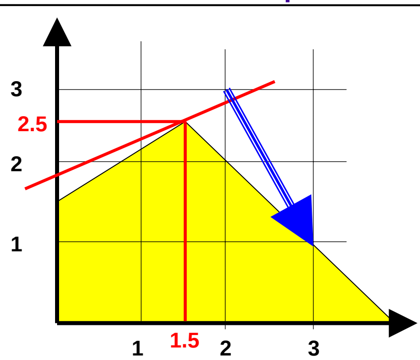
Lecture 6¾: branch and bound revisited

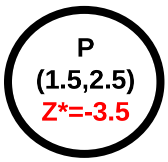
- Divide an conquer
- Fathoming tests
- Generic branch and bound algorithm
- Another fully worked out example

[Winston, Introduction to mathematical programming, Chap. 9, pp.515-524]
[Bertsimas and Tsitsiklis, Introduction to Linear Optimization, Chap. 11, sec. 11.2, pp. 485-490]

Divide and conquer



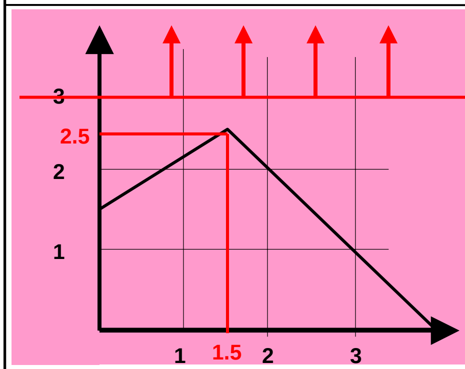
$$\begin{aligned} \textbf{min:} \quad & x_1 - 2x_2 \\ \textbf{s.t.} \quad & -4x_1 + 6x_2 \leq 9 \\ & x_1 + x_2 \leq 4 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \\ & x_1,\ x_2 \text{ integer} \end{aligned}$$

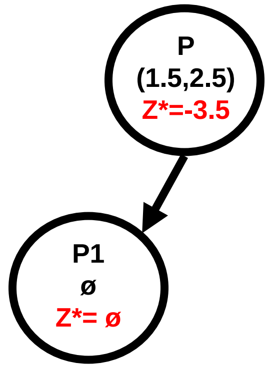


At every step: divide and conquer:

Division is done around the optimum of the LP relaxation solution,

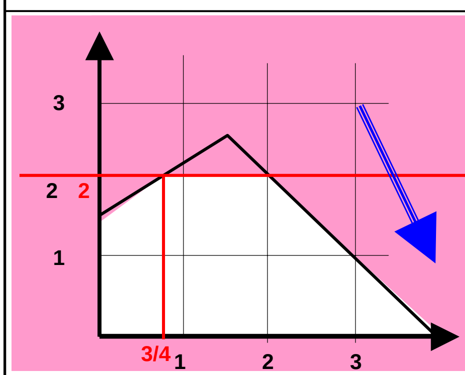
Branch and bound: fathoming tests

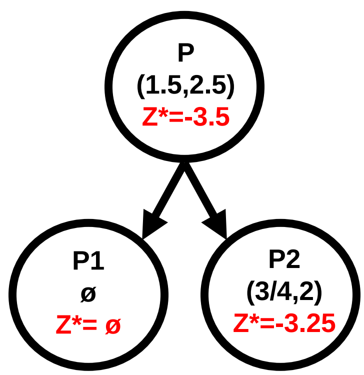




$$
\begin{array}{ll}
\textbf{min:} & x_1 - 2x_2 \\
\textbf{s.t.} & -4x_1 + 6x_2 \leq 9 \\
& x_1 + x_2 \leq 4 \\
& x_1 \geq 0 \\
& x_2 \geq 0 \\
& x_1,\ x_2 \text{ integer}
\end{array}
$$

