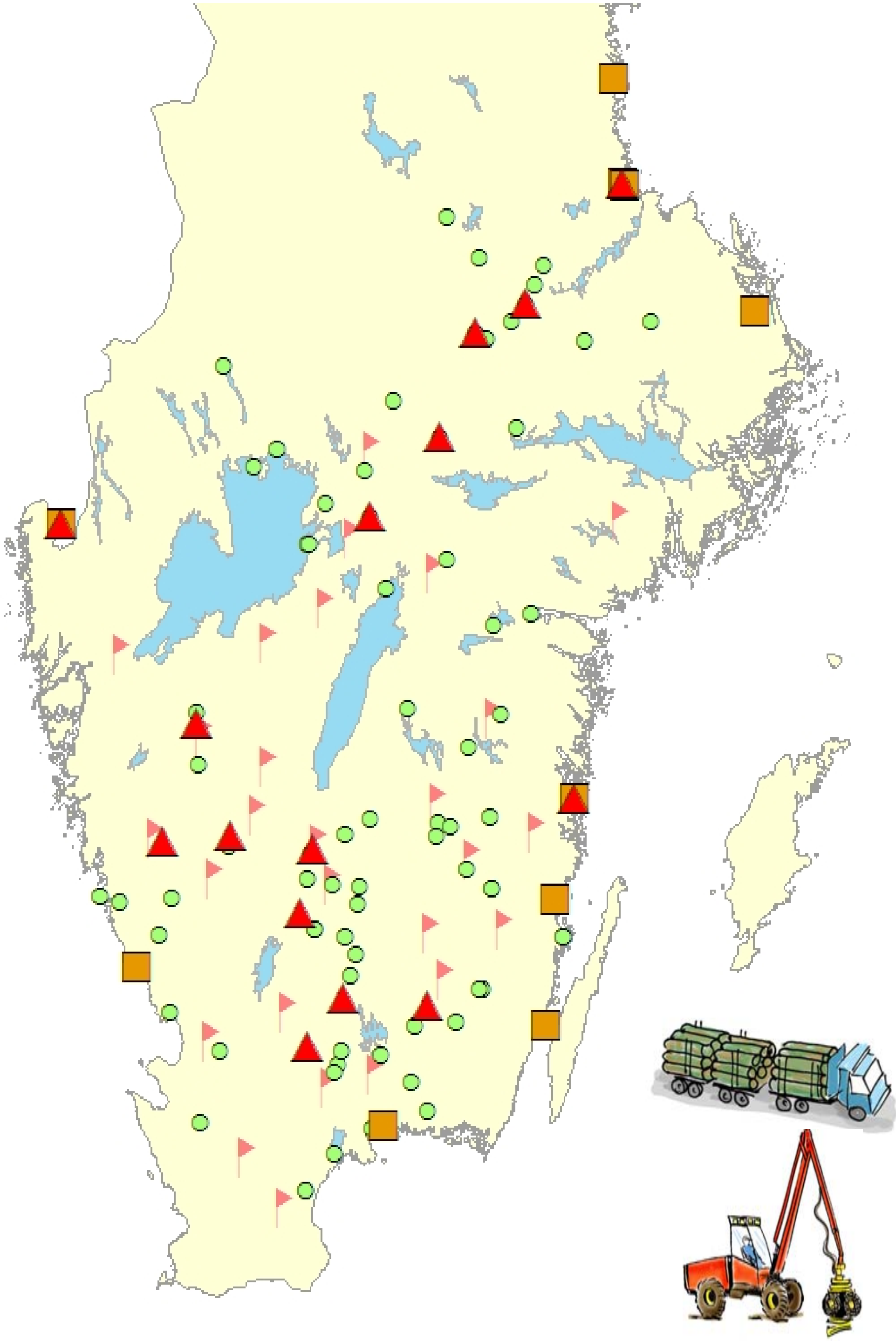
Situation

- 27 aggregated harvest areas
- 4 harvesting classes
- 5 assortment per class
- 2,5 - 3 million m^3
- 92 customers



Situation

- 27 aggregated harvest areas
- 4 harvesting classes
- 5 assortment per class
- 2,5 - 3 million m^3
- 92 customers
- 9 train terminals
- 5 harbours
- 20 train system
- 100 ship routes



Situation

- 27 aggregated harvest areas
- 4 harvesting classes
- 5 assortment per class
- 2,5 - 3 million m^3
- 92 customers
- 9 train terminals
- 5 harbours
- 20 train system
- 100 ship routes

