Linear and Discrete Optimization (G54LDO)

Semester 1 of Academic Session 2017-2018

Dr Dario Landa-Silva

http://www.cs.nott.ac.uk/~pszjds/

Lecture 6 – Selection and Assignment Problems

- Need for Integer Decision Variables
  - To explain some of the situations in which integer decision variables are needed for modelling optimization problems
  - To understand combinatorial optimization as a type of discrete optimization
- Selection Optimization Problems
  - To recognise typical selection problems such as set covering
  - To model and solve various types of selection optimization problems involving binary decision variables
- Assignment Optimization Problems
  - To recognise the typical assignment problem
  - To model and solve various types of assignment optimization problems involving binary decision variables

Additional Reading

Sections 12.1 to 12.5 of (Hillier and Lieberman, 2015).

Chapters 8 and 9 of (Williams, 1999).

Siting New Fire Stations in Istanbul: A risk-based Optimization Approach. Bulent Catay. OR Insight, Vol. 24, No. 2, pp. 77-89, 2011.

A Women's Collegiate Basketball Star Player Her Her Team With OR. F.J.Vasco, R. Wisemiller, P. Gorman. OR Insight, Vol. 23, No. 2, pp. 71-81, 2010.

Classroom Exercises in IP Modeling Sudoku and the Log Pile. Martin J. Chlond. INFORMS Transactions on Education, Vol. 5, No. 2, pp. 77-79, 2005.

Need for Integer Decision Variables

- Problems where decision variables must take discrete values.
- Problems with logical conditions (e.g. produce A only if B is produced, choose at least two ingredients for the mix).
- Many cases of combinatorial optimization problems.
- Few hundreds integer decision variables can make an optimization model computationally difficult to solve.
  - When tackling large discrete optimization models, it might be more practical to build first a smaller version of the model.
  - Using reformulations can help to reduced the difficulty of the discrete optimization model.
  - Good-quality solutions in practical computational time can be obtained with heuristic search methods (e.g. evolutionary algorithms).

Discrete optimization refers in general to those optimization problems (linear or non-linear) involving discrete decision variables, i.e. of the following types:

- Integer Programming (IP)
- Binary Integer Programming (BIP)
- Mixed-Integer Programming (MIP)

In combinatorial optimization problems (COP) the goal is to find the optimal setting of a set of discrete entities such that given requirements and perhaps constraints are satisfied. The optimal setting can be an arrangement, ordering, grouping, selection, distribution, etc.

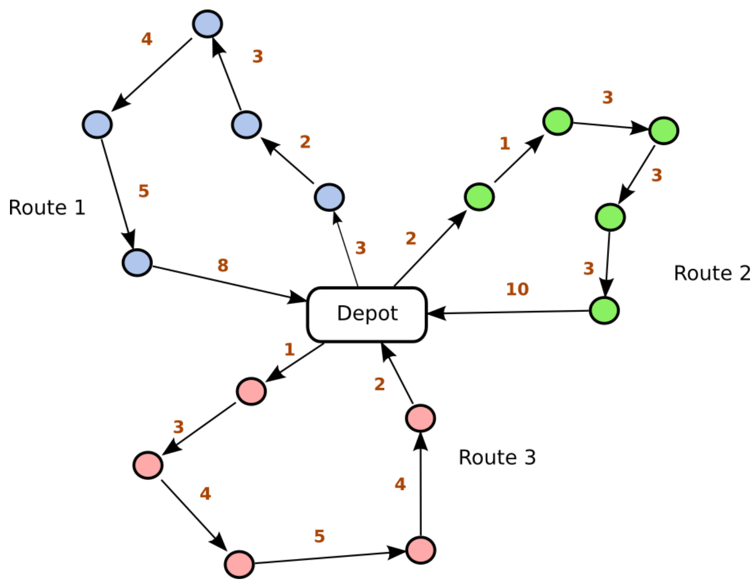
Sometimes it might not be possible to model COPs as linear models, so they have to be tackled as non-linear problems.

