Integer Programming

Xi Chen

Department of Management Science and Engineering
International Business School
Beijing Foreign Studies University

Definition 1.1

An **integer programming problem** *(IP) is an LP in which some or all of the variables are required to be non-negative integers, i.e., the Divisibility Assumption in LP does not hold.*

A **pure integer programming problem**

$$\begin{aligned} \max \quad & 3x_1 + 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 6 \quad x_1, x_2 \geq 0, \ x_1, x_2 \text{ integer.} \end{aligned}$$

A **mixed integer programming problem**

$$\begin{aligned} \max \quad & 3x_1 + 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 6, \quad x_1, x_2 \geq 0, \ x_1 \text{ integer.} \end{aligned}$$

A **0-1 integer programming problem**

$$\begin{aligned} \max \quad & x_1 - x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 \leq 2, \quad 2x_1 - x_2 \leq 1, \quad x_1, x_2 = 0 \text{ or } 1. \end{aligned}$$

Definition 1.2

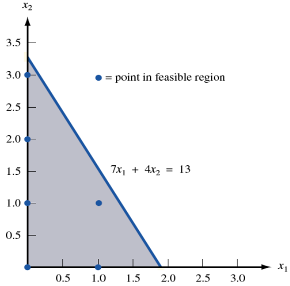
The LP obtained by omitting all integer or $0-1$ *constraints on variables is called the* **LP relaxation** *of the IP.*

The feasible region for any IP must be contained in the feasible region for the corresponding LP relaxation. For any IP that is a max problem, this implies that

$$\text{Optimal } z\text{-value for LP relaxation} \geq \text{optimal } z\text{-value for IP.}$$

Consider

$$\begin{aligned} \max \quad & 21x_1 + 11x_2 \\ \text{s.t.} \quad & 7x_1 + 4x_2 \leq 13, \\ & x_1, x_2 \geq 0, \ x_1, x_2 \text{ integer.} \end{aligned}$$



An IP is very difficult to solve because

- Enumeration may be impossible.
- Roundoff may be wrong or infeasible.

Example 1

Gandhi Cloth Company is capable of manufacturing three types of clothing: shirts, shorts, and pants. The manufacture of each type of clothing requires that Gandhi have the appropriate type of machinery available. The machinery needed to manufacture each type of clothing must be rented at the following rates: shirt machinery, $200 per week; shorts machinery, $150 per week; pants machinery, $100 per week. The manufacture of each type of clothing also requires the amounts of cloth and labor shown in the left table. Each week, 150 hours of labor and 160 sq yd of cloth are available. The variable unit cost and selling price for each type of clothing are shown in the right table.

Formulate an IP whose solution will maximize Gandhi's weekly profits.

Resource Requirements for Gandhi

Clothing Type	Labor (Hours)	Cloth (Square Yards)
Shirt	3	4
Shorts	2	3
Pants	6	4

Revenue and Cost Information for Gandhi

Clothing Type	Sales Price ($)	Variable Cost ($)
Shirt	12	6
Shorts	8	4
Pants	15	8

Define

$$x_1 = \text{number of shirts produced each week}$$
$$x_2 = \text{number of shorts produced each week}$$
$$x_3 = \text{number of pants produced each week}$$

and for $i = 1, 2, 3$, define

$$y_i = \begin{cases} 1, & \text{if } x_i > 0 \text{ are manufactured,} \\ 0, & \text{if } x_i = 0. \end{cases}$$

We can have the following IP model:

$$\begin{aligned} \max \quad & 6x_1 + 4x_2 + 7x_3 - 200y_1 - 150y_2 - 100y_3 \\ \text{s.t.} \quad & 3x_1 + 2x_2 + 6x_3 \leq 150 \quad \text{(Labor constraint)} \\ & 4x_1 + 3x_2 + 4x_3 \leq 160 \quad \text{(Cloth constraint)} \\ & x_1 \leq M_1 y_1,\ x_2 \leq M_2 y_2,\ x_3 \leq M_3 y_3, \quad (\textbf{Fixed charge}) \\ & x_1, x_2, x_3 \geq 0,\ x_1, x_2, x_3 \text{ integer}, \\ & y_1, y_2, y_3 = 0 \text{ or } 1. \end{aligned}$$

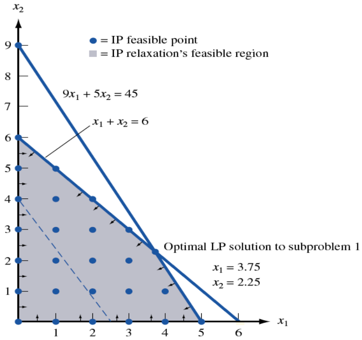
If you solve the LP relaxation of a pure IP and obtain a solution in which all variables are integers, then the optimal solution to the LP relaxation is also the optimal solution to the IP.

