



Introduction

Method of Integer
Forms for IP

Mixed cut for MILP

Row generation for the
TSP

Lecture 5

Cutting Plane Method

MATH3220 Operations Research and Logistics
Jan. 20, 2015

Pan Li
The Chinese University of Hong Kong

Agenda

- 1 Introduction
- 2 Method of Integer Forms for IP
- 3 Mixed cut for MILP
- 4 Row generation for the TSP



Introduction

Method of Integer
Forms for IP

Mixed cut for MILP

Row generation for the
TSP



- The basic idea of the *cutting plane method* is to cut off parts of the feasible region of the LP relaxation, so that the optimal integer solution becomes an extreme point and therefore can be found by the simplex method.

- The cutting plane algorithm in its general form can then be formulated as follows:

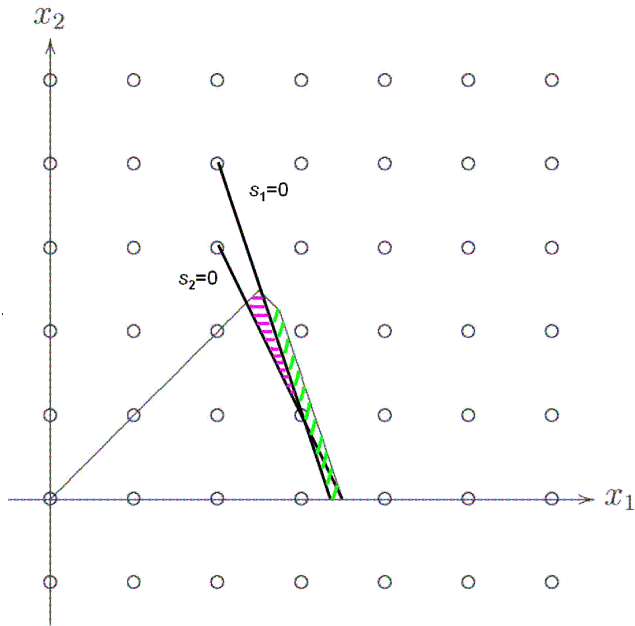
Step 1: Solve the LP relaxation. Get x^* .

Step 2: If x^* integral stop, else find a valid inequality that will exclude x^* .

Step 3: Go to step 1.

- In practice, the branch-and-bound method almost outperform the cutting plane method.
- Cutting plane is the first algorithm developed for integer programming that could be proved to converge in a finite number of steps.
- Even though the algorithm is considered not efficient, it has provided insights into integer programming that have led to other, more efficient, algorithms.

Example



Method of Integer Forms for IP



We assume all problems are comprised of rational data (notice that all numbers stored in digital computers are rational) and since rational data can be transformed into integral data by multiplying by a suitable constant, we shall only consider problems comprised of integral data.

Let an ILP be given. Assume that a non-integer optimal solution \mathbf{x}^* of the corresponding LP has been found, and that the optimal tableau is written as

	$\cdots x_r \cdots$	$\cdots x_j \cdots$	solution
x_r	1	$\cdots y_{ri} \cdots$	x_r^*
z	0		z^*

where $\{x_r : r \in J\}$ are the basic variables and $\{x_i : i \notin J\}$ are the non-basic variables.

Let us denote the set of feasible integer solutions by \mathcal{F}_I .



Suppose that \mathbf{x}^* has a non-integer component x_r^* . Then for any $\mathbf{x} \in \mathcal{F}_I$, from the r th constraint, we have

$$x_r + \sum_{i \notin J} y_{ri} x_i = x_r^*. \quad (1)$$

Since $\mathbf{x} \geq 0$,

$$x_r + \sum_{i \notin J} \lfloor y_{ri} \rfloor x_i \leq x_r^*. \quad (2)$$

Since for $\mathbf{x} \in \mathcal{F}_I$, it has integer components, the L.H.S. of (2) must be an integer. Thus

$$x_r + \sum_{i \notin J} \lfloor y_{ri} \rfloor x_i \leq \lfloor x_r^* \rfloor. \quad (3)$$