Travelling Salesman Problem

- discrete entities are the cities
- find optimal arrangement of cities to be visited in a tour
- minimize total travel distance

Vehicle Routing Problem

- discrete entities are the customers
- find optimal arrangement of customers into a set of routes
- each route starts and ends in the depot
- meet each customer demand
- minimize total travel distance



Timetabling Problem

- discrete entities are the events, people, timeslots, locations
- find optimal arrangement of events and people into timeslots and locations
- avoid conflicts, e.g. more than one event at the same location and at the same time
- maximize the satisfaction of preferences, e.g. avoid timetabling events on Fridays after 4pm

Monday		Tuesday		Wednesday		Thursday		Friday
Slot1	Slot2	Slot3	Slot4	Slot5	Slot6	Slot7	Slot8	Slot9
C2	C 5	PT01	PT04	C 1	C 6	PT02	C 4	PT03
PE03	PE01	PM02	PM04	C7	PE04	PM03	PE22	PM01
PM21	PBS01	C 3	PE02		PBS02	C 9	PBS04	C 8
PT24	PE21	PM24	PL04		PL03	PM23	PM22	PT22
	PL02	PT23	PBS03			PT21	PL01	
Penalty = 4		Penalty = 3		Penalty = 4		Penalty = 5		Penalty = 0
Total penalty = 16								

Sudoku Puzzle

- discrete entities are digits (0-9) and places or locations
- find optimal arrangement of digits into locations
- each 3×3 block, each column and each row must contain all the digits
- maximize the satisfaction of the puzzle constraints, i.e. finding a feasible arrangement

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

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

Selection Optimization Problems

The Set Covering Problem

Given:

A set of N items

A set of desirable features F_1, F_2, …, F_M

$C_{ij} = 1$ if item i contains or includes feature j, otherwise $C_{ij} = 0$

The problem is to select the smallest subset of items $S \subseteq N$ so that overall the desirable features are included or 'covered'.

Example. Select the smallest subset of students from a group of 8 so that there is at least one Java expert (F_1), one female (F_2), one graphic design expert (F_3) and one Chinese speaker (F_4).

Example (cont.) Sketch of set covering problem for N = 8 and M = 4.

$$
\begin{array}{c}
\begin{array}{cccc} F_1 & F_2 & F_3 & F_4 \end{array} \\
\begin{bmatrix}
c_{11} & \boxed{c_{12}} & c_{13} & c_{14} \\
c_{21} & \boxed{c_{22}} & \boxed{c_{23}} & c_{24} \\
c_{31} & c_{32} & \boxed{c_{33}} & c_{34} \\
c_{41} & \boxed{c_{42}} & c_{43} & c_{44} \\
c_{51} & c_{52} & \boxed{c_{53}} & c_{54} \\
\boxed{c_{61}} & \boxed{c_{62}} & c_{63} & c_{64} \\
c_{71} & \boxed{c_{72}} & c_{73} & c_{74} \\
c_{81} & c_{82} & c_{83} & \boxed{c_{84}}
\end{bmatrix}
\end{array}
\Rightarrow
\begin{array}{c}
\begin{array}{cccc} F_1 & F_2 & F_3 & F_4 \end{array} \\
\begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\end{array}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix}
$$

Student 6 is java expert and female

BIP Model for Set Covering Problem

$$\text{Minimize}: \quad Z = \sum_{i=1}^{N} X_i$$

$$\text{Subject to}: \quad \sum_{i=1}^{N} C_{ij} X_i \geq 1 \quad \text{for} \quad j = 1 \ldots M \qquad (1)$$

$$X_i \in \{0,1\} \quad \text{for} \quad i = 1 \ldots N \qquad (2)$$

Where:

N :

M :

C_{ij} :

X_i :