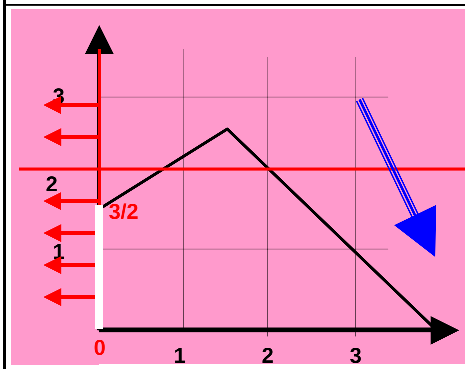
Branch and bound: fathoming tests



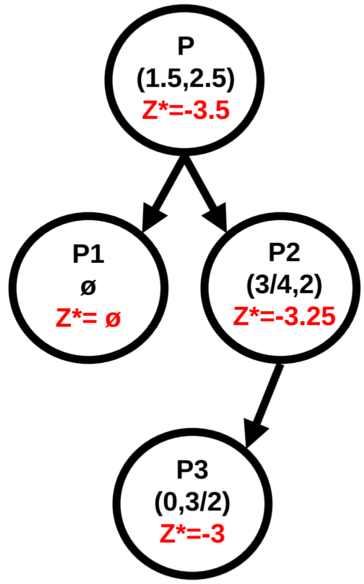


$$
\begin{array}{ll}
\textbf{min:} & x_1 - 2x_2 \\
\textbf{s.t.} & -4x_1 + 6x_2 \leq 9 \\
& x_1 + x_2 \leq 4 \\
& x_1 \geq 0 \\
& x_2 \geq 0 \\
& x_1,\ x_2 \text{ integer}
\end{array}
$$

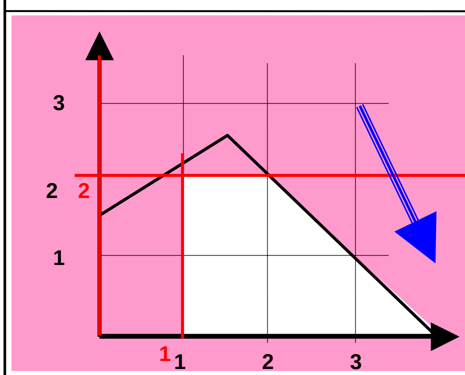
Branch and bound: fathoming tests



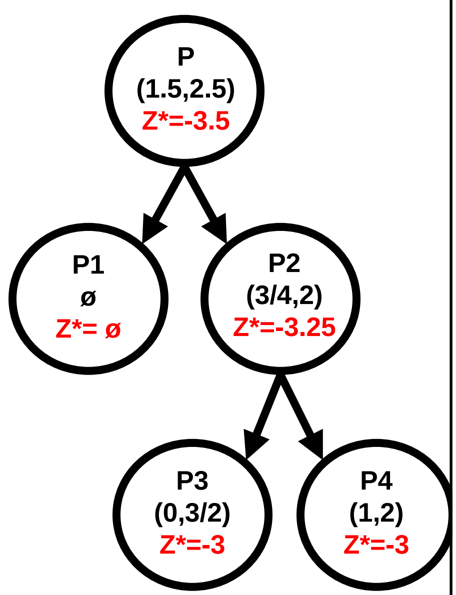
$$\begin{aligned} \textbf{min:} \quad & x_1 - 2x_2 \\ \textbf{s.t.} \quad & -4x_1 + 6x_2 \le 9 \\ & x_1 + x_2 \le 4 \\ & x_1 \ge 0 \\ & x_2 \ge 0 \\ & x_1,\ x_2 \text{ integer} \end{aligned}$$



