

# Asymmetric Traveling Salesman Problem (ATSP): Models

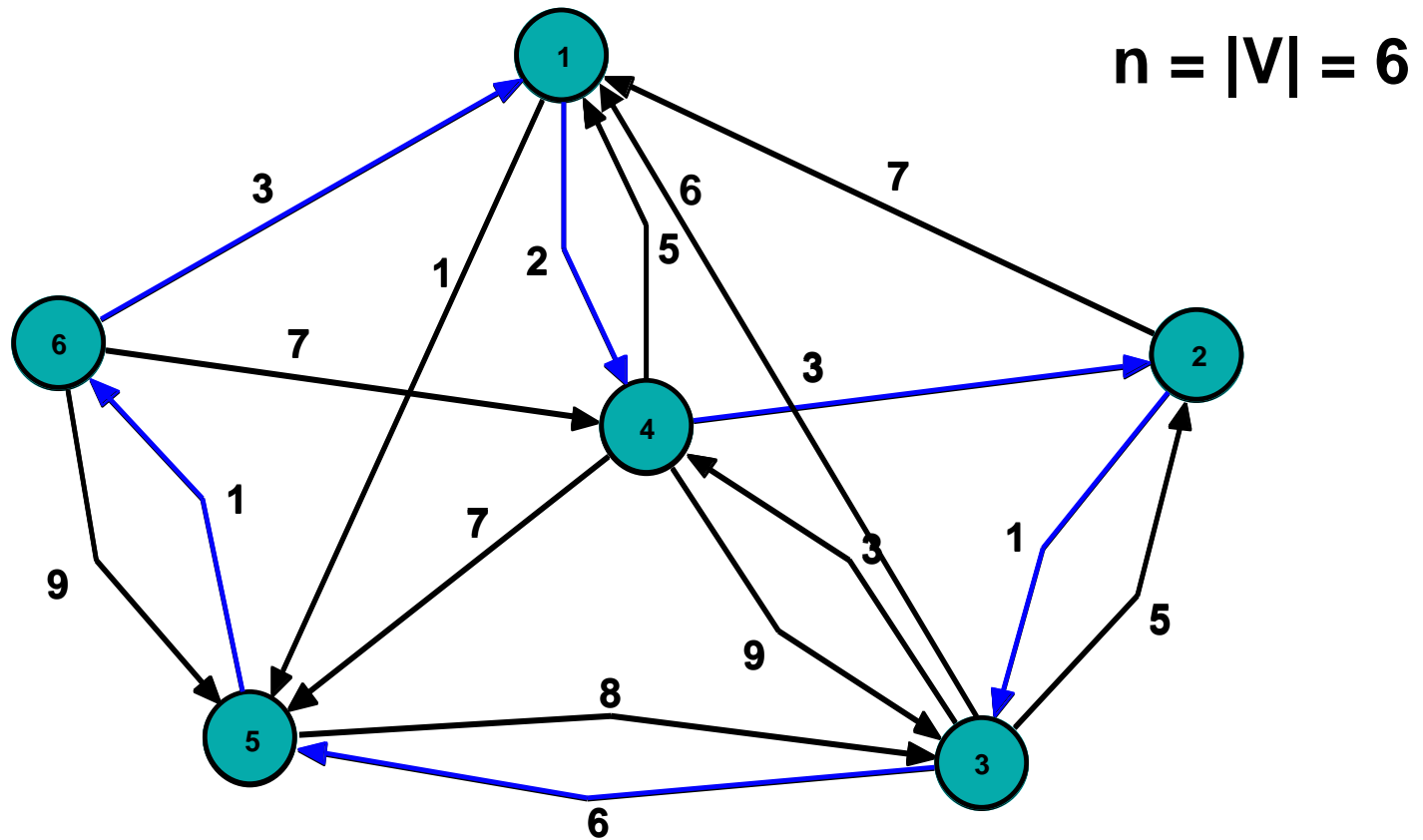
- Given a DIRECTED GRAPH  $G = (V, A)$  with
  - $V = \{1, \dots, n\}$  vertex set
  - $A = \{(i, j) : i \in V, j \in V\}$  arc set (complete digraph)
  - $c_{ij}$  = cost associated with arc  $(i, j) \in A$  ( $c_{ii} = \infty, i \in V$ )
- Find a HAMILTONIAN CIRCUIT (Tour) whose global cost is minimum (Asymmetric Travelling Salesman Problem: ATSP).