BIP model for problem instance with $N = 8$ and $M = 4$

$$\text{Minimize}: \quad Z = x_1 + x_2 + x_3 + x_4 + x_5 + x_5 + x_7 + x_8$$

$$\text{Subject to}: \quad c_{11}x_1 + c_{21}x_2 + c_{31}x_3 + c_{41}x_4 + c_{51}x_5 + c_{61}x_6 + c_{71}x_7 + c_{81}x_8 \geq 1$$

$$c_{12}x_1 + c_{22}x_2 + c_{33}x_3 + c_{42}x_4 + c_{52}x_5 + c_{62}x_6 + c_{72}x_7 + c_{82}x_8 \geq 1$$

$$c_{13}x_1 + c_{23}x_2 + c_{33}x_3 + c_{43}x_4 + c_{53}x_5 + c_{63}x_6 + c_{73}x_7 + c_{83}x_8 \geq 1$$

$$c_{14}x_1 + c_{24}x_2 + c_{34}x_3 + c_{44}x_4 + c_{54}x_5 + c_{64}x_6 + c_{74}x_7 + c_{84}x_8 \geq 1$$

$$x_i \in \{0,1\} \quad \text{for } i = 1 \ldots 8$$

Combinatorial Nature of Set Covering

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1M} \\ c_{21} & c_{22} & \cdots & c_{2M} \\ \vdots & \vdots & \cdots & \vdots \\ c_{N1} & c_{N2} & \cdots & c_{NM} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

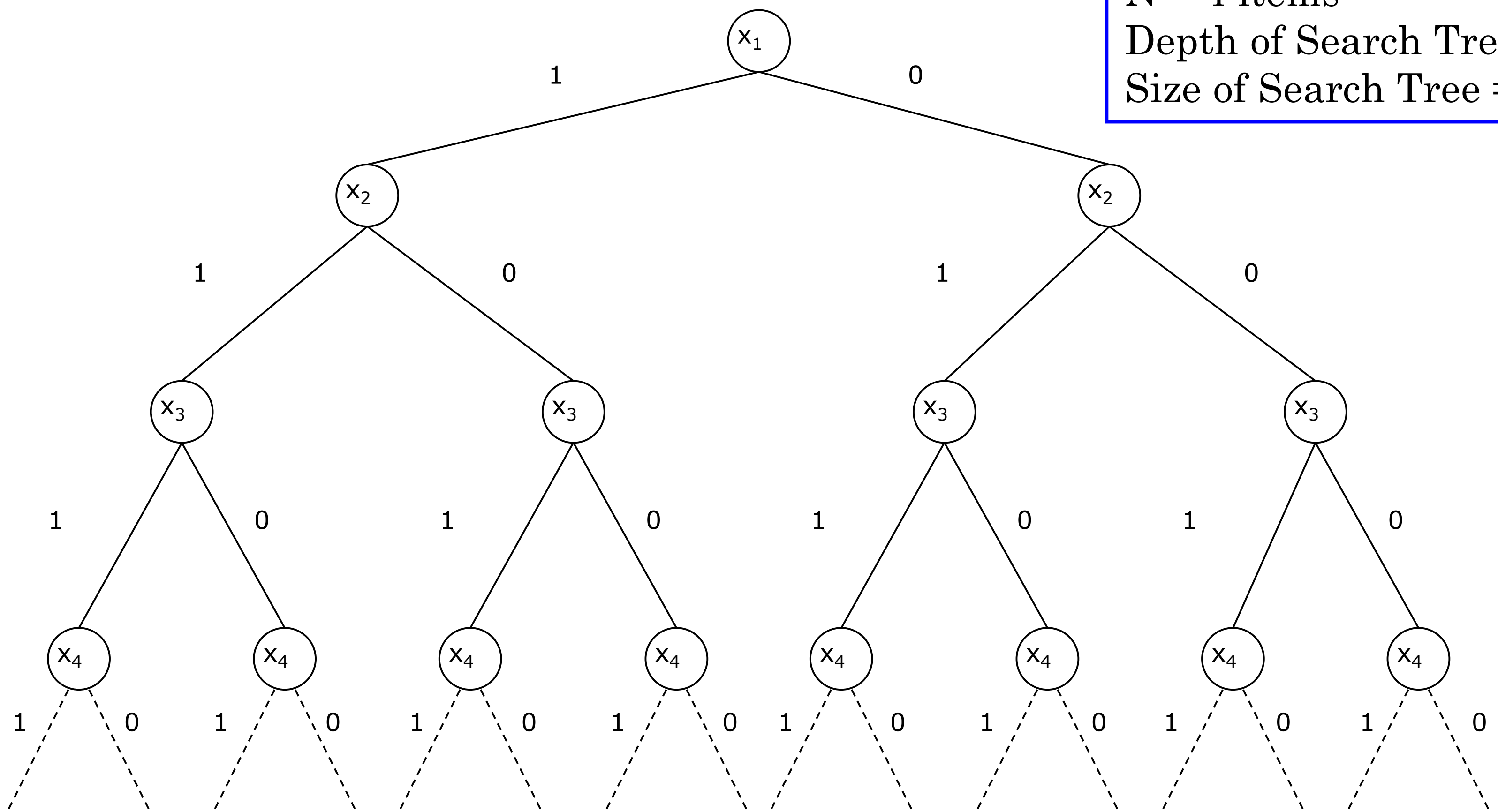
There are 2^N ways of selecting a subset from the set of N items. This includes the empty set and the whole set.