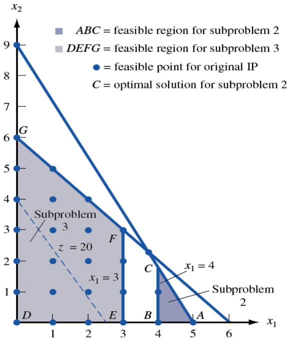
Example 2

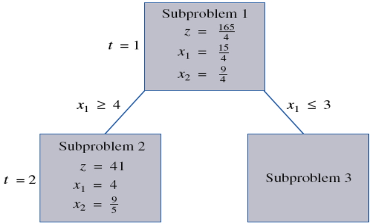
The Telfa Corporation manufactures tables and chairs. A table requires 1 hour of labor and 9 square board feet of wood, and a chair requires 1 hour of labor and 5 square board feet of wood. Currently, 6 hours of labor and 45 square board feet of wood are available. Each table contributes \$8 to profit, and each chair contributes \$5 to profit.

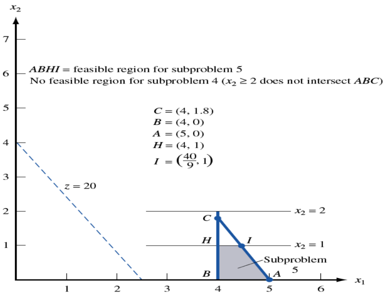
Formulate and solve an IP to maximize Telfa's profit.

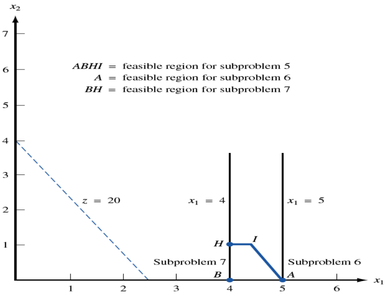
$$\begin{aligned} \max \quad & 8x_1 + 5x_2 \\ \text{s.t.} \quad & x_1 + x_2 \le 6 \quad \text{(Labor constraint)} \\ & 9x_1 + 5x_2 \le 45 \quad \text{(Wood constraint)}, \quad x_1, x_2 \ge 0,\ x_1, x_2 \text{ integer.} \end{aligned}$$

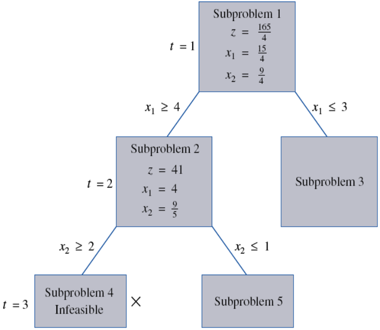

x_2
x_1
= IP feasible point
= IP relaxation's feasible region
$9x_1 + 5x_2 = 45$
$x_1 + x_2 = 6$
Optimal LP solution to subproblem 1
$x_1 = 3.75$
$x_2 = 2.25$

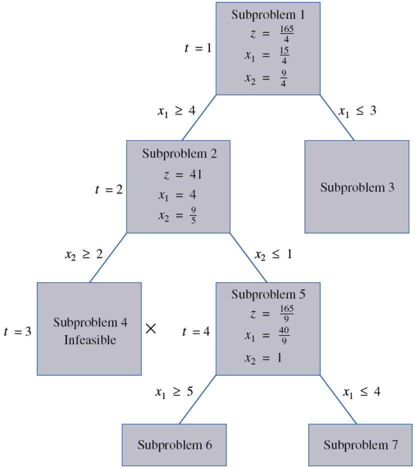

x_2
x_1
ABC = feasible region for subproblem 2
$DEFG$ = feasible region for subproblem 3
= feasible point for original IP
C = optimal solution for subproblem 2
Subproblem 3
$z = 20$
$x_1 = 3$
$x_1 = 4$
Subproblem 2

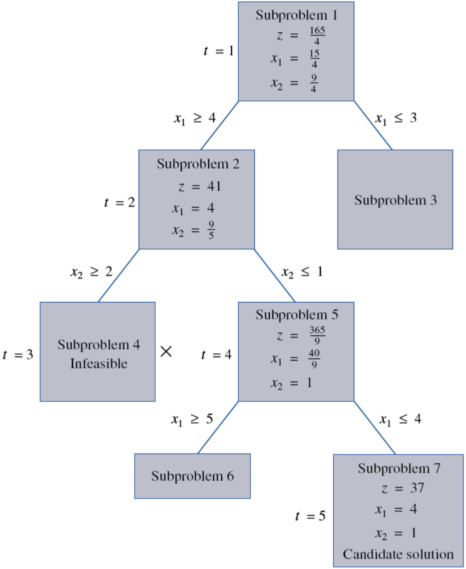

Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
$t = 1$
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
$t = 2$
Subproblem 3


ABHI = feasible region for subproblem 5
No feasible region for subproblem 4 ($x_2 \geq 2$ does not intersect ABC)
C = (4, 1.8)
B = (4, 0)
A = (5, 0)
H = (4, 1)
I = $\left(\frac{40}{9}, 1\right)$
z = 20
$x_2 = 2$
$x_2 = 1$
Subproblem 5
x_1
x_2


ABHI = feasible region for subproblem 5
A = feasible region for subproblem 6
BH = feasible region for subproblem 7
z = 20
$x_1 = 4$
$x_1 = 5$
Subproblem 7
Subproblem 6
x_1
x_2


Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
t = 1
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
t = 2
Subproblem 3
$x_2 \geq 2$
$x_2 \leq 1$
t = 3
Subproblem 4
Infeasible
Subproblem 5


Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
$t = 1$
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
$t = 2$
Subproblem 3
$x_2 \geq 2$
$x_2 \leq 1$
$t = 3$
Subproblem 4
Infeasible
$t = 4$
Subproblem 5
$z = \frac{165}{9}$
$x_1 = \frac{40}{9}$
$x_2 = 1$
$x_1 \geq 5$
$x_1 \leq 4$
Subproblem 6
Subproblem 7


Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
$t = 1$
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
$t = 2$
Subproblem 3
$x_2 \geq 2$
$x_2 \leq 1$
$t = 3$
Subproblem 4
Infeasible
$t = 4$
Subproblem 5
$z = \frac{365}{9}$
$x_1 = \frac{40}{9}$
$x_2 = 1$
$x_1 \geq 5$
$x_1 \leq 4$
Subproblem 6
Subproblem 7
$z = 37$
$x_1 = 4$
$x_2 = 1$
Candidate solution
$t = 5$

Definition 2.1

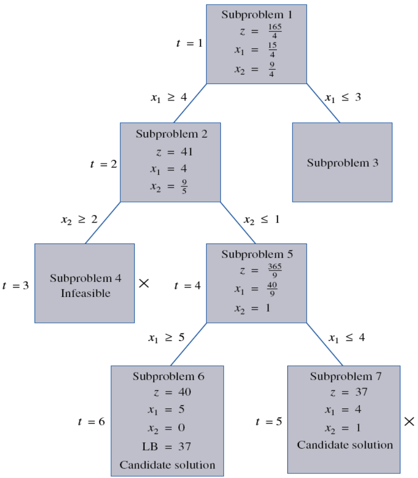
When further branching on a subproblem cannot yield any useful information, we say that the subproblem (or node) is **fathomed.**

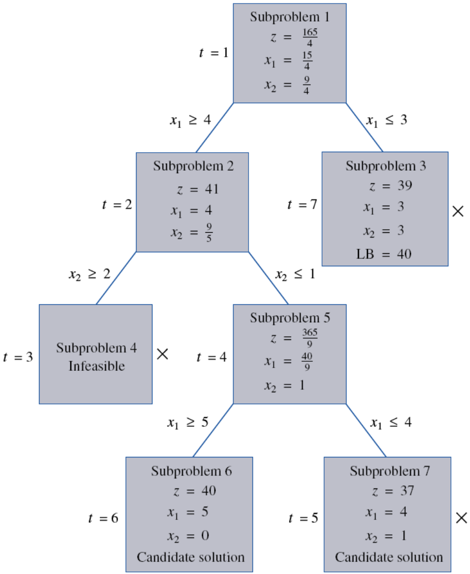
Definition 2.2

A solution obtained by solving a subproblem in which all variables have integer values is a **candidate solution***. The z-value for the candidate solution is a* **lower bound** *on the optimal z-value for the original IP.*

Definition 2.3

The **LIFO** *(last in first out) approach, which chooses to solve the most recently created subproblem, is often called* **backtracking***. When branching on a node, the* **jumptracking** *approach solves all the problems created by the branching. Then it branches again on the node with the best z-value.*


Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
$t = 1$
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
$t = 2$
Subproblem 3
$x_2 \geq 2$
$x_2 \leq 1$
Subproblem 4
Infeasible
$t = 3$
Subproblem 5
$z = \frac{365}{9}$
$x_1 = \frac{40}{9}$
$x_2 = 1$
$t = 4$
$x_1 \geq 5$
$x_1 \leq 4$
Subproblem 6
$z = 40$
$x_1 = 5$
$x_2 = 0$
LB = 37
Candidate solution
$t = 6$
Subproblem 7
$z = 37$
$x_1 = 4$
$x_2 = 1$
Candidate solution
$t = 5$


Subproblem 1
$z = \frac{165}{4}$
$x_1 = \frac{15}{4}$
$x_2 = \frac{9}{4}$
$t = 1$
$x_1 \geq 4$
$x_1 \leq 3$
Subproblem 2
$z = 41$
$x_1 = 4$
$x_2 = \frac{9}{5}$
$t = 2$
Subproblem 3
$z = 39$
$x_1 = 3$
$x_2 = 3$
LB = 40
$t = 7$
$x_2 \geq 2$
$x_2 \leq 1$
Subproblem 4
Infeasible
$t = 3$
Subproblem 5
$z = \frac{365}{9}$
$x_1 = \frac{40}{9}$
$x_2 = 1$
$t = 4$
$x_1 \geq 5$
$x_1 \leq 4$
Subproblem 6
$z = 40$
$x_1 = 5$
$x_2 = 0$
Candidate solution
$t = 6$
Subproblem 7
$z = 37$
$x_1 = 4$
$x_2 = 1$
Candidate solution
$t = 5$

The key aspects of the branch-and-bound method for solving pure IPs may be summarized as follows:

Step 1 If it is unnecessary to branch on a subproblem, then it is fathomed. The following three situations result in a subproblem being fathomed:

(1) The subproblem is infeasible;
(2) the subproblem yields an optimal solution in which all variables have integer values; and
(3) the optimal z-value for the subproblem does not exceed (in a max problem) the current LB.