Subtracting (3) from (1), we get

$$\sum_{i \notin J} (y_{ri} - \lfloor y_{ri} \rfloor) x_i \geq x_r^* - \lfloor x_r^* \rfloor. \quad (4)$$

Letting $y_{ri} = \lfloor y_{ri} \rfloor + f_{ri}$ and $x_r^* = \lfloor x_r^* \rfloor + f_r$, we obtain (a Gomory cut)

$$\sum_{i \notin J} f_{ri} x_i \geq f_r, \quad \forall \mathbf{x} \in \mathcal{F}_I. \quad (5)$$

Hence (5) does not exclude any $\mathbf{x} \in \mathcal{F}_I$.



The current optimal solution will be excluded by (5)

$$\sum_{i \notin J} f_{ri} x_i \geq f_r, \quad \forall \mathbf{x} \in \mathcal{F}_I. \quad (5)$$

$$x_r^* - \lfloor x_r^* \rfloor = f_r > 0.$$

\Rightarrow the right hand side of (5) is positive.

However, at the current optimal solution \mathbf{x}^* , $x_i = 0$ for all $i \notin J$, and hence

$$\sum_{i \notin J} f_{ri} x_i = 0 \quad \forall i \notin J$$

Thus the current optimal solution is excluded by (5) as it does not satisfy (5).



Define the surplus variable of (5) as s :

$$s := -f_r + \sum_{i \notin J} f_{ri} x_i \text{ with } s \geq 0. \quad (6)$$

We note that s must be integral for all $\mathbf{x} \in \mathcal{F}_I$ since by (1),

$$x_r = -(-f_r + \sum_{i \notin J} f_{ri} x_i) + ([x_r^*] - \sum_{i \notin J} [y_{ri}] x_i) = -s + ([x_r^*] - \sum_{i \notin J} [y_{ri}] x_i)$$

and both x_r and the expression inside the second parenthesis on the R.H.S. are integral.

The Gomory Method

How to use valid inequality? Consider the integer program

$$\max\{cx : Ax \leq b, x \geq 0 \text{ and integer} \}$$

Idea:

First solve the associated LP relaxation and then generate a C-G inequality on the constraint associated with the basic variable that is not integer (if there is any), resolve the new LP relaxation...





Example

Given a problem in standard form:

$$\begin{array}{l} \min \\ \text{subject to} \end{array} \quad \begin{cases} z = -2x_1 - x_2 \\ x_1 + x_2 + x_3 = 5, \\ -x_1 + x_2 + x_4 = 0, \\ 6x_1 + 2x_2 + x_5 = 21, \\ x_1, x_2, \dots, x_5 \geq 0 \text{ and integer.} \end{cases}$$

Solve the LP which yields the following optimal tableau:

	x_1	x_2	x_3	x_4	x_5	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	$\frac{9}{4}$
x_4	0	0	-2	1	$\frac{1}{2}$	$\frac{1}{2}$
z	0	0	$-\frac{1}{2}$	0	$-\frac{1}{4}$	$-\frac{31}{4}$

Note that it is optimal because all the reduced cost coefficients are nonpositive (recall that it is a minimization problem).



Example

	x_1	x_2	x_3	x_4	x_5	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	$\frac{9}{4}$
x_4	0	0	-2	1	$\frac{1}{2}$	$\frac{1}{2}$
z	0	0	$-\frac{1}{2}$	0	$-\frac{1}{4}$	$-\frac{31}{4}$

Each row could serve as the source of a cut, or simply as a *source row*.