Instances of the set covering problem can be computationally difficult to solve for large values of N and M because the size of the search space increases exponentially as the size of the input increases.

There might be multiple optimal solutions, i.e. more than one subset of the same size that satisfy all the 'covering' constraints.

For a Set Covering problem instance with N = 4 decision variables X_i, the corresponding search tree is shown below. Each branch in the tree represents a solution (feasible or infeasible) and a search algorithm needs to explore this search space.

N = 4 items
Depth of Search Tree = N
Size of Search Tree = 2^N



A Selection Problem Involving Fixed-charges

Select the best mix of competing activities given limited resources. The total cost of each activity is given by the cost due to the activity level plus a fixed-charge cost.

Example.

A customer needs at least 3000 minutes of INTERNET CONNECTION every month and can choose the services of any combination from three companies A,B,C. The problem is to minimize the overall cost by deciding how many minutes (must be integer number) to contract from each company. Company A charges a £10 connection fee plus 5 cents per minute. Company B charges a £20 connection fee plus 3 cents per minute. Company C charges a £25 connection fee plus 2.5 cents per minute. Note that the connection fee is paid only when minutes are contracted.

There are conditions that potentially introduce non-linearity into the model. The cost for contracting minutes from each company is given by:

$$\text{Cost (minutes)} = \begin{cases} \text{Connection Fee} + \text{Cost Per Minute} \times \text{minutes, if minutes} > 0 \\ 0\text{, otherwise} \end{cases}$$

The optimization model can be expressed as follows:

$$\text{Minimize: } \mathrm{Z} = Cost_A + Cost_B + Cost_C$$

$$= \begin{cases} 10 + 0.05X_1\text{, if } X_1 > 0 \\ 0\text{, if } X_1 = 0 \end{cases} + \begin{cases} 20 + 0.03X_2\text{, if } X_2 > 0 \\ 0\text{, if } X_2 = 0 \end{cases} + \begin{cases} 25 + 0.025X_3\text{, if } X_3 > 0 \\ 0\text{, if } X_3 = 0 \end{cases}$$

$$\text{Subject to: } X_1 + X_2 + X_3 \geq 3000 \qquad (1)$$

$$X_i \geq 0 \text{ and integer for } i = 1 \ldots 3 \qquad (2)$$

A linear formulation for the objective function requires additional binary decision variables.

$$\text{Minimize}: \quad Z = (10Y_1 + 0.05X_1) + (20Y_2 + 0.03X_2) + (25Y_3 + 0.025X_3)$$

$$X_i \geq 0 \text{ and integer} \quad \text{for } i = 1\ldots3$$

$$Y_i \in \{0,1\} \quad \text{for } i = 1\ldots3$$

Additional linking constraints are also needed because of the additional binary indicator decision variables.

$$X_1 \leq \mathrm{M}Y_1$$
$$X_2 \leq \mathrm{M}Y_2$$
$$X_3 \leq \mathrm{M}Y_3$$

These 'linking' constraints ensure that "if the number of minutes contracted is above zero then the connection fee must be paid." M is usually large.

$$Y_1 \leq X_1$$
$$Y_2 \leq X_2$$
$$Y_3 \leq X_3$$

These 'linking' constraints ensure that "if the connection fee is paid then the number of minutes contracted must be above zero."