Step 2 A subproblem may be eliminated from consideration in the following situations:

(1) The subproblem is infeasible (in the Telfa problem, subproblem 4 was eliminated for this reason);
(2) the LB (representing the z-value of the best candidate to date) is at least as large as the z-value for the subproblem (in the Telfa problem, subproblems 3 and 7 were eliminated for this reason).

Subproblems for branch-and-bound problems are often solved using some variant of the dual simplex algorithm. In the Telfa example, the optimal tableau for the LP relaxation is

$$\begin{array}{rrrrrrrl} z & & + & 1.25s_1 & + & 0.75s_2 & = & 41.25 \\ & x_2 & + & 2.25s_1 & - & 0.25s_2 & = & 2.25 \\ x_1 & & - & 1.25s_1 & + & 0.25s_2 & = & 3.75 \end{array}$$

After solving the LP relaxation, we solved subproblem 2, which is just subproblem 1 plus the constraint $x_1 \geq 4$.

Initial Tableau for Solving Subproblem 2 by Dual Simplex

	Basic Variable
$z + 1.25s_1 + 0.75s_2 = 41.25$	$z = 41.25$
$x_2 + 2.25s_1 - 0.25s_2 = 2.25$	$x_2 = 2.25$
$x_1 - 1.25s_1 + 0.25s_2 = 3.75$	$x_1 = 3.75$
$- 1.25s_1 + 0.25s_2 + e_3 = -0.25$	$e_3 = -0.25$

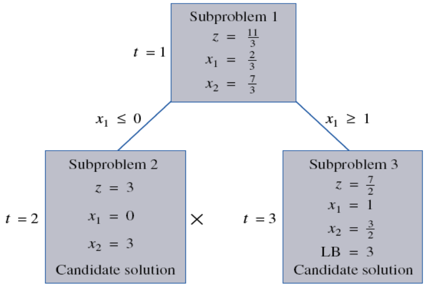
Optimal Tableau for Solving Subproblem 2 by Dual Simplex

	Basic Variable
$z + s_2 + e_3 = 41$	$z = 41$
$x_2 + 0.20s_2 + 1.8e_3 = 1.8$	$x_2 = 1.8$
$x_1 - e_3 = 4$	$x_1 = 4$
$s_1 - 0.20s_2 - 0.80e_3 = 0.20$	$s_1 = 0.20$

Solving Mixed Integer Programming Problems

Modify the branch-and-bound method for solving pure IP by branching only on variables that are required to be integers. Also, for a solution to a subproblem to be a candidate solution, it need only assign integer values to those variables that are required to be integers.

$$\begin{aligned} \max \quad & 2x_1 + x_2 \\ \text{s.t.} \quad & 5x_1 + 2x_2 \le 8, \quad x_1 + x_2 \le 3, \quad x_1, x_2 \ge 0,\ x_1 \text{ integer.} \end{aligned}$$



Solving Knapsack Problems

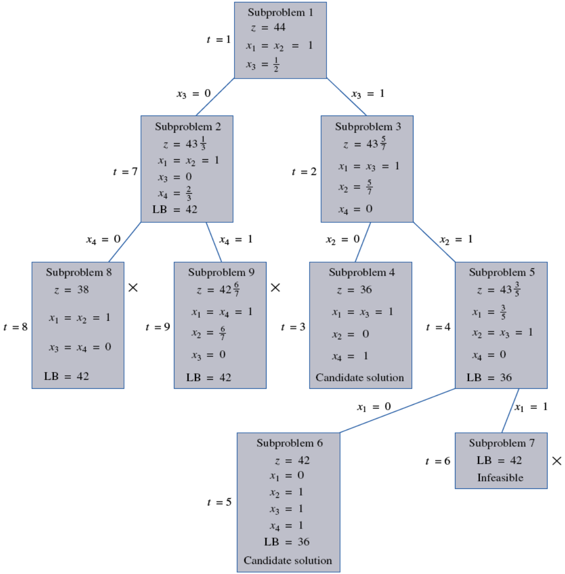
$$\begin{aligned} \max \quad & c_1x_1 + c_2x_2 + \ldots + c_nx_n \\ \text{s.t.} \quad & a_1x_1 + a_2x_2 + \ldots + a_nx_n \le b, \quad x_i = 0 \text{ or } 1, \ (i = 1, 2, \ldots, n) \end{aligned}$$

Example 3

Stockco is considering four investments. Investment 1 will yield a net present value (NPV) of $16,000; investment 2, an NPV of $22,000; investment 3, an NPV of $12,000; and investment 4, an NPV of $8,000. Each investment requires a certain cash outflow at the present time: investment 1, $5,000; investment 2, $7,000; investment 3, $4,000; and investment 4, $3,000. Currently, $14,000 is available for investment. Formulate an IP that will tell Stockco how to maximize the NPV.

$$\begin{aligned} \max \quad & 16x_1 + 22x_2 + 12x_3 + 8x_4 \\ \text{s.t.} \quad & 5x_1 + 7x_2 + 4x_3 + 3x_4 \le 14, \quad x_1, x_2, x_3, x_4 = 0 \text{ or } 1 \end{aligned}$$

The optimal solution obtained by the branch-and-bound method is $z = 42$, $x_1 = 0$, $x_2 = x_3 = x_4 = 1$. The "best" investment is not used.