**Hamiltonian Circuit:** circuit passing through each vertex of  $V$  exactly once.

**A Hamiltonian circuit has  $n = |V|$  arcs.**

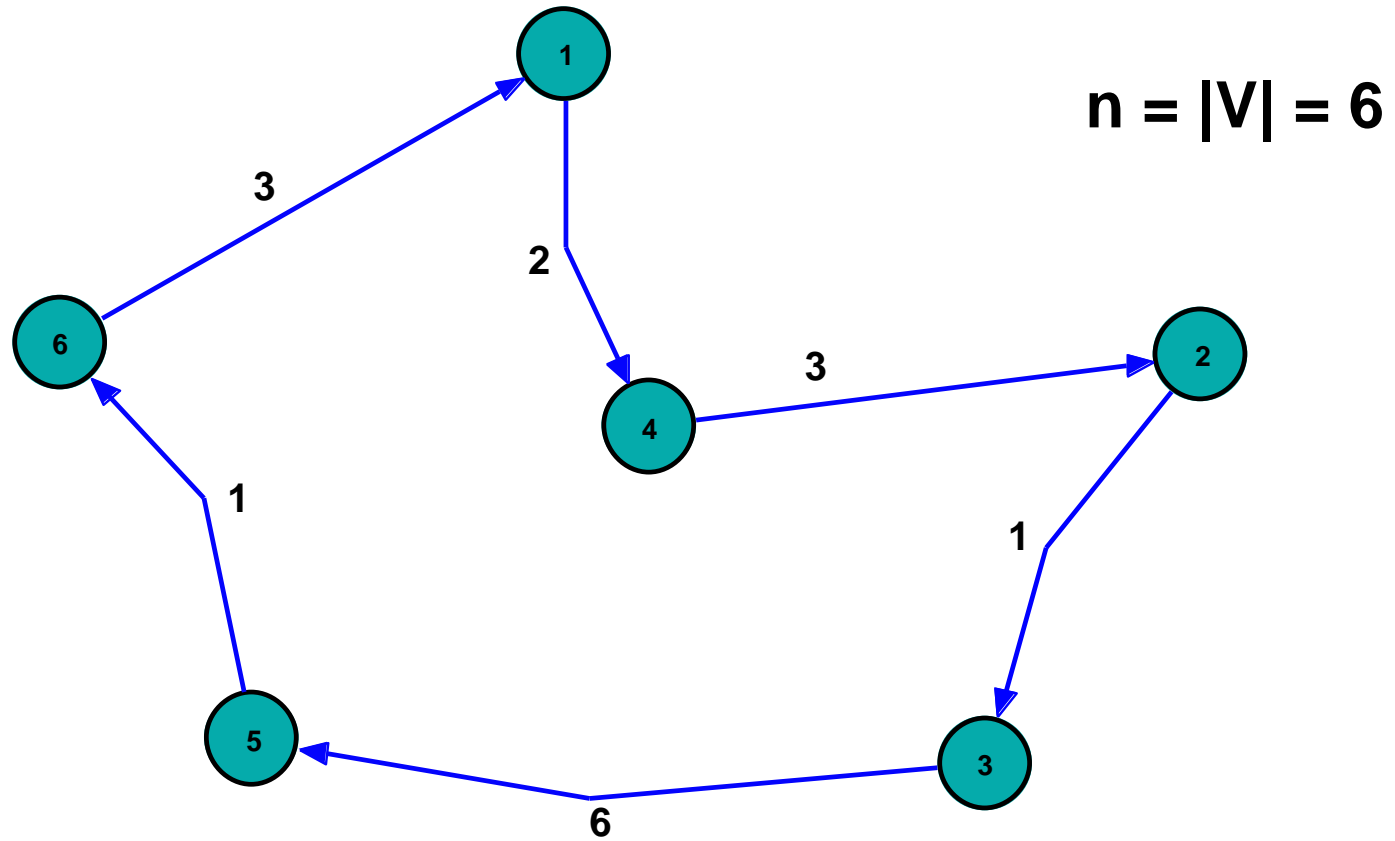
- ATSP is  $\mathcal{NP}$ -Hard in the strong sense.
- If  $G$  is an undirected graph: **Symmetric TSP (STSP)**  
(special case of ATSP arising when  $c_{ij} = c_{ji}$  for each  $(i, j) \in A$ )
- Any ATSP instance with  $n$  vertices can be transformed into an equivalent STSP instance with  $2n$  nodes (Jonker-Volgenant, 1983; Junger-Reinelt-Rinaldi, 1995; Kumar-Li, 2007).
- If  $G = (V, A)$  is a sparse graph:  $c_{ij} = \infty$  for each  $(i, j) \notin A$ .
- **Feasible solutions?**

# Example



**Optimal solution**

# Example



**Optimal solution**

**Optimal solution Cost = 2 + 3 + 1 + 6 + 1 + 3 = 16**

# APPLICATIONS

- \* **Vehicle Routing** (sequencing the customers in each route in an urban area calls for the optimal solution of the ATSP corresponding to the depot and the customers in the route).
- \* **Scheduling** (optimal sequencing of jobs on a machine when the set-up costs depend on the sequence in which the jobs are processed).
- **Picking in an Inventory System** (sequence of movements of a crane to pick-up a set of items stored on shelves).
- ...

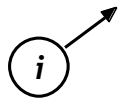
# INTEGER LINEAR PROGRAMMING (ILP) FORMULATION

$$x_{ij} = \begin{cases} 1 & \text{if arc } (i, j) \text{ is in the optimal tour} \\ 0 & \text{otherwise} \end{cases} \quad i \in V, j \in V$$

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

s.t.

