BRANCH AND BOUND METHOD

MŰEGYETEM 1782

Dr. Tibor SIPOS Ph.D.

Dr. Árpád TÖRÖK Ph.D.

Zsombor SZABÓ

2019

BME FACULTY OF TRANSPORTATION ENGINEERING AND VEHICLE ENGINEERING

32708-2/2017/INTFIN COURSE MATERIAL SUPPORTED BY EMMI

Content

Branch and Bound Method

Cnapsack Problem

General Description

- The branch and bound method originally is a method for solving binary integer, and integer programming problems
  - The simplex method cannot guarantee that the decision variables will be integers or binaries
  - A good solution could be the rounding, but that is just a heuristic solution
- The idea of the solution that each of the decision variables have only finite numbers of feasible solutions

Method

- Main step: a decision variable is chosen, and the possible ways are analysed
  - For example in binary programming what if the variable is 1, and what if it is 0
- A spanning tree is built, which trenches are one of the possible ways by setting the decision variable to a feasible value
- The spanning tree is either called solution tree or enumeration tree

Method – Steps

- Before solving the problem two initial tasks need to be accomplished
  - Actual value of the objective function $Z^* = -\infty$ or $Z^* = \infty$ must be set whether we speak about minimization or maximization
  - Solve the LP relaxation, where the binary constraints will be missing, and the new constraints (bigger or equal or lesser or equal constraints for all of the decision variables) are used
- Then comes the iteration with three steps
  - Branching
  - Bounding
  - Fathoming

Method – Branching and Bounding

- Branching: if the chosen decision variable can have n different feasible values, then n subproblems (LP relaxations) should be created
- Bounding: the LP relaxation should be solved
  - By this LP relaxation a bound is created for every feasible trenches

Method – Fathoming

- A subproblem can be fathomed if there is not any chance to find a better solution on that branch
- There are three cases where this will occur
  - The LP relaxation gives feasible solution
  - The LP relaxation cannot be solved
  - The bound is worse than the best feasible solution so far (Z^*)

Method – Fathoming

- The LP Relaxation Gives Feasible Solution
  - If the bound is better than the Z^*, then the bound should be the new Z^*
  - The subproblem can be fathomed because this is the upper bound for the whole subproblem
- The LP Relaxation Cannot Be Solved
  - There are not any feasible solutions for the constraints
- The Bound Is Worse than the Best Feasible Solution so far
  - If the branches bound is worse than the actual best feasible solution
  - The branch should be fathomed because it is the best bound for the whole branch, and the subbranches' solutions would be worse or equal

Example

- Take into consideration the next binary integer programming problem

$$\max Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$

$$6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$$

$$x_3 + x_4 \leq 1$$

$$-x_1 + x_3 \leq 0$$

$$-x_2 + x_4 \leq 0$$

$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

Initial steps

- $Z^* = -\infty$ must be set
- Solve the LP relaxation, where the binary constraints will be missing, and the new constraints ($0 \leq x_j \leq 1, \forall j$) are used
- The optimal solution for the LP relaxation is $Z\left(\frac{5}{6}, 1, 0, 1\right) = 16{,}5$
- This will be a bound for the whole problem, but because it is an integer programming, the $Z \leq 16$ will be the strictest upper bound

Solve by simplex 1.

$$\max Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$
$$6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$$
$$x_3 + x_4 \leq 1$$
$$-x_1 + x_3 \leq 0$$
$$-x_2 + x_4 \leq 0$$
$$0 \leq x_j \leq 1 \ \forall j$$

$$\max Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$
$$6x_1 + 3x_2 + 5x_3 + 2x_4 + s_1 = 10$$
$$x_3 + x_4 + s_2 = 1$$
$$-x_1 + x_3 + s_3 = 0$$
$$-x_2 + x_4 + s_4 = 0$$
$$x_1 + s_5 = 1$$
$$x_2 + s_6 = 1$$
$$x_3 + s_7 = 1$$
$$x_4 + s_8 = 1$$
$$x_j \geq 0 \ \forall j$$

Solve by simplex 2.

