9.7 Implicit Enumeration

- Note that an integer constraint on $x < 2^{n+1}$ by setting $x = 2^n u_n + \cdots + u_0$ with $u_i \in \{0, 1\}$.
- We will see that 0-1 problems are easier to solve.
- The tree used in the implicit enumeration method is similar to those used to solve 0-1 knapsack problems.
- At each node, the values of some of the variables are specified.
- For instance, suppose a 0-1 problem has variables $x_1, x_2, x_3, x_4, x_5, x_6$.
- Part of the tree looks like Figure 26.

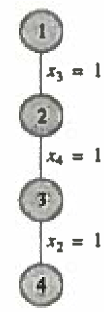


FIGURE 26

- At node 4, the values of x_3, x_4, and x_2 are specified.
- These variables are referred to as **fixed** variables.
- All variables whose values are unspecified at a node are called free variables.
- Thus, at node 4, x_1, x_5, and x_6 are **free** variables.
- For any node, a specification of the values of all the free variables is called a **completion** of the node.
- Thus $x_1 = 1, x_5 = 1, x_6 = 0$ is a completion of node 4.

Some general ideas:

1. At a node, find a completion of the free variables that optimize the objective function.

   If it is feasible, then we have the optimal solution at the node and no branching is needed.

   For example, $\max z = 4x_1 + 2x_2 - x_3 + 2x_4$

   s.t. $x_1 + 3x_2 - x_3 - 2x_4 \geq 1$,

   $x_i \in \{0, 1\}, \quad i = 1, 2, 3, 4.$

   If $(x_1, x_2, x_3, x_4) = (0, 1, 0, 1)$ is given, we see that the solution is feasible. The node is fathomed with $z = 4$.

2. If an optimal completion is not feasible, then we have an upper bound for the completion of the rest of the node.
3. If an optimal completion is worse than a feasible solution found before, the node can be removed.
4. If there is a constraint such that the most feasible completion fails, then there is no completion that are feasible.

Example: Implicit enumeration $\max z = -7x_1 - 3x_2 - 2x_3 - x_4 - 2x_5$

s.t. $\quad -4x_1 - 2x_2 + x_3 - 2x_4 - x_5 \le -3 \qquad (47)$

s.t. $\quad -4x_1 - 2x_2 - 4x_3 + x_4 + 2x_5 \le -7 \qquad (48)$

$\qquad x_i \in \{0, 1\} \quad i = 1, 2, 3, 4, 5.$

- At node 1, all variables are free.
- The optimal solution $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 0, 0, 0)$ is not feasible.
- Then check feasibility for (47), let $(x_1, x_2, x_3, x_4, x_5) = (1, 1, 0, 1, 1)$
- For (48), one may let $(x_1, x_2, x_3, x_4, x_5) = (1, 1, 1, 0, 0)$
- We set node 2 and node 3 for $x_1 = 1$ and 0, respectively.
- Consider node 2. The best completion is $(1, 0, 0, 0, 0)$, which is infeasible.

FIGURE 27

- Check $(1, 1, 0, 1, 1)$ and $(1, 1, 1, 0, 0)$, (47) and (48) are feasible.
- Branch on node 2 to get node 4 and node 5 with $x_2 = 1$ and 0, resp.

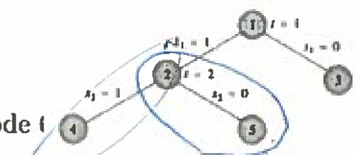


FIGURE 28

- Check optimal solution, feasible solutions at node 5; branch out to node
- Use the LIFO rule, consider node 6.
- The best completion is $(1, 0, 1, 0, 0)$, which is feasible.

  We found a candidate solution with $z = -9$.

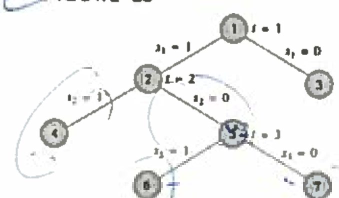


FIGURE 29

- By the LIFO rule, consider node 7.

  The best completion is $(1, 0, 0, 0, 0)$ with $z = -7$, which is better than $z = -9$.
- We check node 7 to see whether it has any feasible completion.

  For (47), $(1, 0, 0, 1, 1)$ is feasible; for (48), $(1, 0, 0, 0, 0)$ is infeasible
- Thus, no completion of node 7 can satisfy (48), and the node can be eliminated (Figure 30).