out-degree  
constraints



$$\sum_{j \in V} x_{ij} = 1$$

$$i \in V$$

in-degree  
constraints



$$\sum_{i \in V} x_{ij} = 1$$

$$j \in V$$

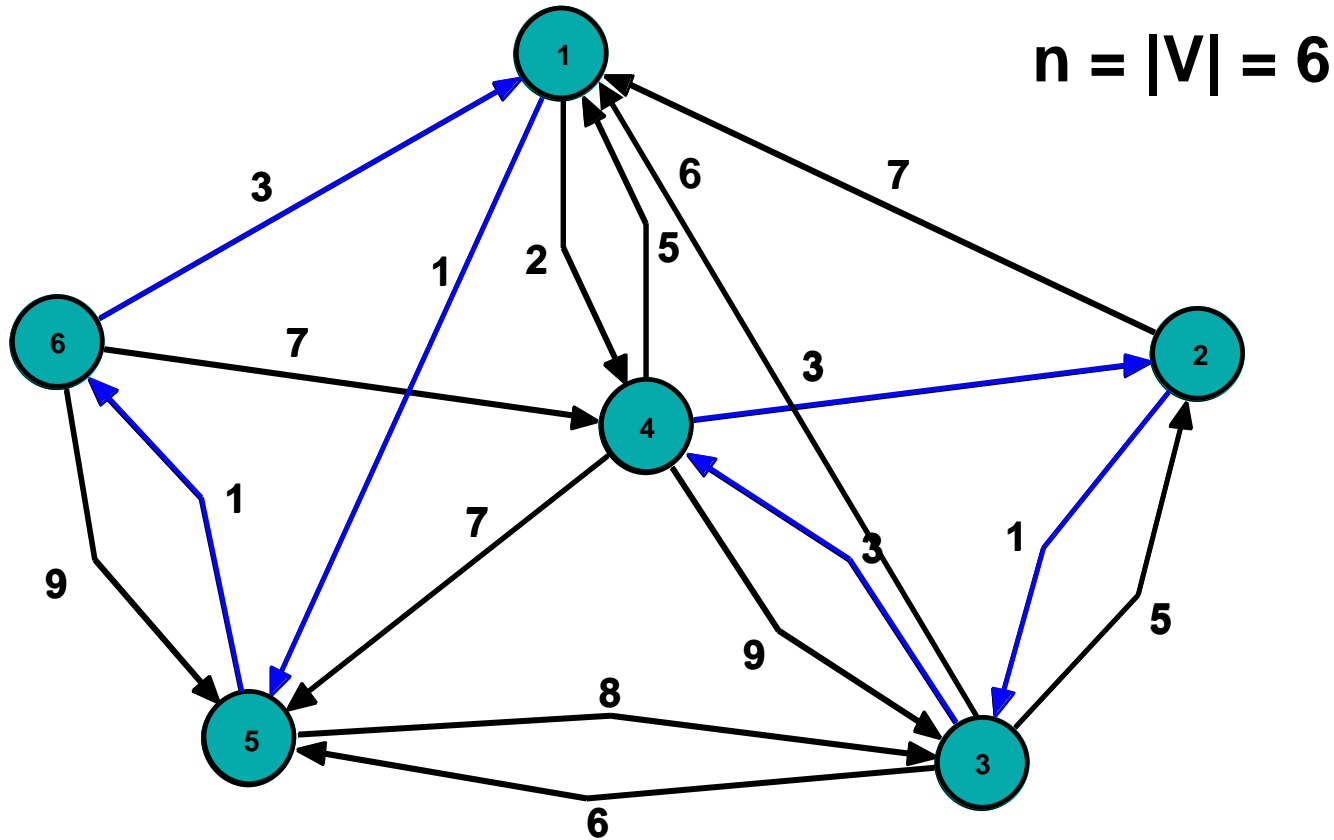
$$x_{ij} \in \{0, 1\}$$

$$i \in V, j \in V$$



Example:

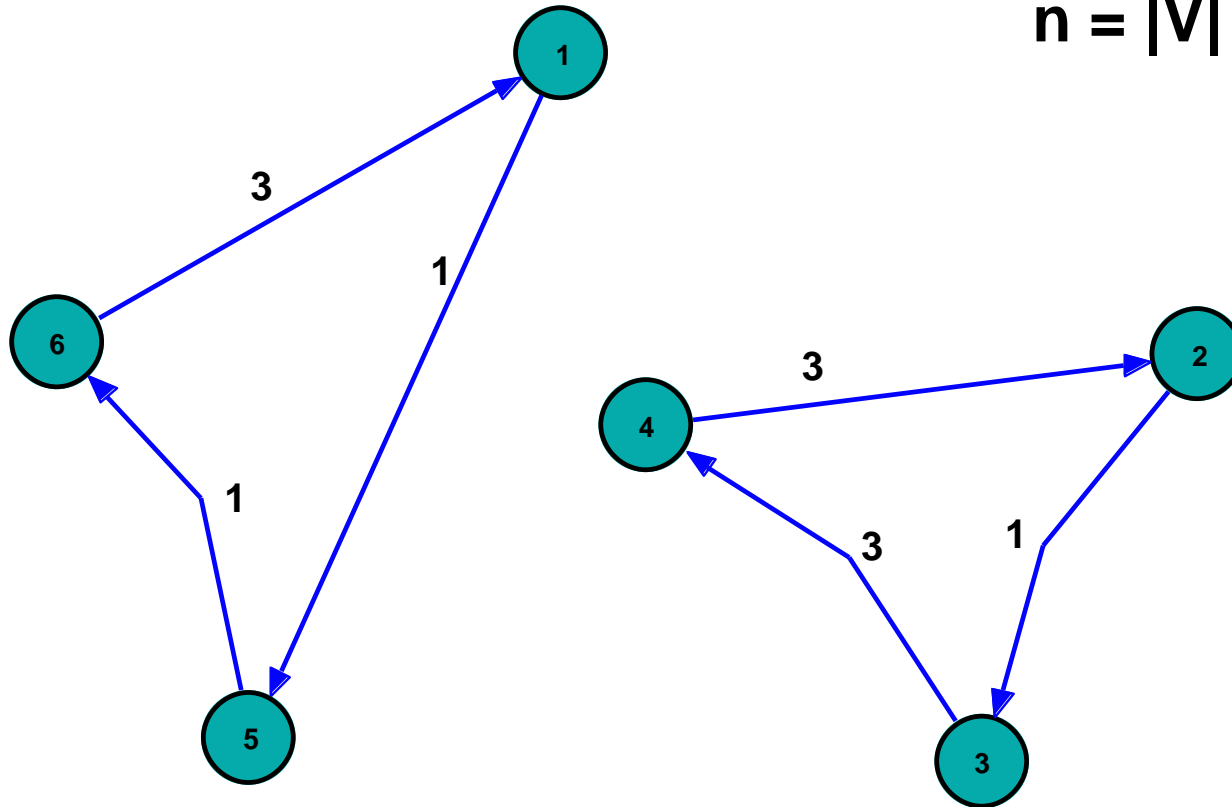
only degree constraints imposed



**Example:**

**only degree constraints imposed**

$$n = |V| = 6$$



**solution Cost =  $(1 + 1 + 3) + (3 + 1 + 3) = 12$**

**Infeasible solution: two partial tours (subtours)**



# INTEGER LINEAR PROGRAMMING (ILP) FORMULATION

## (Dantzig, Fulkerson, Johnson, Oper. Res. 1954)

$$x_{ij} = \begin{cases} 1 & \text{if arc } (i, j) \text{ is in the optimal tour} \\ 0 & \text{otherwise} \end{cases} \quad i \in V, j \in V$$