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
6	3	5	2	1	0	0	0	0	0	0	0	10	1,7
0	0	1	1	0	1	0	0	0	0	0	0	1	INF
-1	0	1	0	0	0	1	0	0	0	0	0	0	0
0	-1	0	1	0	0	0	1	0	0	0	0	0	INF
1	0	0	0	0	0	0	0	1	0	0	0	1	1
0	1	0	0	0	0	0	0	0	1	0	0	1	INF
0	0	1	0	0	0	0	0	0	0	1	0	1	INF
0	0	0	1	0	0	0	0	0	0	0	1	1	INF
-9	-5	-6	-4	0	0	0	0	0	0	0	0	0	

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	3	11	2	1	0	6	0	0	0	0	0	10	0,9
0	0	1	1	0	1	0	0	0	0	0	0	1	1
1	0	-1	0	0	0	-1	0	0	0	0	0	0	0
0	-1	0	1	0	0	0	1	0	0	0	0	0	INF
0	0	1	0	0	0	1	0	1	0	0	0	1	1
0	1	0	0	0	0	0	0	0	1	0	0	1	INF
0	0	1	0	0	0	0	0	0	0	1	0	1	1
0	0	0	1	0	0	0	0	0	0	0	1	1	INF
0	-5	-15	-4	0	0	-9	0	0	0	0	0	0	

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	0,3	1	0,2	0,1	0	0,5	0	0	0	0	0	0,9	5
0	-0,3	0	0,8	-0,1	1	-0,5	0	0	0	0	0	0,1	0,1
1	0,3	0	0,2	0,1	0	-0,5	0	0	0	0	0	0,9	5
0	-1	0	1	0	0	0	1	0	0	0	0	0	0
0	-0,3	0	-0,2	-0,1	0	0,5	0	1	0	0	0	0,1	-0,5
0	1	0	0	0	0	0	0	0	1	0	0	1	INF
0	-0,3	0	-0,2	-0,1	0	-0,5	0	0	0	1	0	0,1	-0,5
0	0	0	1	0	0	0	0	0	0	0	1	1	1
0	-0,9	0	-1,3	1,4	0	-0,8	0	0	0	0	0	14	

- The LP relaxation can be solved by simplex
- The columns and rows coloured by red are the pivot rows and coloumns
- Note that sometimes the 0 cannot be chosen, because it would be a degenerative step

Solve by simplex 3.

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	0,5	1	0	0,1	0	0,5	-0,2	0	0	0	0	0,9	2
0	0,5	0	0	-0,1	1	-0,5	-0,8	0	0	0	0	0,1	0,2
1	0,5	0	0	0,1	0	-0,5	-0,2	0	0	0	0	0,9	2
0	-1	0	1	0	0	0	1	0	0	0	0	0	0
0	-0,5	0	0	-0,1	0	0,5	0,2	1	0	0	0	0,1	-0,2
0	1	0	0	0	0	0	0	0	1	0	0	1	1
0	-0,5	0	0	-0,1	0	-0,5	0,2	0	0	1	0	0,1	-0,2
0	1	0	0	0	0	0	-1	0	0	0	1	1	1
0	-2,2	0	0	1,4	0	-0,8	1,3	0	0	0	0	14	

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	0	1	0	0,2	-0,8	1	0,5	0	0	0	0	0,8	0,8
0	1	0	0	-0,2	1,8	-1	-1,5	0	0	0	0	0,2	-0,2
1	0	0	0	0,2	-0,8	0	0,5	0	0	0	0	0,8	INF
0	0	0	1	-0,2	1,8	-1	-0,5	0	0	0	0	0,2	-0,2
0	0	0	0	-0,2	0,8	0	-0,5	1	0	0	0	0,2	INF
0	0	0	0	0,2	-1,8	1	1,5	0	1	0	0	0,8	0,8
0	0	0	0	-0,2	0,8	-1	-0,5	0	0	1	0	0,2	-0,2
0	0	0	0	0,2	-1,8	1	0,5	0	0	0	1	0,8	0,8
0	0	0	0	1	4	-3	-2	0	0	0	0	14	