Using $\sum_{i \in J} f_{ri} x_i \geq f_r$ to generate a cut yields

$$\begin{aligned}
 x_3/2 + x_5/4 &\geq 3/4 && (x_1 \text{ row}), \\
 x_3/2 + 3x_5/4 &\geq 1/4 && (x_2 \text{ row}), \\
 x_5/2 &\geq 1/2 && (x_4 \text{ row}).
 \end{aligned} \tag{7}$$



Example

Suppose we add the x_4 row cut, or

$$-x_5/2 + s_1 = -1/2, \quad s_1 \geq 0$$

First, from the tableau, we have

$$-2x_3 + x_4 + \frac{1}{2}x_5 = \frac{1}{2}. \quad (8)$$

This is to be satisfied by all $\mathbf{x} \in \mathcal{F}$, and in particular by all $\mathbf{x} \in \mathcal{F}_I$.

$$\because x_5 \geq 0, \Rightarrow -2x_3 + x_4 \leq \frac{1}{2} \quad \Rightarrow \quad -2x_3 + x_4 \leq 0.$$

Subtracting that from (8), we have the cutting-plane

$$x_5/2 \geq 1/2 \quad (7)$$

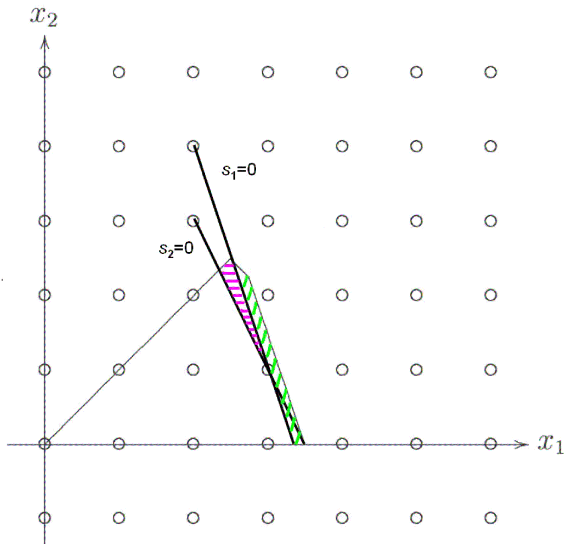
Note that since we derive (7) by assuming that $\mathbf{x} \in \mathcal{F}_I$, we do not exclude any $\mathbf{x} \in \mathcal{F}_I$.

Moreover, since at the current optimal solution, we have $x_5 = 0$. Hence $x_5/2 \not\geq 1/2$. That means the current optimal solution will be excluded by the cutting-plane.

Example

Gomory cut in terms of x_1 and x_2 :

$$x_5 \geq 1 \Rightarrow (21 - 6x_1 - 2x_2) \geq 1 \Rightarrow 10 \geq 3x_1 + x_2$$





Example

Add the cutting plane, yields the next table with $s_1 = -1/2$ as a new basic variable.

	x_1	x_2	x_3	x_4	x_5	s_1	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	0	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	0	$\frac{9}{4}$
source x_4	0	0	-2	1	$\frac{1}{2}$	0	$\frac{1}{2}$
cut s_1	0	0	0	0	$-\frac{1}{2}$	1	$-\frac{1}{2}$
z	0	0	$-\frac{1}{2}$	0	$-\frac{1}{4}$	0	$-\frac{31}{4}$

Dual feasibility but not primal feasibility is retained.

⇒ Reoptimize using Dual simplex method.

⇒ s_1 leaves and x_5 enters the basis.

Example

After pivoting, we obtain the next table (excluding the s_2 row and column):

		x_1	x_2	x_3	x_4	x_5	s_1	s_2	solution
source	x_1	1	0	$-\frac{1}{2}$	0	0	$\frac{1}{2}$	0	$\frac{10}{4}$
	x_2	0	1	$\frac{3}{2}$	0	0	$-\frac{1}{2}$	0	$\frac{10}{4}$
	x_4	0	0	-2	1	0	1	0	0
	x_5	0	0	0	0	1	-2	0	1
cut	s_2	0	0	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$	1	$-\frac{1}{2}$
	z	0	0	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$	0	$-\frac{30}{4}$

$x_r^* \geq 0, \Rightarrow$ Primal feasible.

Reduced cost coefficients $\leq 0, \Rightarrow$ tableau is optimal.