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

s.t.

out-degree constraints

$$\sum_{j \in V} x_{ij} = 1 \quad i \in V$$

in-degree constraints

$$\sum_{i \in V} x_{ij} = 1 \quad j \in V$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \quad S \subset V, |S| \geq 2$$

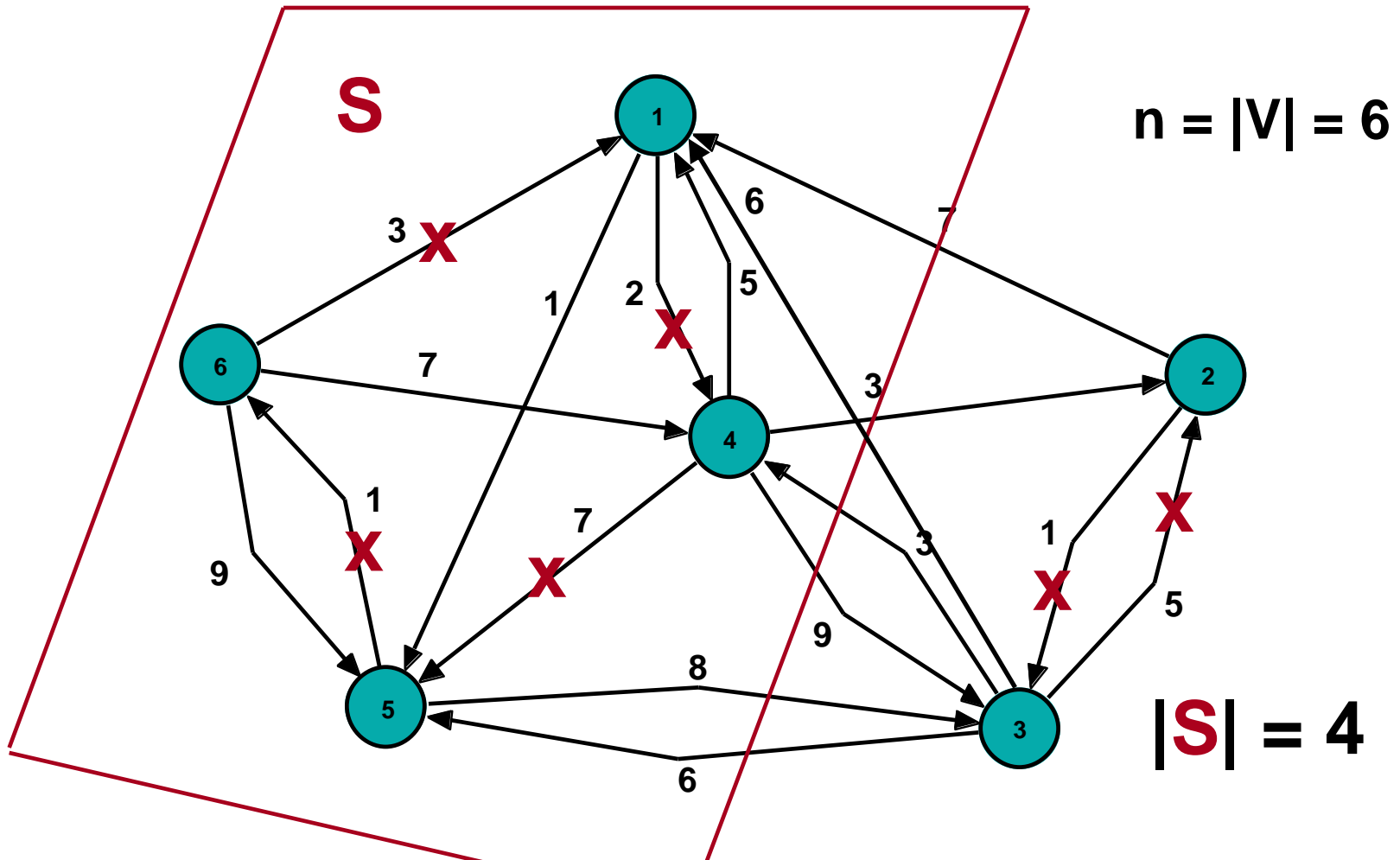
$$x_{ij} \in \{0, 1\}$$

$$i \in V, j \in V$$

SUBTOUR ELIMINATION  
CONSTRAINTS  
( forbid the partial tours;  
 $O(2^n)$  )