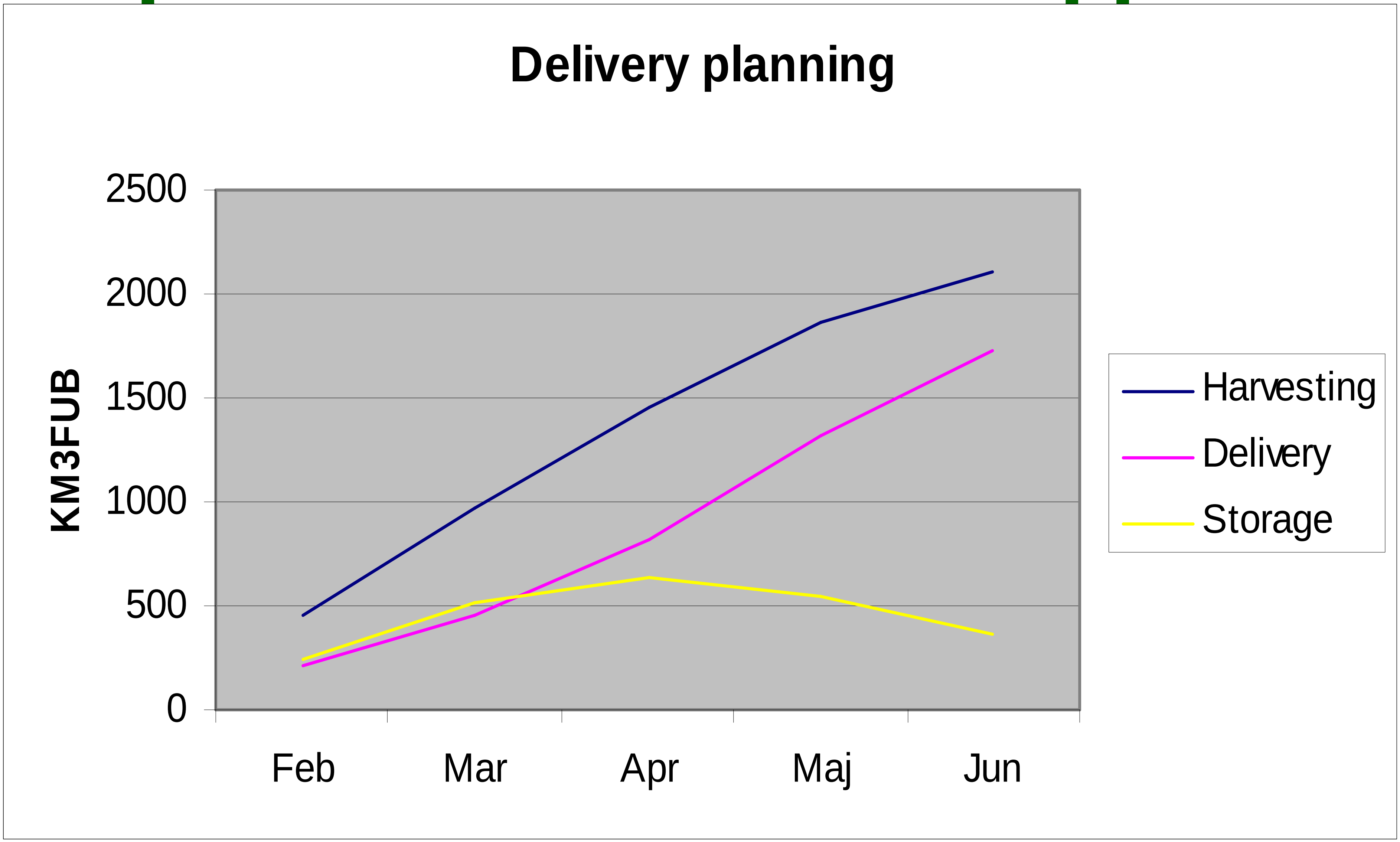
Some important factors:

- The storm felled forest for Sveaskog was not "harbour close".
- The logistics costs associated with deliveries to/from harbours were relatively high.
- Truck and transportation capacities were the most limiting factors.
- Model size:
  - 4,500 constraints
  - 195,000 variables

Experiences and what happened

- The weather conditions with deep snow made the operations difficult. When the snow depths was less, the harvest level increased rapidly.
- More medium and large units than expected arrived. Instead of 54-59-32 (large-medium-small harvest units) there were 64-74-14 units.
- Increase in the number of trucks was slower and it was not until the middle of April when there was a balance between truck and harvesting capacities (84 trucks were in operation).