The current optimal solution is

$(x_1, x_2, x_3, x_4, x_5, s_1) = (10/4, 10/4, 0, 0, 1, 0)$.



Example

		x_1	x_2	x_3	x_4	x_5	s_1	s_2	solution
source	x_1	1	0	$-\frac{1}{2}$	0	0	$\frac{1}{2}$	0	$\frac{10}{4}$
	x_2	0	1	$\frac{3}{2}$	0	0	$-\frac{1}{2}$	0	$\frac{10}{4}$
	x_4	0	0	-2	1	0	1	0	0
	x_5	0	0	0	0	1	-2	0	1
cut	s_2	0	0	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$	1	$-\frac{1}{2}$
	z	0	0	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$	0	$-\frac{30}{4}$

We can choose x_1 or x_2 row as a source row. Arbitrarily selecting the x_1 row, we obtain

$$x_3/2 + s_1/2 \geq 1/2.$$

Thus we add the constraint

$$-x_3/2 - s_1/2 + s_2 = -1/2$$

to the table above and reoptimize.



Example

After pivoting, we obtain

		x_1	x_2	x_3	x_4	x_5	s_1	s_2	solution
source	x_1	1	0	0	0	0	1	1	3
	x_2	0	1	0	0	0	-2	3	1
	x_4	0	0	0	1	0	3	-4	2
	x_5	0	0	0	0	1	-1	0	1
cut	x_3	0	0	1	0	0	1	-2	1
	z	0	0	0	0	0	0	-1	-7

The solution is primal feasible, optimal and integral; therefore it is optimal to the IP.

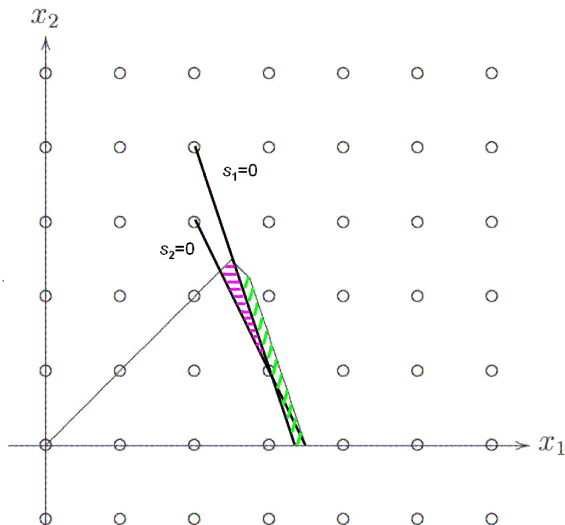


Example

Gomory cut in terms of x_1 and x_2 :

$$x_3 + s_1 \geq 1 \Rightarrow 2x_3 + (x_5 - 1) \geq 2$$

$$\Rightarrow 2(5 - x_1 - x_2) + (21 - 6x_1 - 2x_2 - 1) \geq 2 \Rightarrow 7 \geq 2x_1 + x_2.$$



Mixed cut for MILP

Let x_r be an integer variable of the MILP.

Consider the x_r -equation in the optimal continuous solution.

This is given by

$$x_r + \sum_i y_{ri} x_i = x_r^* \quad (\text{Source row})$$

and

$$x_r + \sum_i y_{ri} x_i = \lfloor x_r^* \rfloor + f_r \quad (9)$$

Because some of the nonbasic variables may not be restricted to integer values in this case, it is incorrect to use the cut developed in the preceding section.