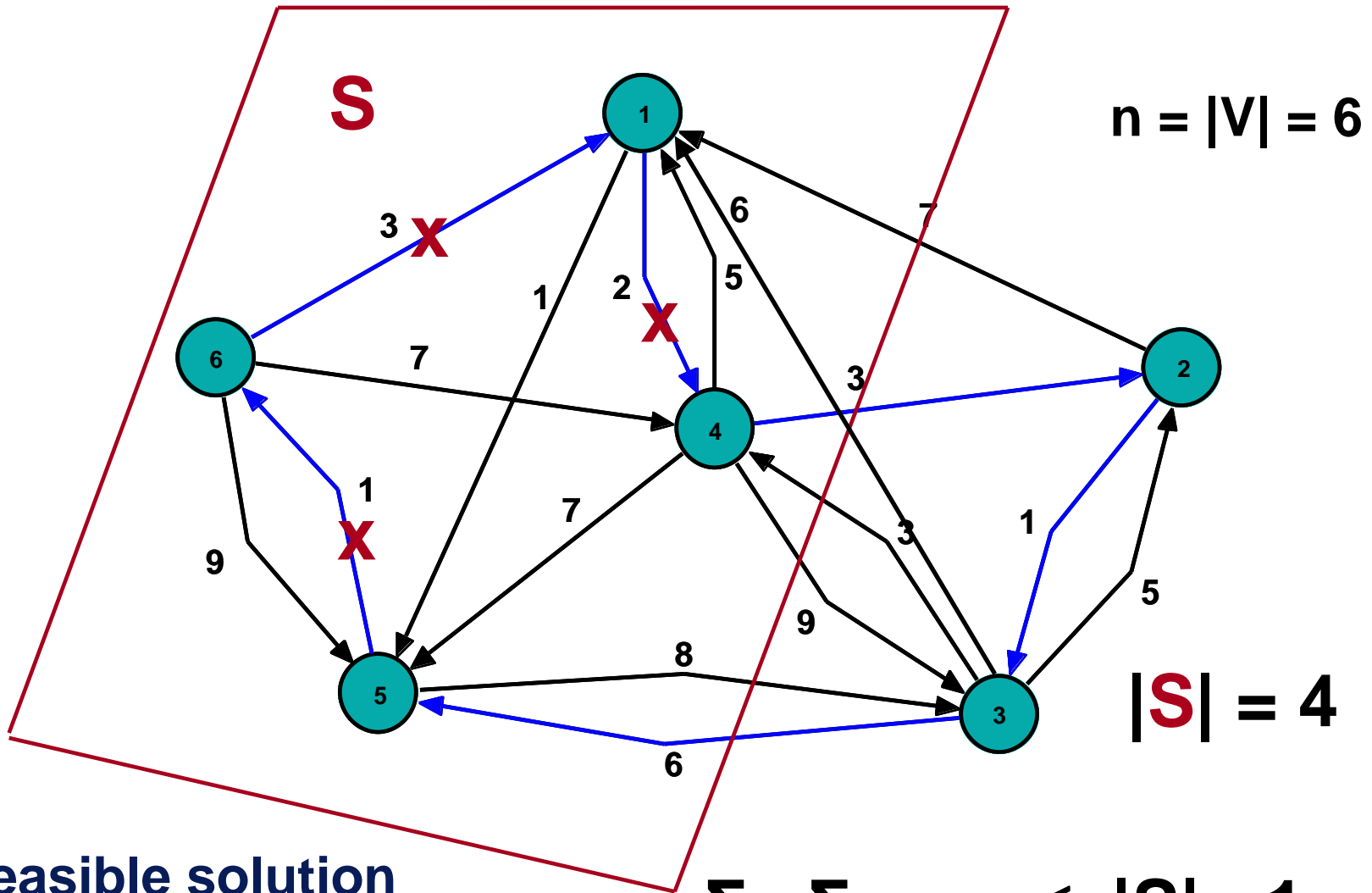
!!!