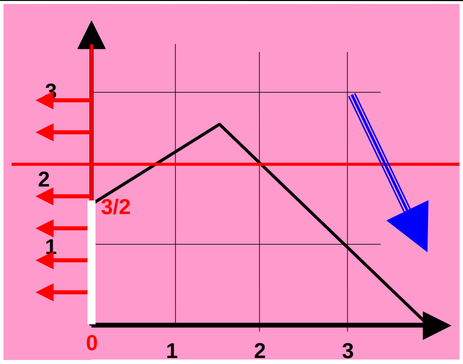
Branch and bound: fathoming tests

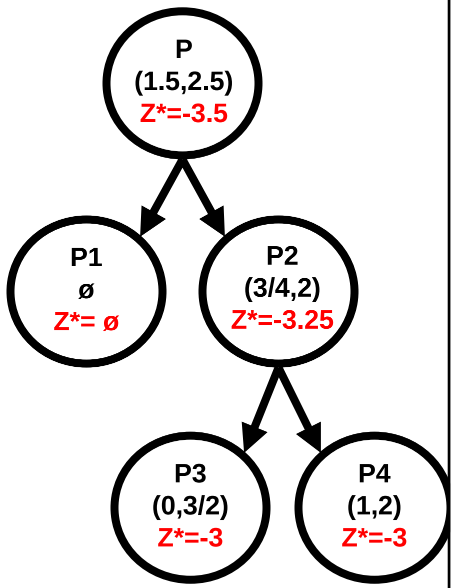


$$\begin{aligned} \textbf{min:} \quad & x_1 - 2x_2 \\ \textbf{s.t.} \quad & -4x_1 + 6x_2 \le 9 \\ & x_1 + x_2 \le 4 \\ & x_1 \ge 0 \\ & x_2 \ge 0 \\ & x_1,\ x_2 \text{ integer} \end{aligned}$$



Branch and bound: fathoming tests

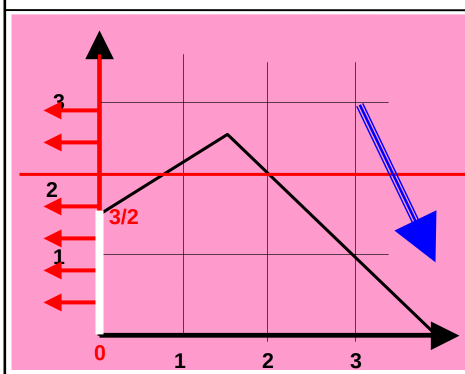


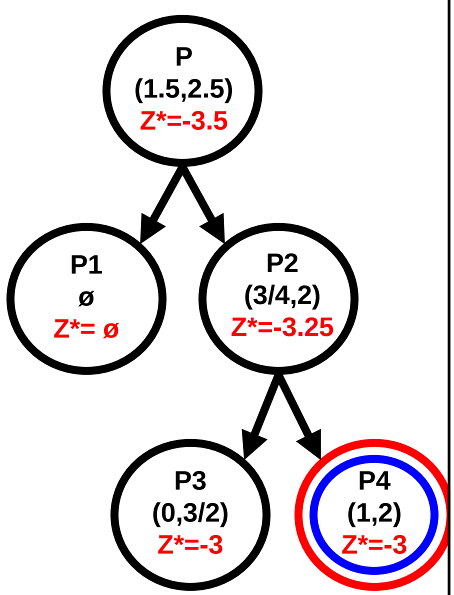


Question: should one discard problemP3 right away?

Answer: No, unless one can show that no better solution is obtainable from P3

Branch and bound: fathoming tests





Question: should one discard problemP3 right away?

Answer: **YES**

Fathoming tests

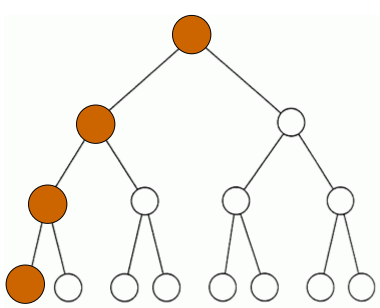
1. Subproblem is infeasible: discard
2. Subproblem has integer solution
   - Stop branching
   - Keep the value for future comparisons
3. Subproblem has an optimum below the optimum provided by the other branch → discard problem

Note: it is sometimes not possible to tell right away if case 3 enables to discard a problem until later in the algorithm.

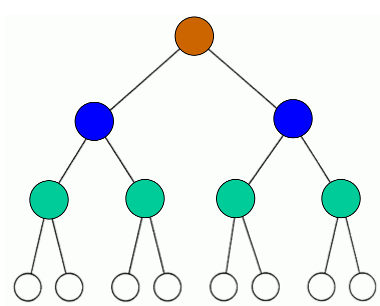
Construction of the tree

Both approaches are equally valid. You have the choice between the two (or any other mixed approach)

Depth first



Breadth first



Branch and bound algorithm (maximization)