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	0	1	0	0,2	-0,8	1	0,5	0	0	0	0	0,8	1,7
0	1	1	0	0	1	0	-1	0	0	0	0	1	-1
1	0	0	0	0,2	-0,8	0	0,5	0	0	0	0	0,8	1,7
0	0	1	1	0	1	0	0	0	0	0	0	1	INF
0	0	0	0	-0,2	0,8	0	-0,5	1	0	0	0	0,2	-0,3
0	0	-1	0	0	-1	0	1	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0	1	0	1	INF
0	0	-1	0	0	-1	0	0	0	0	0	1	0	INF
0	0	3	0	1,5	1,5	0	-0,5	0	0	0	0	16,5	

x1	x2	x3	x4	s1	s2	s3	s4	s5	s6	s7	s8	b	Ind.
0	0	1,5	0	0,2	-0,3	1	0	0	-0,5	0	0	0,8	
0	1	0	0	0	0	0	0	0	1	0	0	1	
1	0	0,5	0	0,2	-0,3	0	0	0	-0,5	0	0	0,83	
0	0	1	1	0	1	0	0	0	0	0	0	1	
0	0	-0,5	0	-0,2	0,3	0	0	1	0,5	0	0	0,2	
0	0	-1	0	0	-1	0	1	0	1	0	0	0	
0	0	1	0	0	0	0	0	0	0	1	0	1	
0	0	-1	0	0	-1	0	0	0	0	0	1	0	
0	0	2,5	0	1,5	1	0	0	0	0,5	0	0	16,5	

- In the last table, the optimal solution can be observed
- $Z\left(\frac{5}{6}, 1, 0, 1\right) = 16{,}5$

First Iteration – Branching

- In the example in the branching step two subproblems are created

$$\max Z = 5x_2 + 6x_3 + 4x_4$$

$$3x_2 + 5x_3 + 2x_4 \leq 10$$
$$x_3 + x_4 \leq 1$$
$$x_3 \leq 0$$
$$-x_2 + x_4 \leq 0$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

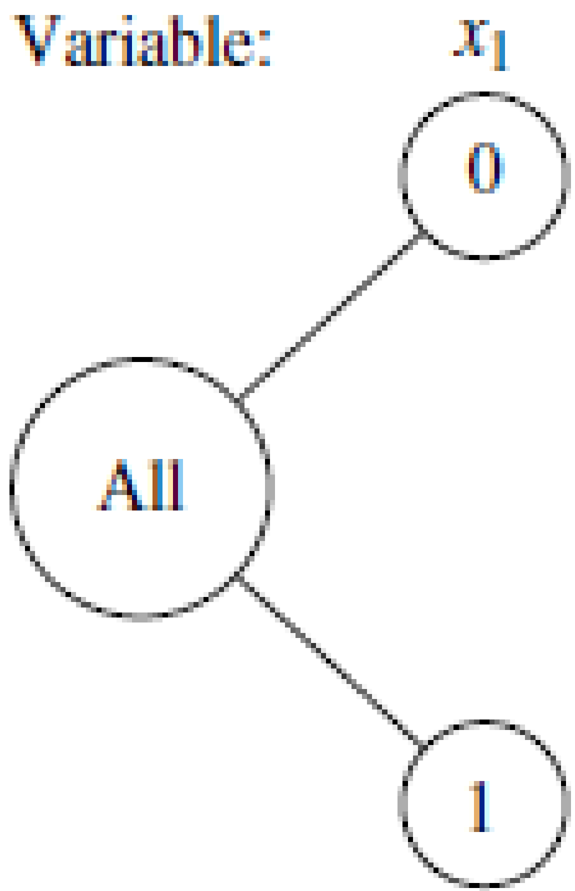
- If $x_1 = 0$

$$\max Z = 9 + 5x_2 + 6x_3 + 4x_4$$

$$3x_2 + 5x_3 + 2x_4 \leq 4$$
$$x_3 + x_4 \leq 1$$
$$x_3 \leq 1$$
$$-x_2 + x_4 \leq 0$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