For x_r to be integer, either $x_r \leq \lfloor x_r^* \rfloor$ or $x_r \geq \lfloor x_r^* \rfloor + 1$ must be satisfied. From the source row, these conditions are equivalent to:

$$\sum_i y_{ri} x_i \geq f_r \quad (10)$$

$$\sum_i y_{ri} x_i \leq f_r - 1 \quad (11)$$





Let
 J^+ = set of subscripts of non-basic variables for which $y_{ri} \geq 0$
 J^- = set of subscripts of non-basic variables for which $y_{ri} < 0$

Then from the above inequalities we obtain the following two inequalities

$$\sum_{i \in J^+} y_{ri} x_i \geq f_r \quad (12)$$

$$\left(\frac{f_r}{f_r - 1} \right) \sum_{i \in J^-} y_{ri} x_i \geq f_r \quad (13)$$

One of these two inequalities must hold, hence we can combine them to give the inequality:

$$\sum_{i \in J^+} y_{ri} x_i + \left(\frac{f_r}{f_r - 1} \right) \sum_{i \in J^-} y_{ri} x_i \geq f_r$$

By adding one slack variable s , it becomes the "mixed cut equation":

$$s - \left(\sum_{i \in J^+} y_{ri} x_i + \left(\frac{f_r}{f_r - 1} \right) \sum_{i \in J^-} y_{ri} x_i \right) = -f_r$$

[Introduction](#)

[Method of Integer
Forms for IP](#)

[Mixed cut for MILP](#)

[Row generation for the
TSP](#)

Example

$$\begin{array}{ll} \max & z = 2x_1 + x_2 \\ \text{subject to} & \left\{ \begin{array}{l} x_1 + x_2 + x_3 = 5, \\ -x_1 + x_2 + x_4 = 0, \\ 6x_1 + 2x_2 + x_5 = 21, \\ x_1, x_2, \dots, x_5 \geq 0 \\ x_1 \text{ integer} \end{array} \right. \end{array}$$

Solve the LP which yields the following optimal tableau:

	x_1	x_2	x_3	x_4	x_5	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	$\frac{9}{4}$
x_4	0	0	-2	1	$\frac{1}{2}$	$\frac{1}{2}$
z	0	0	$\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{31}{4}$



Example

	x_1	x_2	x_3	x_4	x_5	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	$\frac{9}{4}$
x_4	0	0	-2	1	$\frac{1}{2}$	$\frac{1}{2}$
z	0	0	$\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{31}{4}$

From the x_1 -row in tableau, the mixed cut is

$$s_1 - \left\{ \frac{1}{4}x_5 + \left(\frac{\frac{3}{4}}{\frac{3}{4} - 1} \right) \left(-\frac{1}{2} \right) x_3 \right\} = -\frac{3}{4},$$

or

$$s_1 - \frac{1}{4}x_5 - \frac{3}{2}x_3 = -\frac{3}{4}$$



Example

	x_1	x_2	x_3	x_4	x_5	s_1	solution
x_1	1	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	0	$\frac{11}{4}$
x_2	0	1	$\frac{3}{2}$	0	$-\frac{1}{4}$	0	$\frac{9}{4}$
source x_4	0	0	-2	1	$\frac{1}{2}$	0	$\frac{1}{2}$
cut s_1	0	0	$-\frac{3}{2}$	0	$-\frac{1}{4}$	1	$-\frac{3}{4}$
z	0	0	$\frac{1}{2}$	0	$\frac{1}{4}$	0	$\frac{31}{4}$

Applying the dual simplex method we obtain finally the tableau.

	x_1	x_2	x_3	x_4	x_5	s_1	solution
x_1	1	0	0	0	$\frac{1}{3}$	$-\frac{1}{3}$	3
x_2	0	1	0	0	$-\frac{1}{2}$	1	$\frac{3}{2}$
source x_4	0	0	0	1	$\frac{5}{6}$	$-\frac{4}{3}$	$\frac{3}{2}$
cut x_3	0	0	1	0	$\frac{1}{6}$	$-\frac{2}{3}$	$\frac{1}{2}$
z	0	0	0	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{15}{2}$

\Rightarrow Optimal solution $x_1 = 3, x_2 = 3/2$ and the maximum value of the objective function is $15/2$.





Example

Question: Can you write down the "geometric equation" of the MILP cut?

Doing this, you can see better why the cut only cut out x_1 and does not affect x_2 .