Solving Combinatorial Optimization Problems

Example 4 (Machine Scheduling)

Four jobs must be processed on a single machine. The time required to process each job and the date the job is due are shown in the following table. The delay of a job is the number of days after the due date that a job is completed (if a job is completed on time or early, the job's delay is zero). In what order should the jobs be processed to minimize the total delay of the four jobs?

$$x_{ij} = \begin{cases} 1 & \text{if job } i \text{ is the } j\text{th job to be processed} \\ 0 & \text{otherwise} \end{cases}$$

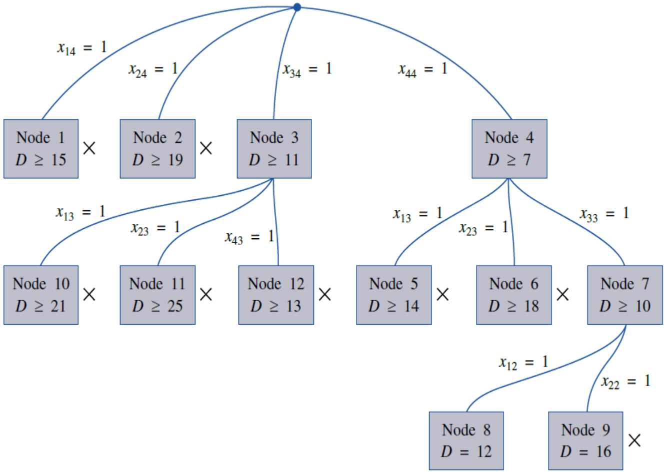
Durations and Due Date of Jobs

Job	Days Required to Complete Job	Due Date
1	6	End of day 8
2	4	End of day 4
3	5	End of day 12
4	8	End of day 16

Delays Incurred If Jobs Are Processed in the Order 1–2–3–4

Job	Completion Time of Job	Delay of Job
1	6	0
2	$6 + 4 = 10$	$10 - 4 = 6$
3	$6 + 4 + 5 = 15$	$15 - 12 = 3$
4	$6 + 4 + 5 + 8 = 23$	$23 - 16 = 7$

The jobs should be processed in the order $2-1-3-4$, with a total delay of 12 days resulting.



Example 5 (Traveling Salesperson Problem)

Joe State lives in Gary, Indiana. He owns insurance agencies in Gary, Fort Wayne, Evansville, Terre Haute, and South Bend. Each December, he visits each of his insurance agencies. The distance between each agency (in miles) is shown in the following table. What order of visiting his agencies will minimize the total distance traveled?

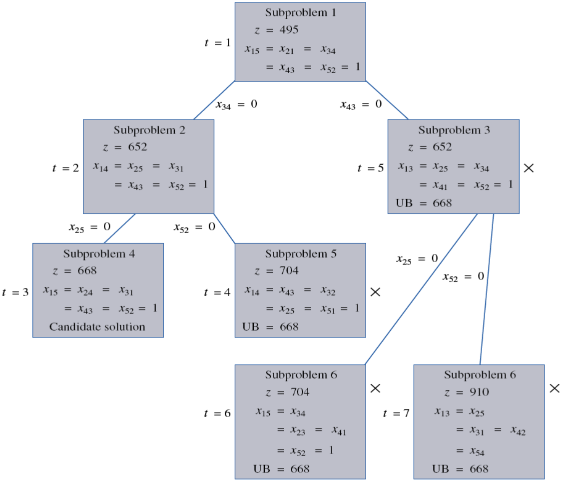
Distance between Cities in Traveling Salesperson Problem

Day	Gary	Fort Wayne	Evansville	Terre Haute	South Bend
City 1 Gary	0	132	217	164	58
City 2 Fort Wayne	132	0	290	201	79
City 3 Evansville	217	290	0	113	303
City 4 Terre Haute	164	201	113	0	196
City 5 South Bend	58	79	303	196	0

Definition 2.4

An itinerary that begins and ends at the same city and visits each city once is called a **tour***. A* **subtour** *is a round trip that does not pass through all cities.*

If the solution to the preceding assignment problem yields a tour, then it is the optimal solution to the TSP. The optimal solution to the assignment problem might be $x_{15} = x_{21} = x_{34} = x_{43} = x_{52} = 1$, which contains two subtours. The results of the branch-and-bound procedure are given below.



We first solve the assignment problem (Subproblem 1) in the following table. The optimal solution is $x_{15} = x_{21} = x_{34} = x_{43} = x_{52} = 1$.

Cost Matrix for Subproblem 1

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	113	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

We choose to exclude the subtour $3-4-3$ by branching on Subproblem 1 by adding two subproblems: Subproblem 1 + ($x_{34} = 0$, or $c_{34} = M$) (Subproblem 2); Subproblem 1 + ($x_{43} = 0$, or $c_{43} = M$) (Subproblem 3).

Cost Matrix for Subproblem 2

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

We arbitrarily choose Subproblem 2 to solve. The optimal solution is $z = 652$, $x_{14} = x_{25} = x_{31} = x_{43} = x_{52} = 1$ with two subtours $1 - 4 - 3 - 1$ and $2 - 5 - 2$.