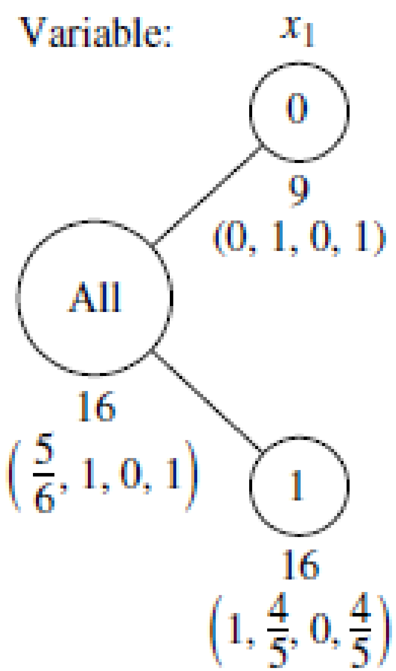
- If $x_1 = 1$

First Iteration – Branching



First Iteration – Bounding

- In this step the LP relaxation should be solved
- With the bounding, after the first iteration the next tree will occur



First Iteration – Fathoming

- If $x_1 = 0$ then a feasible solution will occur
  - $Z^* = 9$ should be set
  - The branch should be fathomed
- The only feasible way is $x_1 = 1$

Second Iteration – Branching

- Two subproblems are created

$$\max Z = 9 + 6x_3 + 4x_4$$

$$5x_3 + 2x_4 \leq 4$$
$$x_3 + x_4 \leq 1$$
$$x_3 \leq 1$$
$$x_4 \leq 0$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