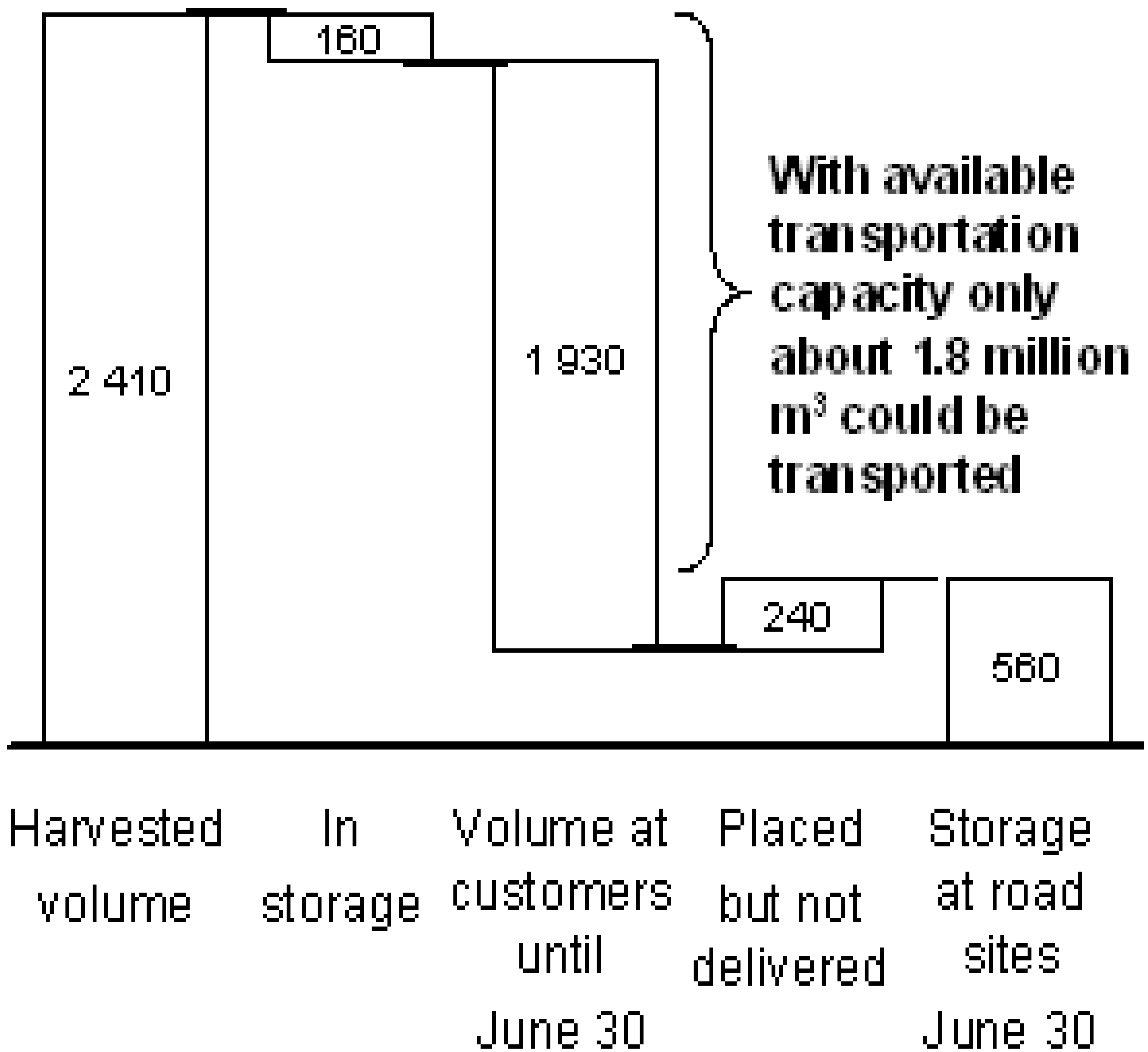
Experiences and what happened



Experiences and what happened

- The storm felled volumes were less than the estimations (measured 2,45 against estimated 3,1 million m3).
- The volume carried by train was planned to be at 369,000 m3 and this was the level in practice.
- The average distance for the trucks was 83 km (2004: 97 km). The total volume carried on trucks were 1,73 million m3.
- The transportation work on train was 47% of the total work with an average distance of 340 km. This transportation work on train represented 64 trucks.
- The volume carried by ship was smaller than planned (23,000 against 52,000 m3). One reason for this was a Finish strike.

Summary of operations



Concluding remarks

- Quick and efficient change of logistic system not possible without OR support.
- OR models easy to solve, modelling relatively easy due to ”similar” model and cost structures difficult to compute.
- Project possible with dedicated manager at Sveaskog.
- Increased acceptance of OR for non-OR persons at Sveaskog.

Concluding remarks

Concluding remarks

- There are many general and advanced solvers and modeling languages available (CPLEX, XPRESS, COIN-OR, AMPL, EXCEL, MPL, GAMS, etc) for discrete optimization.
- There are many specific solvers available for particular applications/ models (TSP, VRP, GAP, Facility location, Knapsack, etc.)
- Actual implementations require knowledge in both modeling and solution methods.
- Data is available through databases. But, care needs to be taken for error in data.
- Trends: Robustness, uncertainty, real-time, coordination, even larger and detailed models