We now branch on Subproblem 2 in an effort to exclude the subtour $2 - 5 - 2$ by adding two subproblems: Subproblem 2 + ($x_{25} = 0$, or $c_{25} = M$) (Subproblem 4); Subproblem 2 + ($x_{52} = 0$, or $c_{52} = M$) (Subproblem 5).

Cost Matrix for Subproblem 4

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	M
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	79	303	196	M

We arbitrarily choose to solve Subproblem 4, and obtain the optimal solution $z = 668$, $x_{15} = x_{24} = x_{31} = x_{43} = x_{52} = 1$ as a candidate solution.

The optimal solution to Subproblem 5 is $z = 704$, $x_{14} = x_{43} = x_{32} = x_{25} = x_{51} 1$. This solution is a tour, but $z = 704 > 668$. Thus, Subproblem 5 may be eliminated from consideration.

Cost Matrix for Subproblem 5

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	M	303
City 4	164	201	113	M	196
City 5	58	M	303	196	M

We find the optimal solution to Subproblem is $x_{13} = x_{25} = x_{34} = x_{41} = x_{52} = 1$, $z = 652$. This solution contains the subtours $1 - 3 - 4 - 1$ and $2 - 5 - 2$.

Cost Matrix for Subproblem 3

	City 1	City 2	City 3	City 4	City 5
City 1	M	132	217	164	58
City 2	132	M	290	201	79
City 3	217	290	M	113	303
City 4	164	201	M	M	196
City 5	58	79	303	196	M

Because $652 < 668$, we now branch on Subproblem 3 in an effort to exclude the subtour $2 - 5 - 2$by adding two subproblems: Subproblem 3 + ($x_{25} = 0$, or $c_{25} = M$) (Subproblem 6); Subproblem 3 + ($x_{52} = 0$, or $c_{52} = M$) (Subproblem 7).

The optimal solutions are $z = 704$ for Subproblem 6 and $z = 910$ for Subproblem 7. Neither can yield an optimal solution.

Hence, Subproblem 4 thus yields the optimal solution: Joe should travel from Gary to South Bend, from South Bend to Fort Wayne, from Fort Wayne to Terre Haute, from Terre Haute to Evansville, and from Evansville to Gary. Joe will travel a total distance of 668 miles.

Definition 2.5

A **heuristic** *is a method used to solve a problem by trial and error when an algorithmic approach is impractical.*

We now discuss two heuristics for the TSP: the nearest-neighbor (NNH) and the cheapest-insertion heuristics (CIH).

- To apply NNH, we begin at any city and then "visit" the nearest city. Then we go to the unvisited city closest to the city we have most recently visited. Continue in this fashion until a tour is obtained.
- In CIH, we begin at any city and find its closest neighbor. Then we create a subtour joining those two cities. Next, we replace an arc in the subtour [say, arc (i, j)] by the combination of two arcs$-(i, k)$ and (k, j), where k is not in the current subtour–that will increase the length of the subtour by the smallest (or cheapest) amount. Let c_{ij} be the length of arc (i, j). Note that if arc (i, j) is replaced by arcs (i, k) and (k, j), then a length $c_{ik} + c_{kj} - c_{ij}$ is added to the subtour. Then we continue with this procedure until a tour is obtained.

Determining Which Arc of (1, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added to Subtour	Added Length
(1, 5)*	(1, 2)–(2, 5)	$c_{12} + c_{25} - c_{15} = 153$
(1, 5)	(1, 3)–(3, 5)	$c_{13} + c_{35} - c_{15} = 462$
(1, 5)	(1, 4)–(4, 5)	$c_{14} + c_{45} - c_{15} = 302$
(5, 1)*	(5, 2)–(2, 1)	$c_{52} + c_{21} - c_{51} = 153$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$
(5, 1)	(5, 4)–(4, 1)	$c_{54} + c_{41} - c_{51} = 302$

Determining Which Arc of (1, 2)–(2, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added	Added Length
(1, 2)	(1, 3)–(3, 2)	$c_{13} + c_{32} - c_{12} = 375$
(1, 2)*	(1, 4)–(4, 2)	$c_{14} + c_{42} - c_{12} = 233$
(2, 5)	(2, 3)–(3, 5)	$c_{23} + c_{35} - c_{25} = 514$
(2, 5)	(2, 4)–(4, 5)	$c_{24} + c_{45} - c_{25} = 318$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$
(5, 1)	(5, 4)–(4, 1)	$c_{54} + c_{41} - c_{51} = 302$

Determining Which Arc of (1, 4)–(4, 2)–(2, 5)–(5, 1) Is Replaced

Arc Replaced	Arcs Added	Added Length
(1, 4)*	(1, 3)–(3, 4)	$c_{13} + c_{34} - c_{14} = 166$
(4, 2)	(4, 3)–(3, 2)	$c_{43} + c_{32} - c_{42} = 202$
(2, 5)	(2, 3)–(3, 5)	$c_{23} + c_{35} - c_{25} = 514$
(5, 1)	(5, 3)–(3, 1)	$c_{53} + c_{31} - c_{51} = 462$