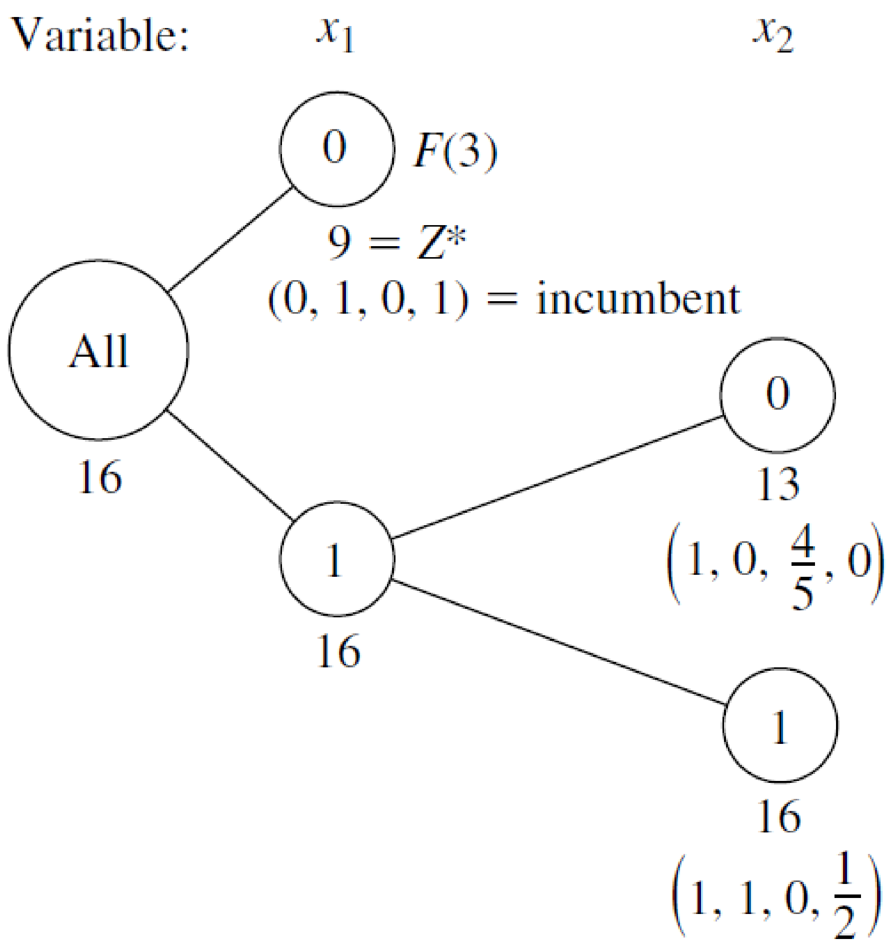
- If $x_2 = 0$

$$\max Z = 14 + 6x_3 + 4x_4$$

$$5x_3 + 2x_4 \leq 1$$
$$x_3 + x_4 \leq 1$$
$$x_3 \leq 1$$
$$x_4 \leq 1$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

- If $x_2 = 1$

Second Iteration – Bounding



Second Iteration – Fathoming

- All of the ways are feasible