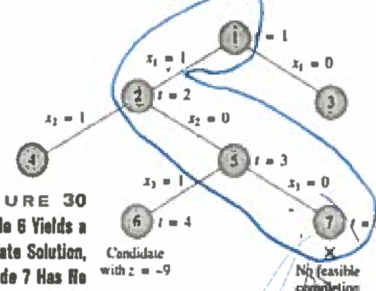


FIGURE 30 Node 6 Yields a Candidate Solution, and Node 7 Has No Feasible Completion

- By the LIFO rule, we analyze node 4.
- The best completion is $(1, 1, 0, 0, 0)$ with $z = -10$.
- Thus, node 4 cannot beat the previous candidate solution from node 6, where $z = -9$.
- Then node 4 may be eliminated; see Figure 31.
- It remains to consider node 3.
- The best completion is $(0, 0, 0, 0, 0)$, which is impossible.
- Because $z = 0$, it is possible that node 3 can yield better solution.
- We now check whether node 3 has any feasible completion:

  $(0, 1, 1, 1, 1)$ satisfies (47), but $(0, 1, 1, 0, 0)$ fails (48).
- So, node 3 may be eliminated from consideration.
- We now have the tree in Figure 32.
- Because there are no nodes left to analyze, the node 6 with

  $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 1, 0, 0)$ and $z = -9$ is optimal.

FIGURE 31
Node 4 Cannot Beat Node 6 Candidate

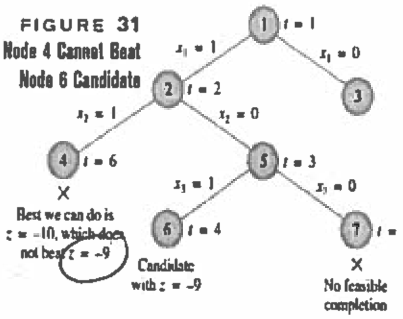
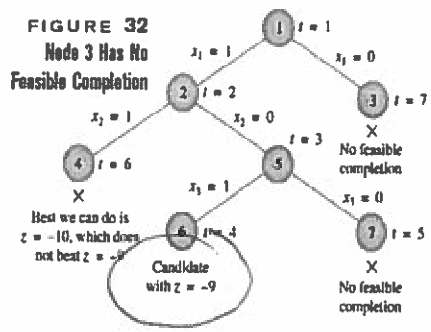


FIGURE 32
Node 3 Has No Feasible Completion



Cutting plane algorithm

Example $\max z = 8x_1 + 5x_2$

s.t. $x_1 + x_2 \leq 6$

s.t. $9x_1 + 5x_2 \leq 45$

$x_1, x_2 \geq 0$, x_1, x_2 integer.

We use slack variable and get the optimal solution for the relaxation problem $z = 41.25$ with $(x_1, x_2) = (3.75, 2.25)$.

Select a variable in fractional form and modify a constraint, say, the second one, to rule out this solution, but keep all integer solutions as follows.

- Let $\lfloor x \rfloor$ be the largest integer smaller than $x \in \mathbb{R}$.
- Change the second constraint

  $x_1 + 1.25s_1 + 0.25s_2 = 3.75$ in the final tableau to:

  $x_1 - 2s_1 + 0.75s_1 + 0s_2 + 0.25s_2 = 3 + 0.75$, i.e.,

  $x_1 - 2s_1 + 0s_2 - 3 = 0.75 - 0.75s_1 - 0.25s_2$.
- Now we add the **cut** constraint:

  $0.75 - 0.75s_1 - 0.25s_2 \leq 0$ so that

  1) Any feasible point for the IP will satisfy the cut.

  2) The current optimal solution to the LP relaxation will not satisfy the cut.
- We can now solve the LP problem with the new cut constraint. (See Figure 33 and Table 84.)
- Use dual simplex method to get the solution in Table 85: $z = 40, (x_1, x_2) = (5, 0)$.
- Recall that a cut does not eliminate any points that are feasible for the IP.
- We can apply several cuts until we solve the IP.

TABLE 83

Optimal Tableau for LP Relaxation of Telfa

z	x_1	x_2	s_1	s_2	rhs
0	0	1	2.25	−0.25	2.25
0	1	0	−1.25	0.25	3.75
1	0	0	1.25	0.75	41.25

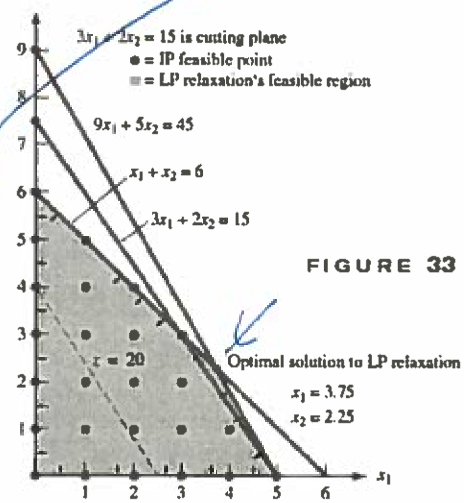


FIGURE 33

TABLE 84

Cutting Plane Tableau After Adding Cut (55)

z	x_1	x_2	s_1	s_2	s_3	rhs
0	0	1	2.25	−0.25	0	2.25
0	1	0	−1.25	0.25	0	3.75
0	0	0	−0.75	−0.25	1	−0.75
1	0	0	1.25	0.75	0	41.25

TABLE 85

Optimal Tableau for Cutting Plane

z	x_1	x_2	s_1	s_2	s_3	rhs
1	0	0	0	0.33	1.67	40
0	0	1	0	−1	3	0
0	1	0	0	0.67	−1.67	5
0	0	0	1	0.33	−1.33	1

Math 323 Operations Research **Chapters 7-9 review** C.K. Li

Chapter 7 Transportation problem and transshipment problem

Transportation problem $\min z = \sum_{ij} c_{ij}x_{ij}$ s.t. $\sum_j x_{ij} \leq s_i, \sum_i x_{ij} \geq d_j$, $x_{ij} \geq 0$.