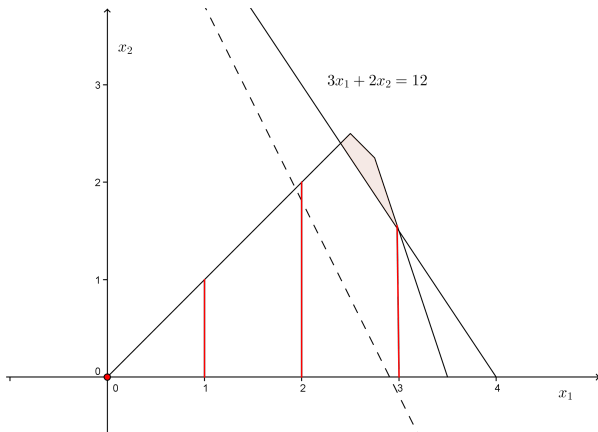
$$\begin{array}{l} \max \\ \text{subject to} \end{array} \quad \begin{array}{l} z = 2x_1 + x_2 \\ \left\{ \begin{array}{l} x_1 + x_2 + x_3 = 5, \\ -x_1 + x_2 + x_4 = 0, \\ 6x_1 + 2x_2 + x_5 = 21, \\ x_1, x_2, \dots, x_5 \geq 0 \\ x_1 \text{ integer} \end{array} \right. \end{array}$$

$$s_1 - \frac{1}{4}x_5 - \frac{3}{2}x_3 = -\frac{3}{4}$$

Example

The cut is

$$\begin{aligned} \frac{3}{2}x_3 + \frac{1}{4}x_5 &\geq \frac{3}{4} \\ \Rightarrow \frac{3}{2}(5 - x_1 - x_2) + \frac{1}{4}(21 - 6x_1 - 2x_2) &\geq \frac{3}{4} \\ \Rightarrow 3x_1 + 2x_2 &\leq 12 \end{aligned}$$



Disadvantages of Cutting plane method

- (i) When this algorithm is implemented on digital computer, round-off errors may cause serious problem, especially in distinguishing integers from non-integers.
- (ii) The solution of the problem remains infeasible until the optimal integer solution is reached. This means that there will be no “good” integer solution on hand, if the calculations are stopped prematurely owing to budget or time limitations.



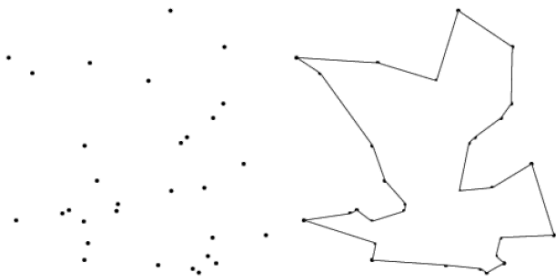
Cutting Plane algorithm

- Overall strategy:
 - 1 Solve LP relaxation, get relaxed solution x^* .
 - 2 If x^* is a feasible solution of P , exit.
 - 3 Use solution x^* of the relaxed one to construct a valid cut C for P .
 - 4 Add the constraint C to the formulation of P .
 - 5 Go back to 1.
- Crux of the algorithm: step 3 (separation problem)
- Cutting plane algorithms may depends on the particular problem structure of be completely general.
- Independent of problem structure: Gomory cutting planes
- Problem structure: Row generation for the TSP



Travelling Salesman Problem

Given a collection of cities and the cost of travel between each pair of them, the travelling salesman problem, or TSP for short, is to find the cheapest way of visiting all the cities and returning to the starting point.



Comments on TSP

- First formulated as a mathematical problem in 1930 and is one of the most intensively studied problems in optimization.
- Applications: planning, logistics, microchip design, DNA sequencing, and more.
- TSP formulation has an exponential number of constraints.
- The straight-forward application of branch-and-bound method can process TSPs containing 40-60 cities.
- Using branch-and-bound and problem-specific cut generation, currently it can solve a TSP with 85,900 cities.