- The best feasible way is $x_2 = 1$ so that will be used in the next

Third Iteration – Branching

- Two subproblems are created

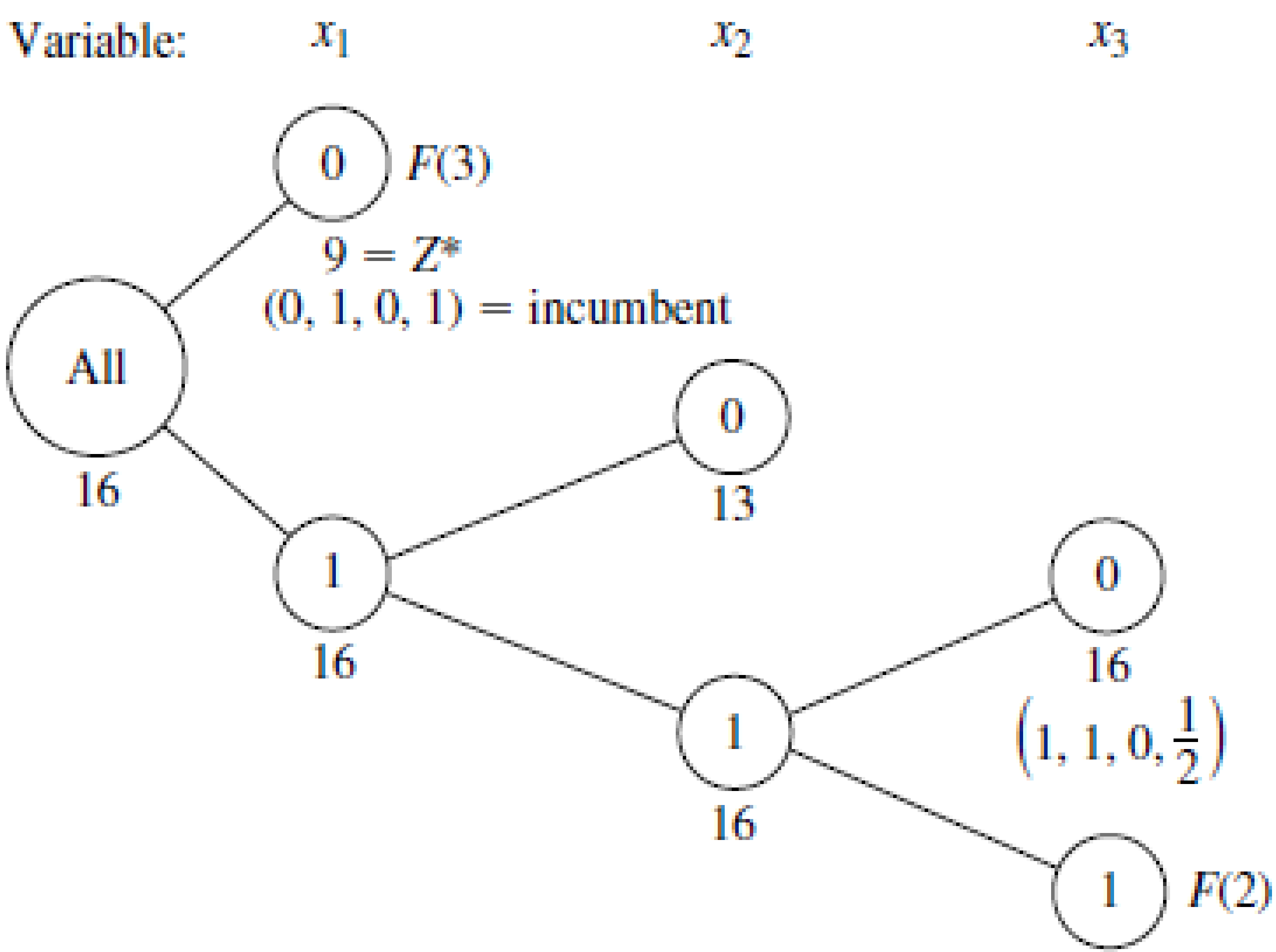
$$\max Z = 14 + 4x_4$$
$$2x_4 \leq 1$$
$$x_4 \leq 1$$
$$x_4 \leq 1$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