The full linear MIP model for the fixed-charge problem:

X_1, X_2, X_3 are minutes from company A, B, C respectively

Y_1, Y_2, Y_3 indicate if service from company A, B, C is used

$M = 3000$

$$\text{Minimize: } Z = 10Y_1 + 0.05X_1 + 20Y_2 + 0.03X_2 + 25Y_3 + 0.025X_3$$

$$\begin{aligned}
\text{Subject to: } & X_1 + X_2 + X_3 \geq 3000 && (1) \\
& X_1 \leq MY_1 && (2) \quad \text{and } Y_1 \leq X_1 \quad (2a) \\
& X_2 \leq MY_2 && (3) \quad \text{and } Y_2 \leq X_2 \quad (3a) \\
& X_3 \leq MY_3 && (4) \quad \text{and } Y_3 \leq X_3 \quad (4a) \\
& X_1, X_2, X_3 \geq 0 \text{ and integer} && (5) \\
& Y_1, Y_2, Y_3 \in \{0,1\} && (6)
\end{aligned}$$

Minimum Cost Z = 100

Example. Modified INTERNET CONNECTION problem where the objective is to maximize the number of minutes that can be purchased given a fixed budget of exactly £300. At least two companies must be used (for reliability). It is also known that no company can provide more than 5000 minutes of service.

$$\mathrm{M} = 5000$$

$$\text{Maximize:} \quad Z = X_1 + X_2 + X_3$$

$$\text{Subject to:} \quad 10Y_1 + 0.05X_1 + 20Y_2 + 0.03X_2 + 25Y_3 + 0.025X_3 = 300 \qquad (1)$$

$$X_1 \le \mathrm{M}Y_1 \quad (2) \quad \text{and} \quad Y_1 \le X_1 \quad (2a) \quad X_1 - Y_1 \ge 0 \ \text{ is equivalent to (2a)}$$

$$X_2 \le \mathrm{M}Y_2 \quad (3) \quad \text{and} \quad Y_2 \le X_2 \quad (3a) \quad X_2 - Y_2 \ge 0 \ \text{ is equivalent to (3a)}$$

$$X_3 \le \mathrm{M}Y_3 \quad (4) \quad \text{and} \quad Y_3 \le X_3 \quad (4a) \quad X_3 - Y_3 \ge 0 \ \text{ is equivalent to (4a)}$$

$$Y_1 + Y_2 + Y_3 \ge 2 \qquad (5)$$

$$X_1, X_2, X_3 \ge 0 \ \text{ and integer} \qquad (6)$$

$$Y_1, Y_2, Y_3 \in \{0,1\} \qquad (7)$$

Maximum Minutes Z = 9333

LP relaxation for the previous fixed charge problem (variant that maximizes minutes for fixed budget).

M = 5000 $\quad CF_i$ is connection fee $\quad CP_i$ is cost per minute

$$\text{Maximize}: \quad Z = \sum_{i=1}^{3} X_i$$

$$\text{Subject to}: \quad \sum_{i=1}^{3} \left(CF_i Y_i + CP_i X_i \right) = 300 \qquad (1)$$

$$X_i \le M Y_i \quad \text{for} \quad i = 1 \ldots 3 \qquad (2)$$

$$Y_i \le X_i \quad \text{for} \quad i = 1 \ldots 3 \qquad (3)$$

$$\sum_{i=1}^{3} Y_i \ge 2 \qquad (4)$$

$$X_i \ge 0 \ \text{and integer} \quad \text{for} \quad i = 1 \ldots 3 \qquad (5)$$

$$Y_i \in \{0,1\} \quad \text{for} \quad i = 1 \ldots 3 \qquad (6)$$

LP relaxation[1] solution:

$X_1 = 0.13, Y_1 = 0.13$

$X_2 = 4374.80, Y_2 = 0.87$

$X_3 = 5000, Y_3 = 1.00$

$Z = 9374.93$

LP relaxation[2] solution:

$X_1 = 0, Y_1 = 0$

$X_2 = 4333.33, Y_2 = 1$

$X_3 = 5000, Y_3 = 1$

$Z = 9333.33$

IP solution:

$X_1 = 0, Y_1 = 0, X_2 = 4335, Y_2 = 1, X_3 = 4998, Y_3 = 1$

$Z = 9333$

Assignment Optimization Problems

The Assignment Problem

Given:

A set of N tasks

Large assignment problems can be solved very efficiently with the Hungarian Algorithm

A set of N workers

C_{ij} is the cost for assigning task i to worker j

Each task must be assigned to exactly one worker and each worker must undertake exactly one task

The problem is to assign all the tasks to the workers so that the total cost is minimized.

Not all assignment problems require a 1 to 1 assignment.

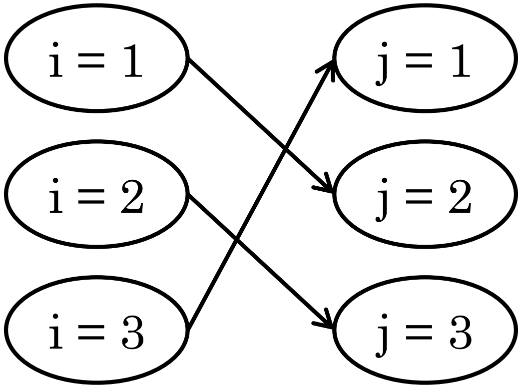
Example. Sketch of an assignment problem with N = 3.

$$\begin{array}{cc} & \text{Workers} \\ \text{Tasks} & \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \end{array}$$

$$\begin{array}{cc} & \text{Workers} \\ \text{Tasks} & \begin{bmatrix} x_{11} & \boxed{x_{12}} & x_{13} \\ x_{21} & x_{22} & \boxed{x_{23}} \\ \boxed{x_{31}} & x_{32} & x_{33} \end{bmatrix} \end{array}$$

Assume the solution is:

N = 3 tasks N = 3 workers



BIP Model for Assignment Problem

$$\text{Minimize: } Z = \sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij} X_{ij}$$

$$\text{Subject to: } \sum_{j=1}^{N} X_{ij} = 1 \quad \text{for } i = 1,2,\ldots N \quad (1)$$

$$\sum_{i=1}^{N} X_{ij} = 1 \quad \text{for } j = 1,2,\ldots N \quad (2)$$

$$X_{ij} = 1 \text{ if task } i \text{ is assigned to worker } j, 0 \text{ otherwise} \quad (3)$$

Where:

N :

C_{ij} :

X_i :

$$\text{Minimize: } Z = c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{21}x_{21} + c_{22}x_{22} + c_{23}x_{23} + c_{31}x_{31} + c_{32}x_{32} + c_{33}x_{33}$$