Initialization: relax the initial problem

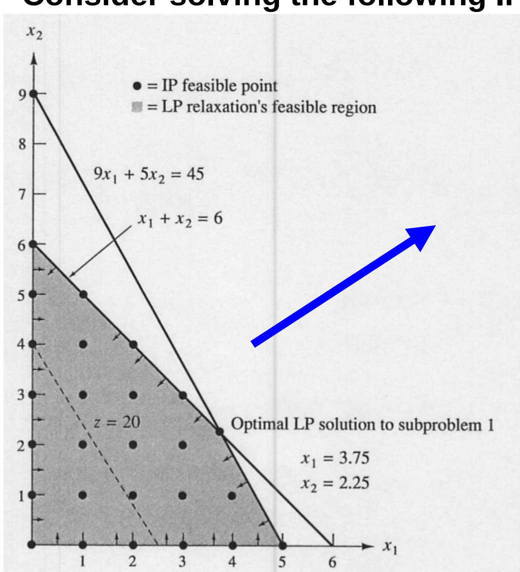
Steps for each iteration

- **Solve the relaxed LP**
- **Branching: among the unfathomed problems, branch next subproblem, by dividing around the fractional solution**
- **Fathom the problems (if possible)**

Optimality test: stop when there is no remaining subproblems

Example

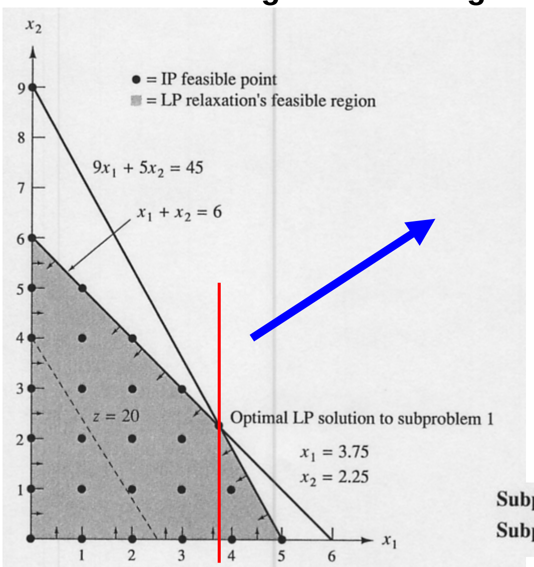
Consider solving the following IP:



$$\max z = 8x_1 + 5x_2$$
$$\text{s.t.} \quad x_1 + x_2 \le 6$$
$$9x_1 + 5x_2 \le 45$$
$$x_1, x_2 \ge 0;\ x_1, x_2 \text{ integer}$$

Example (iteration 1)

Consider solving the following IP:

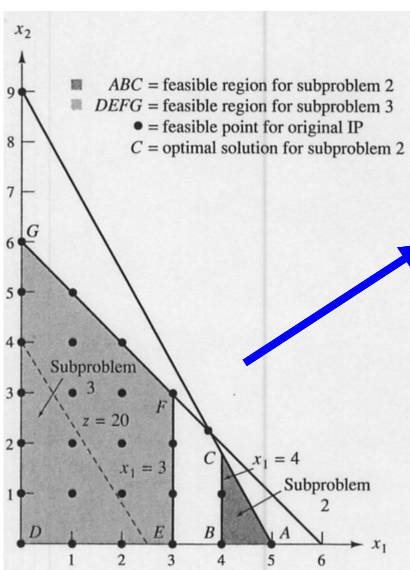


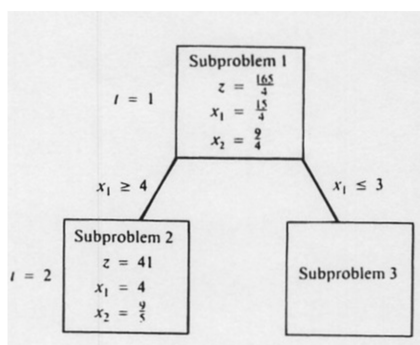
$$\max z = 8x_1 + 5x_2$$
$$\text{s.t.}\quad x_1 + x_2 \le 6$$
$$9x_1 + 5x_2 \le 45$$
$$x_1, x_2 \ge 0;\ x_1, x_2 \text{ integer}$$

Subproblem 2 Subproblem 1 + Constraint $x_1 \ge 4$.