- If $x_3 = 0$

$$\max Z = 20 + 4x_4$$
$$2x_4 \leq -4$$
$$x_4 \leq 0$$
$$x_4 \leq 1$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

- If $x_3 = 1$

Third Iteration – Bounding



Third Iteration – Fathoming

- If $x_3 = 1$ then there are not any feasible solutions
  - The branch should be fathomed
- The only feasible way is $x_3 = 0$

Fourth Iteration – Branching

- Two subproblems are created

$$\max Z = 14$$

$$0 \leq 1$$
$$0 \leq 1$$
$$0 \leq 1$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

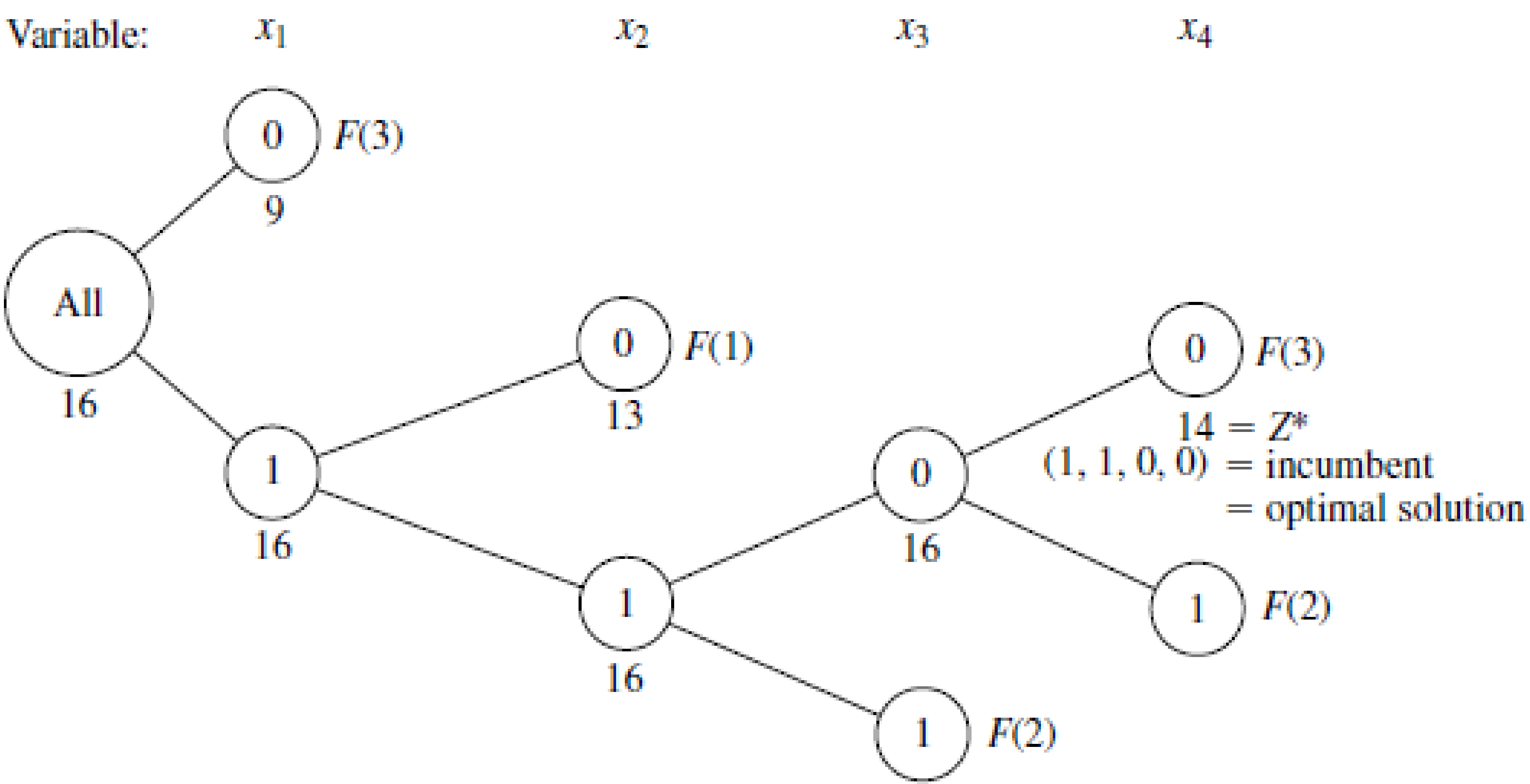
- If $x_4 = 0$

$$\max Z = 18$$

$$0 \leq -1$$
$$1 \leq 1$$
$$1 \leq 1$$
$$x_j = \begin{cases} 0 \\ 1 \end{cases}, \forall j$$