# Example



$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \quad \text{NO!}$$

# Example

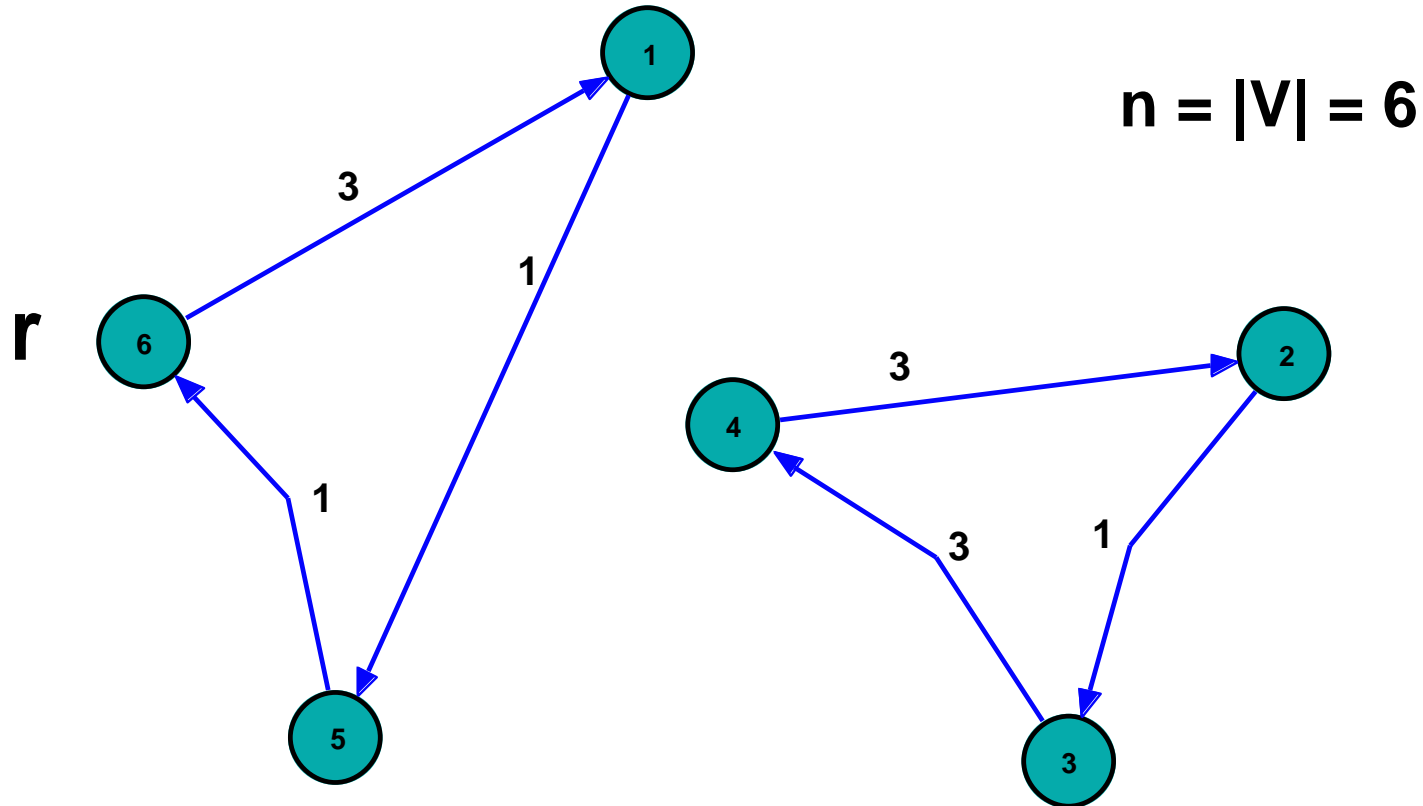


Feasible solution

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1 \quad \text{YES!}$$

**Example:**

**only degree constraints imposed**



**Infeasible solution: the vertices are not connected:**

**In a Hamiltonian circuit:**

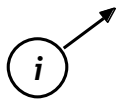
**from any vertex (say  $r$ ) we must reach all the other vertices (connectivity from  $r$ ), and viceversa (connectivity toward  $r$ )**

# INTEGER LINEAR PROGRAMMING FORMULATION

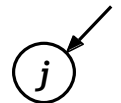
$$x_{ij} = \begin{cases} 1 & \text{if arc } (i, j) \text{ is in the optimal tour} \\ 0 & \text{otherwise} \end{cases} \quad i \in V, j \in V$$

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

s.t.



$$\sum_{j \in V} x_{ij} = 1 \quad i \in V$$



$$\sum_{i \in V} x_{ij} = 1 \quad j \in V$$

CONNECTIVITY  
CONSTRAINTS  
(impose the connectivity  
of the solution;  $O(2^n)$ )

Cut inequalities

$$\sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq 1 \quad S \subset V, r \in S$$

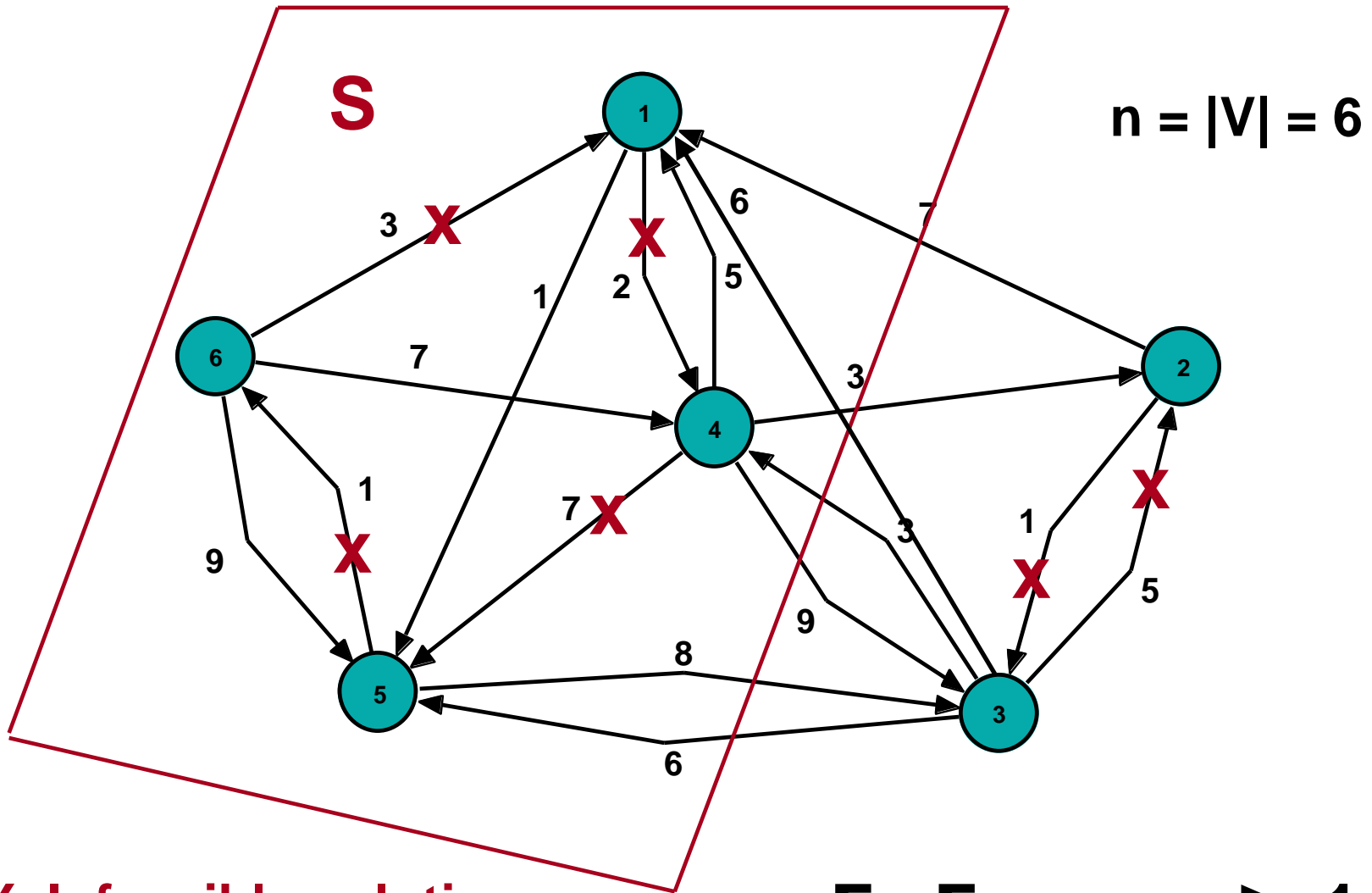
(for a fixed  $r \in V$ )