1. Convert it to a balanced problem. Add dummy demand column with 0 cost; add dummy supply with penalty cost (in different form).
2. Use NW corner method, minimum cost, or Vogel;s method to find a bfs.
3. Set $u_1 = 0$ and solve $u_i + v_j = c_{ij}$ for the basic variables.
4. If $(u_i + u_j) \leq c_{ij}$ for all i, j, then we have optimal.
5. Else choose x_{ij} to be the entering basic variable, where maximum $u_i - u_j - c_{ij} > 0$.
6. Choose a loop and Δ to determine the leaving variable; determine the new feasible solution.

Remarks a) Do suitable adjustment if one considers $\max z = c_{ij}x_{ij}$.
b) One can do sensitivity analysis by changing c_{ij} or changing s_i, d_j simultaneously.
c) One can use the techniques do inventory problem, and transshipment problem, etc.
d) One can also do assignment problem; the Hungarian method is preferred.

Chapter 8 Network models

Shortest path problem Dijkstra's algorithm.

Remarks a) One can formulate the problem as a transshipment problem.
b) One can use the method to solve equipment replacement problem.

Maximum flow / minimum cut problem Ford-Fulkerson Algorithm.

1. Find an initial flow; then find $so - si$ chain to improve the flow.
2. If there is no $s0 - si$ chain then the flow is maximum; one can determine the minimum cut.

Remark Special case include the optimal matching.

Critical Path Method and Project Management and Review Techniques

1. Formulate the network problem (label the activities as edges).
2. Find the critical path (by minimum path algorithm (in terms of the total flow).
3. One may crash the project in m days by solving the LP problem with n nodes:
   $\min z = \sum_{ij} c_{ij}A_{ij}$ s.t. $x_j - x_i \geq d_{ij} - A_{ij}$, $x_n - x_1 \leq m$, x_j urs, $1 \leq A_{ij} \leq m_{ij}$.
4. Note that one can set $x_1 = 0$ so that $x_j \geq 0$.
5. Using statistical techniques, one can evaluate and estimate the completion day. But there are limitation.

Mimumum spanning tree An easy greedy algorithm.

Minimum cost network problem $\min z = c_{ij}x_{ij}, \quad \sum_j x_{ij} - x_{ki} = b_i, \quad L_{ij} \le x_{ij} \le U_{ij}$.

Network Simplex method

1 Determine a starting bfs. The $n-1$ basic variables will correspond to a spanning tree. Indicate nonbasic variables at their upper bound by dashed arcs.

2 Compute $y_1, y_2, ...y_n$ (often called the simplex multipliers) by solving $y_1 = 0$, $y_i - y_j = c_{ij}$ for all basic variables x_{ij}.

For all nonbasic variables, determine the first / last row coefficient $\bar{c}_{ij} = y_i - y_j - c_{ij}$.

The current bfs is optimal if $\bar{c}_{ij} \le 0$ for all $x_{ij} = L_{ij}$ and $\bar{c}_{ij} \ge 0$ for all $x_{ij} = U_{ij}$.

If the bfs is not optimal, choose the nonbasic variable that most violates the optimality conditions as the entering basic variable.

3 Identify the cycle (there will be exactly one!) created by adding the arc corresponding to the entering variable to the current spanning tree of the current bfs.

Use conservation of flow to determine the new values of the variables in the cycle.

The variable that exits the basis will be the variable that first hits its upper or lower bound as the value of the entering basic variable is changed.

4 Find the new bfs by changing the flows of the arcs in the cycle found in step 3. Now go to step 2.

Integer Programming

- Many LP program may require some or all the variables $x_i$to be integers.
- We have mixed or pure IP. Sometimes, we require $x_i \in \{0, 1\}$.
- We may change all IP constraints to 0-1 constraints using binary numbers representation $x = u_n 2^n + \cdots + u_0$.

Branch and bound method

1. Solve Subproblem - LP relaxation.
2. Branch at the variables assuming fractional values.
3. Each branching remove some non-integral points in subproblem 1.
4. For each subprogram, one may fathom the corresponding node if:

   a) an integer optimal solution (a candidate solution) for the subproblem is found,

   b) the subproblem is infeasible,

   c) the subproblem has optimal solution less than or equal to a candidate solution.

Remarks One may consider the special cases:

a) knapsack problem (use c_i/a_i ratios to find an initial solution),

b) TSL problem, and 0-1 (use assignment problem to solve subproblem),

c) 0-1 problem, use optimal choices of x_i for $\max z = \sum_i c_i x_j$ and $\sum_j a_{ij} x_j \le b_i$.