An integer programming formulation

Suppose there are n cities. Let d_{ij} be the distance between cities i and j , where $d_{ij} = \infty$. Define

$$x_{ij} = \begin{cases} 1, & \text{if city } j \text{ is reached from city } i \\ 0, & \text{otherwise.} \end{cases}$$

Then, the objective function of TSP can be written as

$$\min z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}$$

Constraints:

$$\sum_{i=1}^n x_{ij} = 1, j = 1, 2, \dots, n$$

$$\sum_{j=1}^n x_{ij} = 1, i = 1, 2, \dots, n$$

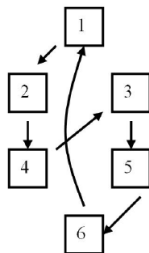
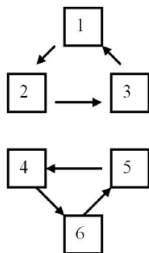
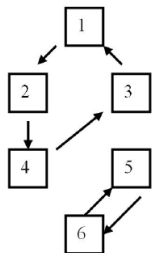
$$x_{ij} \in \{0, 1\}, i, j = 1, 2, \dots, n$$

and

solution is a n -city tour.



Subtour and tour

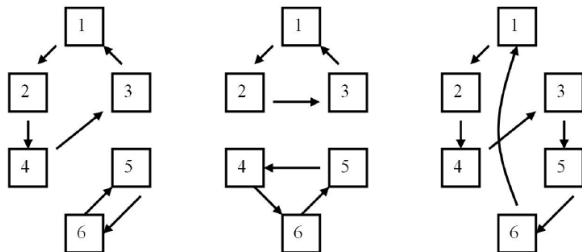


The left and center figures are subtours of the 6 cities, and are not feasible solutions to the TSP.

The right figure is a tour of 6 cities, and is a feasible solution of the TSP problem.



Break the subtour



Let $S = \{1, 2, 3\}$. If they form a subtour, we should have

$$x_{12} + x_{21} + x_{13} + x_{31} + x_{23} + x_{32} = 3$$

If they don't form a subtour,

$$x_{12} + x_{21} + x_{13} + x_{31} + x_{23} + x_{32} \leq 3 - 1$$

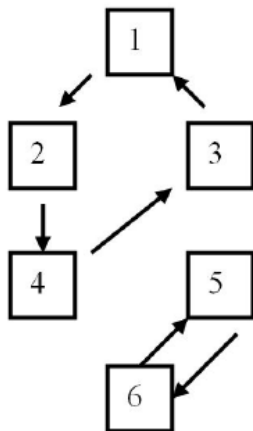
\Rightarrow

$$\sum_{i \neq j \in S} x_{ij} \leq |S| - 1 \quad \forall S \subset \{1, 2, \dots, n\}$$



Exercise

- Write a subtour elimination constraint that can cut off the following solution.



Row generation for TSP

- TSP formulation has an exponential number of constraints (one for each proper subset of $\{1, 2, \dots, n\}$)
- LP relaxation solution becomes unmanageable as n grows.
- Try relaxing (i.e., removing) problematic constraints

$$\sum_{i \neq j \in S} x_{ij} \leq |S| - 1 \quad \forall S \subset \{1, 2, \dots, n\}$$

\Rightarrow TSP becomes an *Assignment Problem* and can be solved very fast.

- Algorithm:
 - 1 The LP without subtour breaking constraints is solved.
 - 2 If the LP relaxation solution is not a tour, a tour breaking constraint is added, and the LP relaxation is solved again. The branch-and-bound idea is used to break the sub-tours one by one.
 - 3 Keep going until all tour breaking constraints are satisfied by the LP optimum.