$$x_{ij} \in \{0, 1\}$$

$$i \in V, j \in V$$

The two formulations are “equivalent”

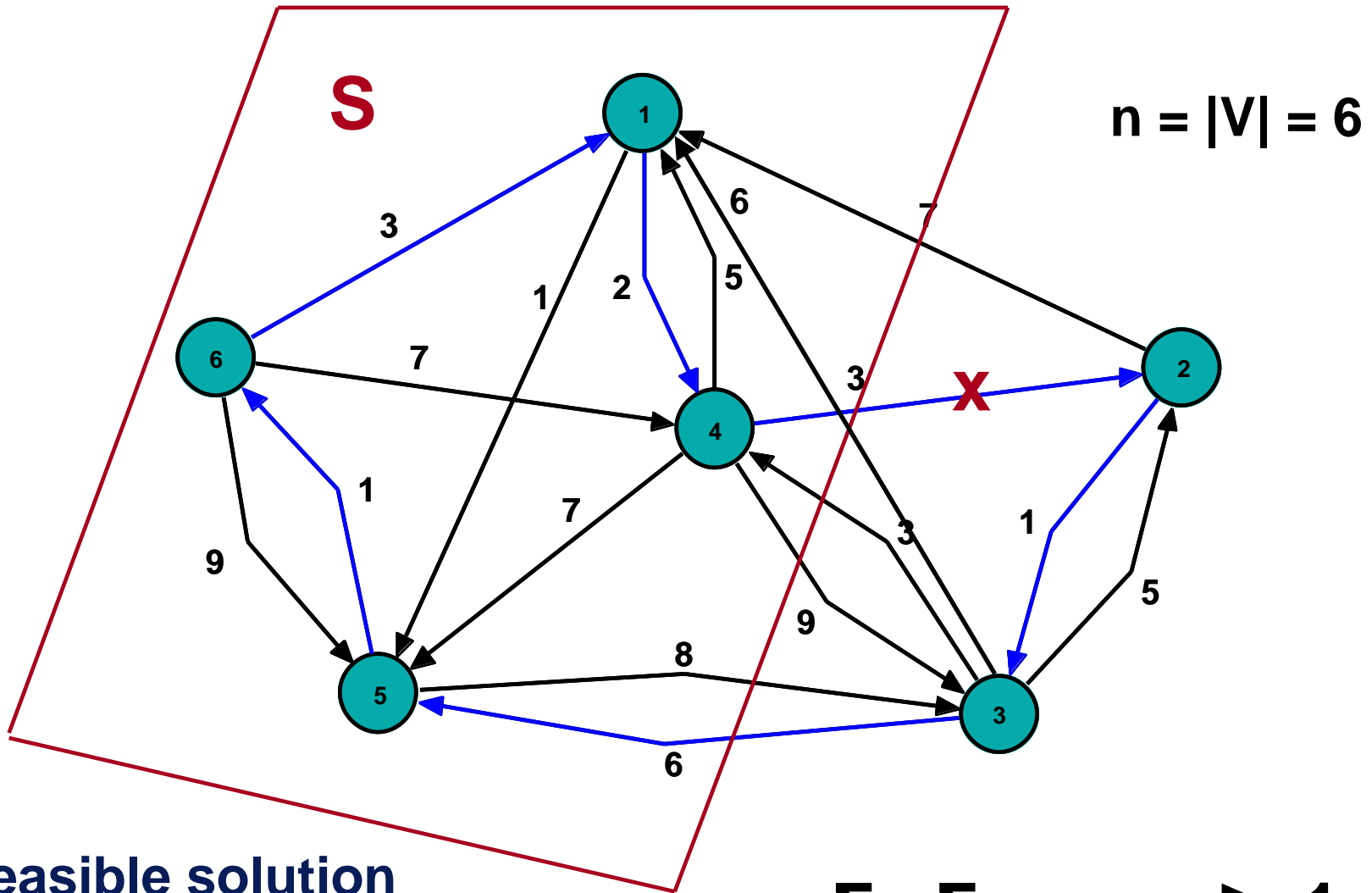
# Example



**X Infeasible solution**

$$\sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq 1 \quad \text{NO!}$$

# Example



$n = |V| = 6$

Feasible solution

$$\sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq 1 \quad \text{YES!}$$

# LP LOWER BOUND

- The value of the optimal solution of the **Linear Programming (LP) Relaxation** (or **Continuous Relaxation**) of the previous formulations, obtained by replacing

$$x_{ij} \in \{0, 1\} \quad i \in V, j \in V$$

with

$$0 \leq x_{ij} \leq 1 \quad i \in V, j \in V$$

represents a valid **Lower Bound** on the value of the optimal solution of the ATSP.