$$\text{Subject to: } x_{11} + x_{12} + x_{13} = 1 \quad (1)$$

$$x_{21} + x_{22} + x_{23} = 1 \quad (2)$$

$$x_{31} + x_{32} + x_{33} = 1 \quad (3)$$

$$x_{11} + x_{21} + x_{31} = 1 \quad (4)$$

$$x_{12} + x_{22} + x_{32} = 1 \quad (5)$$

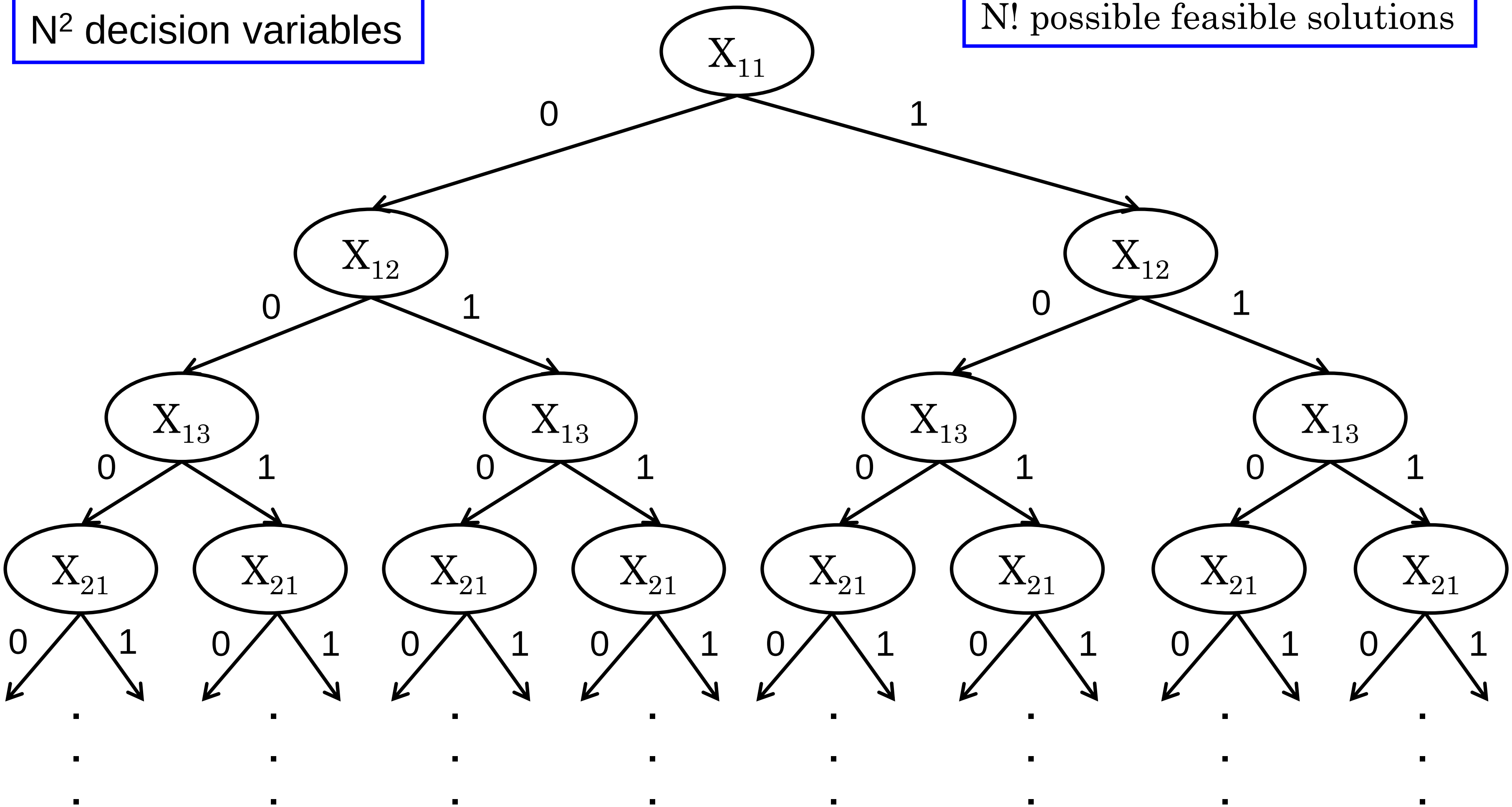
$$x_{13} + x_{23} + x_{33} = 1 \quad (6)$$

$$x_{ij} \in \{0,1\} \quad (7)$$

Combinatorial Nature of Assignment Problem

N = 3 tasks
N = 3 workers
N^2 decision variables

Depth of Search Tree = N^2
Size of Search Tree = $2^{N \times N}$
N! possible feasible solutions



Questions OR Comments