Suppose the TSP consists of cities $1, 2, 3, \ldots, N$. For $i \neq j$, let $c_{ij} =$ distance from city i to city j and let $c_{ii} = M$, where M is a very large number (relative to the actual distances in the problem). Also define

$$x_{ij} = \begin{cases} 1 & \text{if the solution to TSP goes from city } i \text{ to city } j \\ 0 & \text{otherwise} \end{cases}$$

Then the solution to a TSP can be found by solving

$$\begin{aligned} \min \quad & \sum_i \sum_j c_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{N} x_{ij} = 1, \quad j = 1, 2, \ldots, N, \quad \sum_{j=1}^{N} x_{ij} = 1, \quad i = 1, 2, \ldots, N, \\ & u_i - u_j + N x_{ij} \leq N - 1, \quad i \neq j,\ i = 2, 3, \ldots, N,\ j = 2, 3, \ldots, N, \\ & x_{ij} = 0 \text{ or } 1,\ u_j \geq 0. \end{aligned}$$

The constraints $u_i - u_j + Nx_{ij} \leq N - 1$ are the key to the formulation. They ensure the following:

1 Any set of x_{ij}'s containing a subtour will be infeasible (that is, they violate $u_i - u_j + Nx_{ij} \leq N - 1$).

2 Any set of x_{ij}'s that forms a tour will be feasible (there will exist a set of u_j's that satisfy $u_i - u_j + Nx_{ij} \leq N - 1$).

Exercise 2.1

1 *Consider the solution* $x_{15} = x_{21} = x_{34} = x_{43} = x_{52} = 1$ *with two subtours* $1 - 5 - 2 - 1$ *and* $3 - 4 - 3$. *Show that the subtour* $3 - 4 - 3$ *violates* $u_i - u_j + Nx_{ij} \leq N - 1$.

2 *Consider the tour* $1 - 3 - 4 - 5 - 2 - 1$ *and choose* $u_1 = 1$, $u_2 = 5$, $u_3 = 2$, $u_4 = 3$, $u_5 = 4$. *Show that with this choice of the* u_i*'s,* $u_i - u_j + Nx_{ij} \leq N - 1$ *are satisfied.*

To express any pure IP as a $0-1$ IP: Simply express each variable in the original IP as the sum of powers of 2. Let n be the smallest integer such that we can be sure that $x_i < 2^{n+1}$. Then x_i may be (uniquely) expressed as the sum of $2^0, 2^1, \ldots, 2^{n-1}, 2^n$, and

$$x_i = u_n 2^n + u_{n-1} 2^{n-1} + \ldots + u_2 2^2 + u_1 2 + u_0.$$

Example 6

Suppose $x_i \leq 100$. Then $x_i < 2^{6+1} = 128$, which indicates that

$$x_i = 64u_6 + 32u_5 + 16u_4 + 8u_3 + 4u_2 + 2u_1 + u_0, \quad u_i = 0 \text{ or } 1.$$

Suppose $x_i = 93$. Then u_6 will be the largest multiple of $2^6 = 64$ that is contained in 93. This yields $u_6 = 1$; then the rest of the right side of x_i must equal $93 - 64 = 29$. Then u_5 will be the largest multiple of $2^5 = 32$ containing 29. This yields $u_5 = 0$. Continuing in this fashion, we can finally derive $93 = 2^6 + 2^4 + 2^3 + 2^2 + 2^0$.

Definition 3.1

At each node, the variables whose values are specified are referred to as **fixed variables***; and those unspecified variables are called* **free variables***. A specification of the values of all the free variables is called a* **completion** *of the node.*

Three main ideas used in implicit enumeration:

- Suppose we are at any node. Given the values of the fixed variables at that node, is there an easy way to find a good completion of that node that is feasible in the original $0-1$ IP?

Example 7

$$\begin{aligned} \max \quad & 4x_1 + 2x_2 - x_3 + 2x_4, \\ \text{s.t.} \quad & x_1 + 3x_2 - x_3 - 2x_4 \geq 1 \quad x_i = 0 \text{ or } 1. \end{aligned}$$

If $x_1 = 0$ and $x_2 = 1$ are fixed, then the best we can do is set $x_3 = 0$ and $x_4 = 1$. Since $\mathbf{x}$ is feasible, we have found the best feasible completion.

- Even if the best completion of a node is not feasible, the best completion gives us a bound on the best objective function value that can be obtained via a feasible completion of the node. This bound can often be used to eliminate a node from consideration.

Example 8

Suppose we have previously found a candidate solution with $z = 6$, and our objective is to maximize

$$4x_1 + 2x_2 + x_3 - x_4 + 2x_5.$$

Also suppose that the fixed variables are $x_1 = 0$, $x_2 = 1$, and $x_3 = 1$. Then the best completion of this node is $x_4 = 0$ and $x_5 = 1$, which yields $z = 5$. Because $z = 5$ cannot beat the candidate with $z = 6$, we can immediately eliminate this node from consideration (whether or not the completion is feasible is irrelevant).

- At any node, is there an easy way to determine if all completions of the node are infeasible?

Example 9

Suppose that the fixed variables are $x_4 = 1$, $x_2 = 1$, and $x_3 = 1$ and one of the constraints is

$$-2x_1 + 3x_2 + 2x_3 - 3x_4 - x_5 + 2x_6 \leq -5.$$

We assign values to the free variables that make the left side as small as possible. If this completion of the node will not satisfy the constraint, then certainly no completion of the node can. Thus, we set $x_1 = 1$, $x_5 = 1$, and $x_6 = 0$, which yields $-1 \leq -5$, a contradiction.