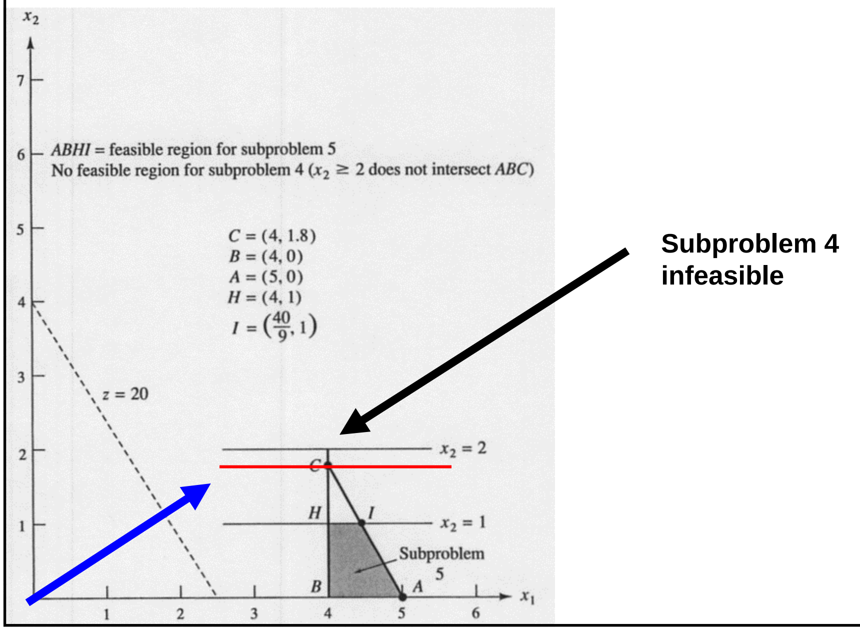
Subproblem 3 Subproblem 1 + Constraint $x_1 \le 3$.

Example (iteration 2)

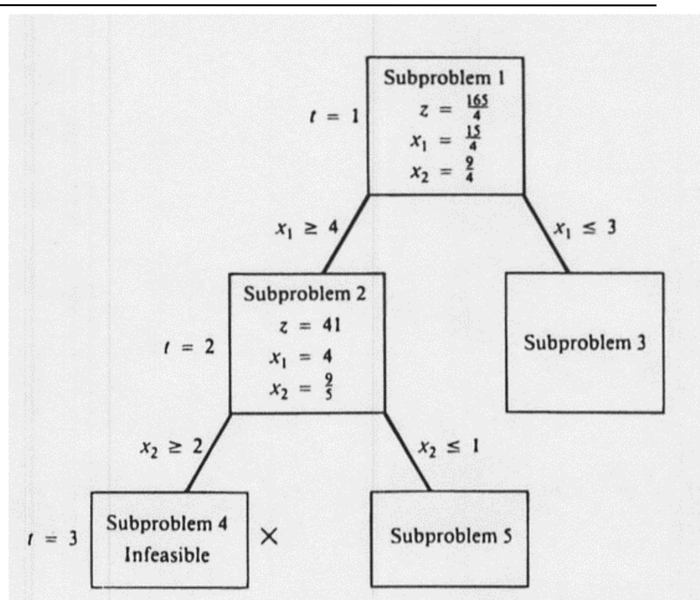




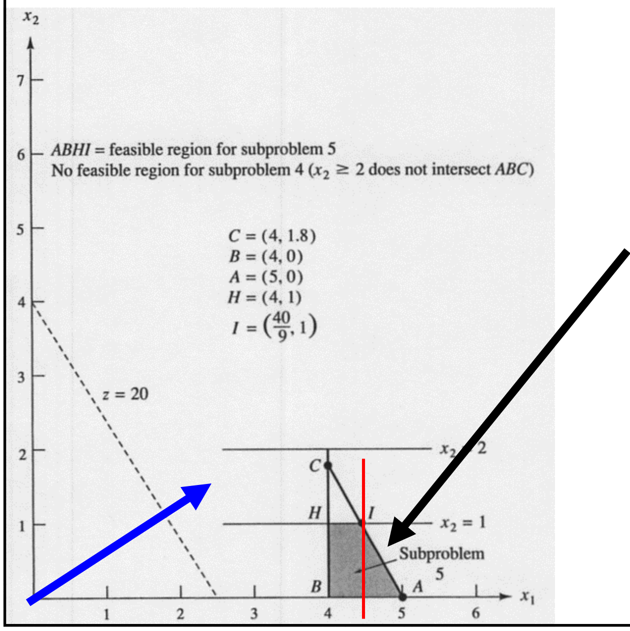
Example (iteration 3)