- This LP Relaxation can be **efficiently (polynomially) solved** by using appropriate **Separation Procedures (Polynomial Separation Problem)**.
- The LP Relaxation can be **strengthened** (so as to obtain better lower bounds) by adding **valid inequalities**, which are “**redundant**” for the ILP model, but can be violated by its LP Relaxation (**exact and/or heuristic “separation procedures”**).

# ALTERNATIVE ILP FORMULATIONS

- Miller-Tucker-Zemlin (J. ACM, 1960);
- Fox-Gavish-Graves (Operations Research 1980);
- Wong (IEEE Conference ..., 1980);
- Claus (SIAM J. on Algebraic Discrete Methods, 1984);
- Finke-Claus-Gunn (Congressus Numerantium, 1984);
- Langevin-Soumis-Desrosiers (Operations Research Letters, 1990)
- Desrochers-Laporte (Operations Research Letters, 1991);
- Gouveia-Voss (European Journal of Operational Research, 1995)
- Gouveia-Pires (European Journal of Operational Research, 1999);
- Myung (International Journal of Management, 2001);
- Gouveia-Pires (Discrete Applied Mathematics, 2001);
- Sherali-Driscoll (Operations Research 2002);
- Sarin-Sherali-Bhootra (Oper. Res. Letters, 2005);
- Sherali-Sarin-Tsai (Discrete Optimization, 2006);
- Oncan-Altinel-Laporte (Review, Computers & Operations Research, 2009)

# ALTERNATIVE ILP FORMULATIONS

- These formulations involve a polynomial number of constraints (“compact formulations”),

but

- their Linear Programming Relaxations generally produce Lower Bounds weaker (and more time consuming) than those corresponding to the Dantzig-Fulkerson-Johnson formulation (with the addition of valid inequalities).

# Hamiltonian Circuit Problem (HC)

- Given a (directed or undirected) graph  $G = (V, A)$  with:
    - $V = \{1, \dots, u\}$  vertex set
    - $A = \{(i, j)\}$  arc set, with  $m = |A|$ ,  $m \leq u * u$ ,  $u \leq m$
- Arc  $h: (i_h, j_h)$   $h = 1, \dots, m$
- **HC: determine if a HAMILTONIAN CIRCUIT exists in G.**
- \* **HC is known to be NP-Hard**

# *ATSP* is NP-hard

- **Input:**  $n, (c_{ij})$  : size  $1 + n^2$  :  $n^2$
- **Binary decision tree with  $n^2$  levels (variables  $x_{ij}$ )**
- *ATSP*  $\in$  *Class NP*

# *ATSP* is NP-Hard

*ATSP*  $\in$  *Class NP*

*HC*  $\propto$  *ATSP*

Given any instance of *HC*:  $u, m, A$  (Size:  $u * u$ )

1) Define (in time  $O(u * u)$ ) an instance  $(n, (c_{ij}))$  of *ATSP*:

\*  $n := u$

\*  $c_{ij} := 0$  if  $(i, j) \in A$ ,  $c_{ij} := 1$  otherwise ( $i = 1, \dots, n; j = 1, \dots, n$ ).

2) Determine the optimal solution  $(x_{ij}, z)$  of *ATSP*.

3) If  $z = 0$  : *HC* has a feasible solution  $(x_{ij})$

If  $z \geq 1$  : *HC* has a no feasible solution

Computing time  $O(n*n)$  (hence  $O(u*u)$ , polynomial in the size of *HC*)

# ***Shortest Spanning Arborescence with root $r$*** **(SSA( $r$ ))**

- **Given a complete DIRECTED GRAPH  $G = (V, A)$  with:**
  - **$V = \{1, \dots, n\}$  vertex set;  $A$  arc set;  $r \in V$ ;**
  - **$c_{ij}$  = cost associated with arc  $(i, j) \in A$  ( $c_{ii} = \infty, i \in V$ ).**

***Spanning Arborescence with root in vertex  $r$ :***

- $(n - 1)$  arcs;**
- “Connected” with respect to vertex  $r$ ;**
- “Acyclic” (with no circuit).**

***SSA( $r$ ): Find a Spanning Arborescence with root  $r$  whose global cost is minimum.***

***SSA( $r$ ) is a polynomial problem (Edmonds alg: ( $O(n^2)$ ))***

# INTEGER LINEAR PROGRAMMING FORMULATION

$$\begin{aligned} x_{ij} &= 1 && \text{if arc } (i, j) \text{ is in the optimal solution} \\ x_{ij} &= 0 && \text{otherwise} \end{aligned} \quad i \in V, j \in V$$

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

s.t.

$$\sum_{i \in V} \sum_{j \in V} x_{ij} = n - 1$$

$$\sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq 1 \quad S \subset V, r \in S$$

$$x_{ij} \in \{0, 1\}$$

$$i \in V, j \in V$$

CONNECTIVITY  
CONSTRAINTS

( impose the connectivity  
of the solution;  $O(2^n)$  )



# INTEGER LINEAR PROGRAMMING FORMULATION

$$\begin{aligned} x_{ij} &= 1 && \text{if arc } (i, j) \text{ is in the optimal solution} \\ x_{ij} &= 0 && \text{otherwise} \end{aligned} \quad i \in V, j \in V$$

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

s.t.

$$\sum_{i \in V} \sum_{j \in V} x_{ij} = n - 1 \quad (\text{redundant if } c_{ij} > 0)$$

CONNECTIVITY  
CONSTRAINTS  
( impose the connectivity  
of the solution;  $O(2^n)$  )

$$\sum_{i \in S} \sum_{j \in V \setminus S} x_{ij} \geq 1 \quad S \subset V, r \in S$$

$$x_{ij} \in \{0, 1\}$$

$$i \in V, j \in V$$