- If $x_4 = 1$

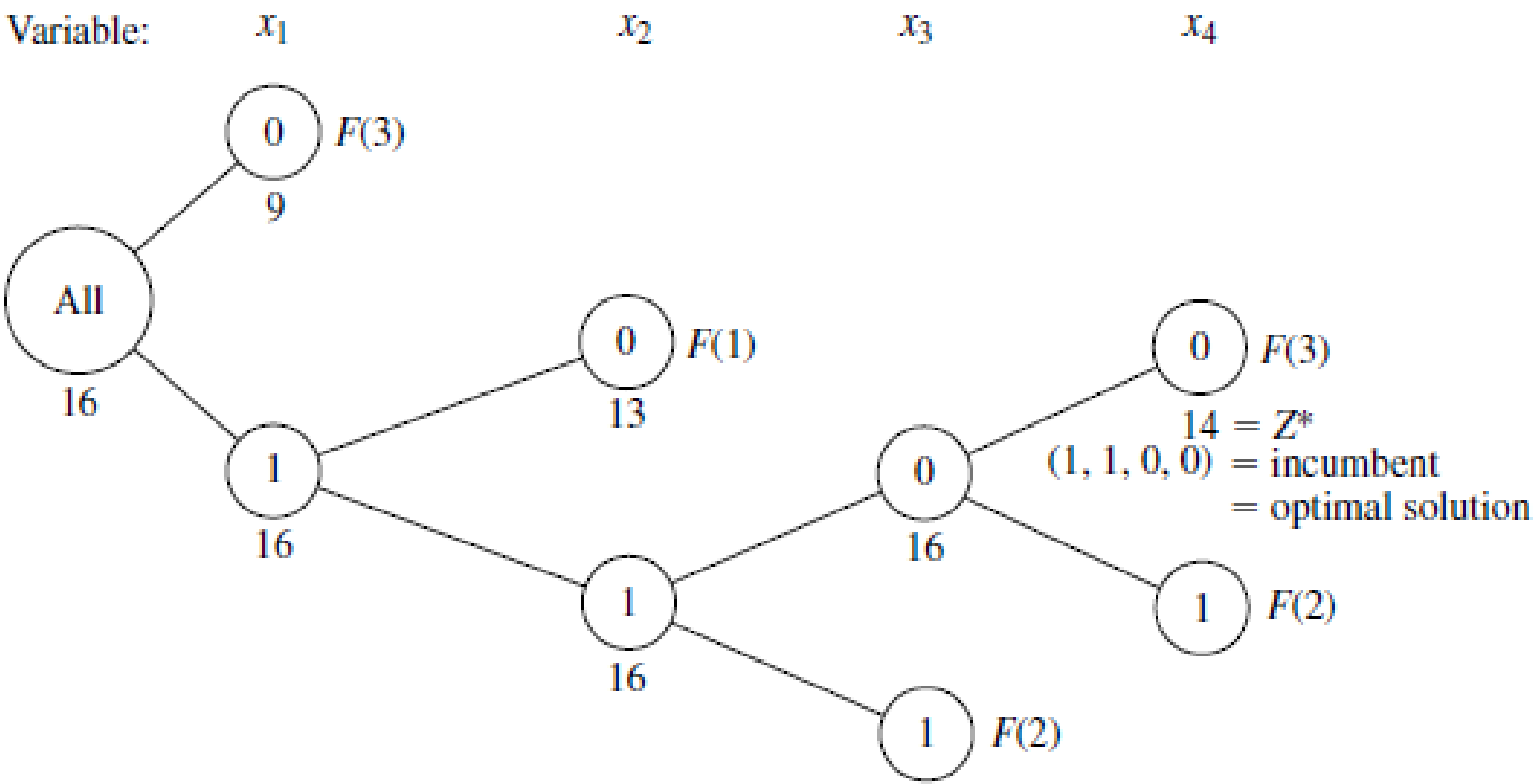
Fourth Iteration – Bounding



Fourth Iteration – Fathoming

- If $x_4 = 1$ then there are not any feasible solutions
  - The branch should be fathomed
- The only feasible way is $x_4 = 0$ which is a feasible solution
  - $Z^* = 14$
  - $x_2 = 0$ branch should be fathomed because its bound is worse than $Z^* = 14$
- All of the possibilities are analysed, so $Z^* = 14$ is the optimal solution with $Z(1, 1, 0, 0)$

Optimal Solution



Content

Branch and Bound Method

Cnapsack problem

Original problem

- A tourist would like to go for hiking
  - There are some useful stuff
    - Value (c_j)
    - Mass (a_j)
  - The backpack has a bound (how many kilograms of tools fit in) (b)
  - Each tool has a decision variable (x_j)

$$\max \sum_j c_j x_j$$

$$\sum_j a_j x_j \leq b$$

$$x_j = \begin{cases} 0 \\ 1 \end{cases} \forall j$$

Effect of the positiveness

- The feasible set is connected
  - A binary vector solution is given ($\boldsymbol{a}^T \boldsymbol{y} \leq b$)
  - If another binary vector $\boldsymbol{z} < \boldsymbol{y}$ is given, $\boldsymbol{z}$ also satisfies the problem
  - If a $\boldsymbol{z} < \boldsymbol{x} < \boldsymbol{y}$ vector is given and $\boldsymbol{y}$ and $\boldsymbol{z}$ vectors are solutions, then $\boldsymbol{x}$ is also a feasible solution
- All of the possible items shoud be bringed
- These properties implies the usage of the greedy algorithm

Solution method

- The items should be ordered based upon their unit cost on mass
- All of the possible items shold be packed until the bound is reached (Lower bound)
- Then the partition of the next item should be packed to satisfy the bound equity (Upper bound)
- The branching step is on the divided item

BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS

Dr. Tibor.SIPOS Ph.D.
Dr. Árpád TÖRÖK Ph.D.
Zsombor SZABÓ

MŰEGYETEM 1782

email: szabo.zsombor@mail.bme.hu

BME FACULTY OF TRANSPORTATION ENGINEERING AND VEHICLE ENGINEERING
32708-2/2017/INTFIN COURSE MATERIAL SUPPORTED BY EMMI

Source of the Images, and the Example

- Hillier F. S, Lieberman G. J. Integer Programming. In: Hillier F. S, Lieberman G. J. Introduction to Operations Research. 7th ed. New York: McGrow-Hill; 2001. p. 576-653. ISBN: 0-07-232169-5