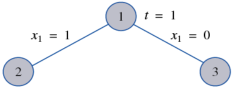
In general, we check whether a node has a feasible completion by looking at each constraint and assigning each free variable the best value for satisfying the constraint.

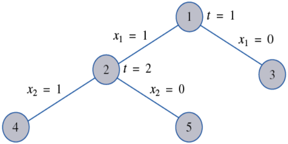
Example 10

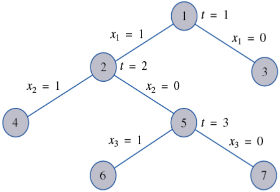
Use implicit enumeration to solve the following $0-1$ IP:

$$\begin{aligned} \max \quad & -7x_1 - 3x_2 - 2x_3 - x_4 - 2x_5 \\ \text{s.t.} \quad & -4x_1 - 2x_2 + x_3 - 2x_4 - x_5 \leq -3 \\ & -4x_1 - 2x_2 - 4x_3 + x_4 + 2x_5 \leq -7 \end{aligned}$$

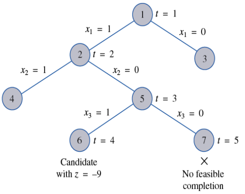
- For x_1, check the best completion (infeasible) and feasibility (feasible).
- For x_2, check the best completion (infeasible) and feasibility (feasible).
- For x_3, check the best completion (infeasible) and feasibility (feasible).

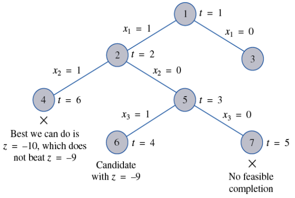


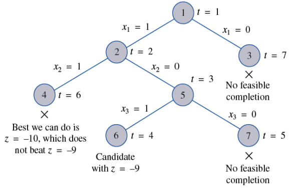




- Using the LIFO rule, for node 6, the best completion yields $z = -9$ (feasible). Using the LIFO rule, The best completion of node 7 yields $z = -7$. However, it has no feasible completion, and it may be eliminated from consideration. So does node 4.
- Node 3 has no completion and may be eliminated from consideration.
- Because there are no nodes left to analyze, the node 6 candidate with $z = -9$ must be optimal.







For the Telfa Corporation problem, after adding slack variables s_1 and s_2, we found the optimal tableau for the LP relaxation as shown below.

Optimal Tableau for LP Relaxation of Telfa

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

To apply the cutting plane method, we begin by choosing any constraint in the LP relaxation's optimal tableau in which a basic variable is fractional.

Definition 4.1

Define $[x]$ to be the largest integer less than or equal to x.

$$\begin{aligned} & x_1 - 1.25 s_1 + 0.25 s_2 = 3.75 \\ \Rightarrow & x_1 - 2 s_1 + 0 s_2 - 3 = 0.75 - 0.75 s_1 - 0.25 s_2 \end{aligned} \tag{1}$$

A **cut** (2) is then added to the LP relaxation's optimal tableau:

$$0.75 - 0.75s_1 - 0.25s_2 \leq 0. \tag{2}$$

The cut generated by this method has two properties:

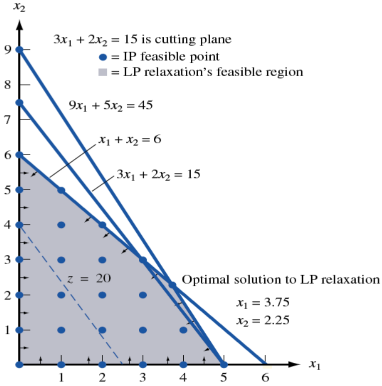
- Any feasible point for the IP will satisfy the cut.
- The current optimal solution to the LP relaxation will not satisfy the cut.

Proof.

Consider any point that is feasible for the IP. For such a point, x_1 and x_2 take on integer values, and the point must be feasible in the LP relaxation. Thus, any feasible point should satisfy (1). Any feasible solution to the IP must have $s_1 \geq 0$ and $s_2 \geq 0$. Because $0.75 < 1$, any feasible solution to the IP will make the rhs of (1) less than 1. For any feasible point to the IP, the rhs of (1) will be an integer. Therefore, for any feasible point to the IP, (2) holds. $\square$

contd.

Since the current optimal solution to the LP relaxation has $s_1 = s_2 = 0$, it cannot satisfy (2). Thus, if we choose any constraint whose right-hand side in the optimal tableau is fractional, we can cut off the LP relaxation's optimal solution. $\square$



Cut (2) may be written as $-0.75s_1 - 0.25s_2 \leq -0.75$ and we add it to the LP relaxation's optimal tableau.

Cutting Plane Tableau After Adding Cut (55)

z	x_1	x_2	s_1	s_2	s_3	rhs
1	0	0	1.25	0.75	0	41.25
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

The dual simplex ratio test indicates that s_1 should enter the basis in the third constraint. The resulting tableau yields the optimal solution $z = 40$, $x_1 = 5$, and $x_2 = 0$.

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