Example

Consider a 5-city TSP problem where the distance d_{ij} are given by:

$$\begin{bmatrix} \infty & 1 & 5 & 4 & 2 \\ 2 & \infty & 3 & 7 & 6 \\ 0 & 4 & \infty & 2 & 3 \\ 6 & 4 & 1 & \infty & 6 \\ 1 & 2 & 5 & 1 & \infty \end{bmatrix}$$



Example - subtour breaking

∞	1	5	4	2
2	∞	3	7	6
0	4	∞	2	3
6	4	1	∞	6
1	2	5	1	∞

∞	0	4	3	0
0	∞	1	5	3
0	4	∞	2	2
5	3	0	∞	4
0	1	4	0	∞

∞	0	6	3	0
0	∞	1	3	1
0	2	∞	0	0
5	1	0	∞	2
2	1	6	0	∞

The solution is $\{x_{12} = x_{21} = 1, x_{35} = x_{54} = x_{43} = 1\}$ with minimum cost $z = 8$.

Note that it is a solution with two subtours and hence is not a feasible solution to the TSP.

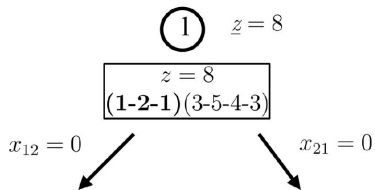
In order to form a 5-city tour, we have to break either one of the loops. Let's break the loop $1 \rightarrow 2 \rightarrow 1$.

\Rightarrow either force $x_{12} = 0$ or $x_{21} = 0$

\Rightarrow set $d_{12} = \infty$ or $d_{21} = \infty$



Example - Enumeration tree



Example - node 2

Set $d_{12} = \infty$. Solve the resulting assignment problem.

∞	∞	5	4	2
2	∞	3	7	6
0	4	∞	2	3
6	4	1	∞	6
1	2	5	1	∞

0	∞	3	2	0
0	∞	1	5	4
0	3	∞	2	3
5	2	0	∞	5
0	0	4	0	∞

0	0	5	2	0
0	∞	1	3	2
0	1	∞	0	1
5	0	0	∞	3
2	0	6	0	∞

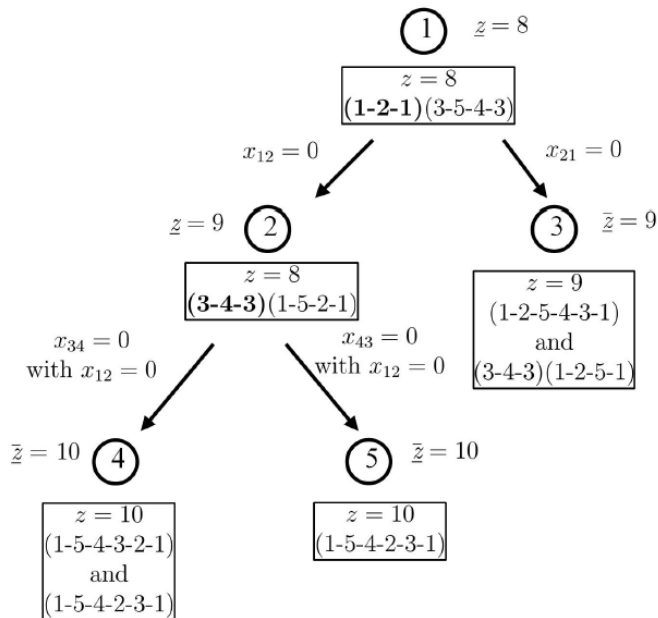
The solution is $\{x_{34} = x_{43} = 1, x_{15} = x_{52} = x_{21} = 1\}$ with minimum cost $z = 9$.

Contains subtours.

\Rightarrow Next break the subtour $3 \rightarrow 4 \rightarrow 3$.



Example - Enumeration tree





Introduction

Method of Integer
Forms for IP

Mixed cut for MILP

Row generation for the
TSP

- Dramatically improve the modeling capability.
 - Economic indivisibility
 - Logical constraint
 - Modeling non-linearities
- Not as easy to model.
- Not as easy to solve.



- Branch-and-Bound
 - very general and adaptive
 - used in practice (e.g. Excel solver)
- Cutting planes (Valid inequalities)
 - clever way of improving bounding
 - used widely in practice
 - researchers continue to make improvements in this approach