Example (iteration 3)

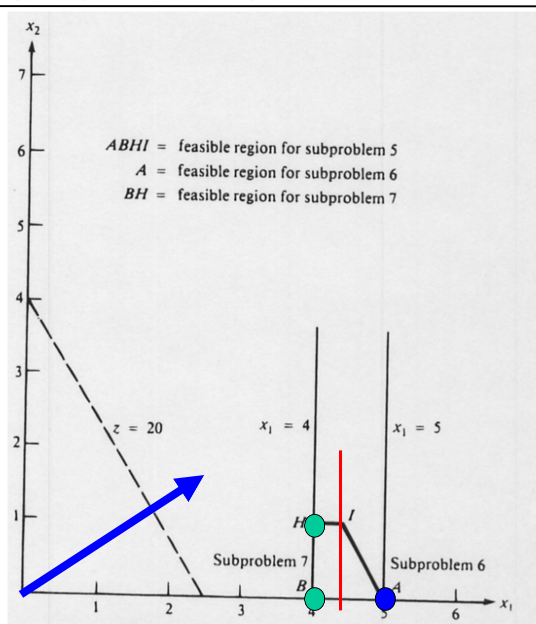


Example (iteration 4)



Subproblem 5: cut in two

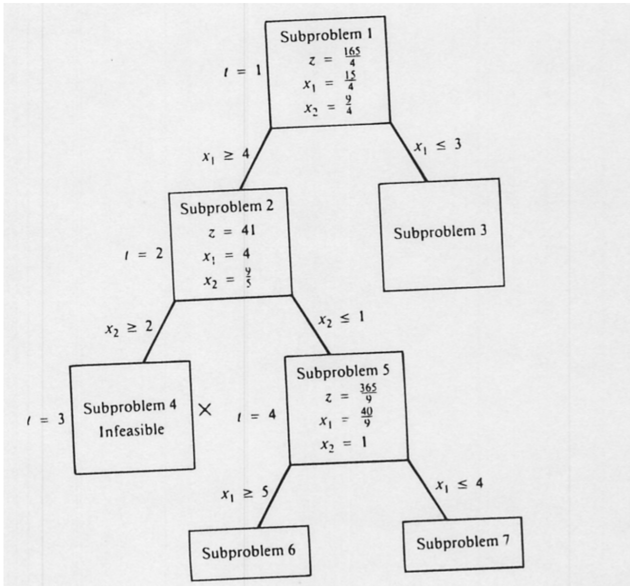
Example (iteration 4)



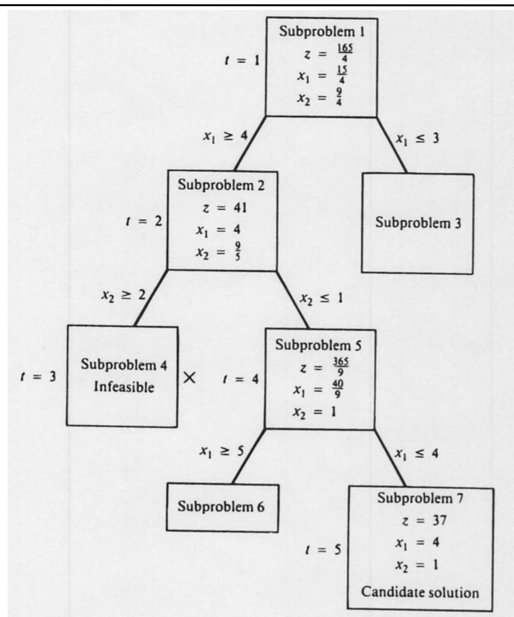
Subproblem 6

Subproblem 7

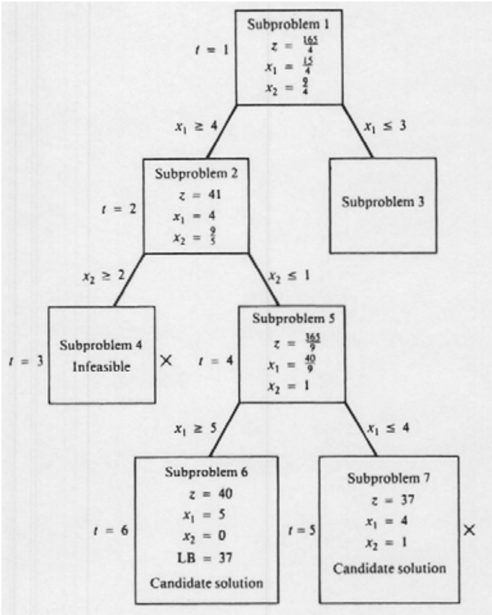
Example (iteration 4)



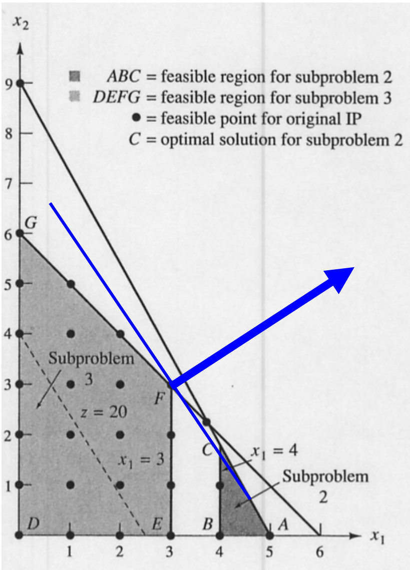
Example (iteration 5)



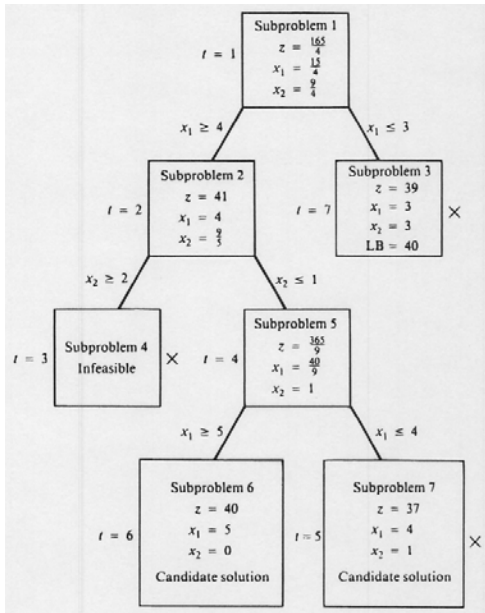
Example (iteration 6)



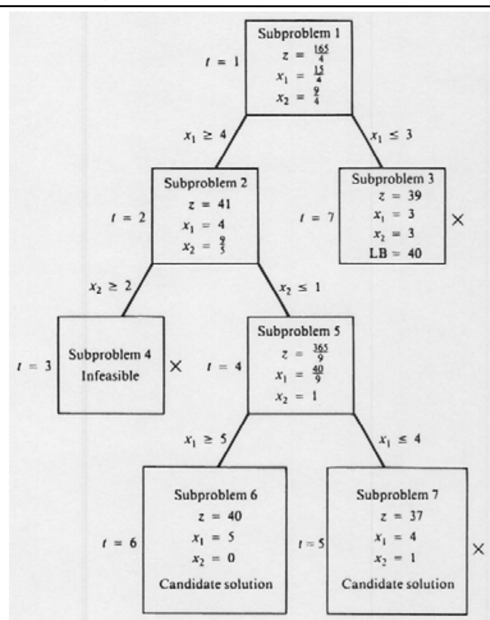
Example (iteration 7)



Example (iteration 7)



Example (iteration 7)



Example (iteration 7)