Operations Research, Jacobs University, Fall 2022 — Session 14

Prof. Sören Petrat — Oct. 18, 2022

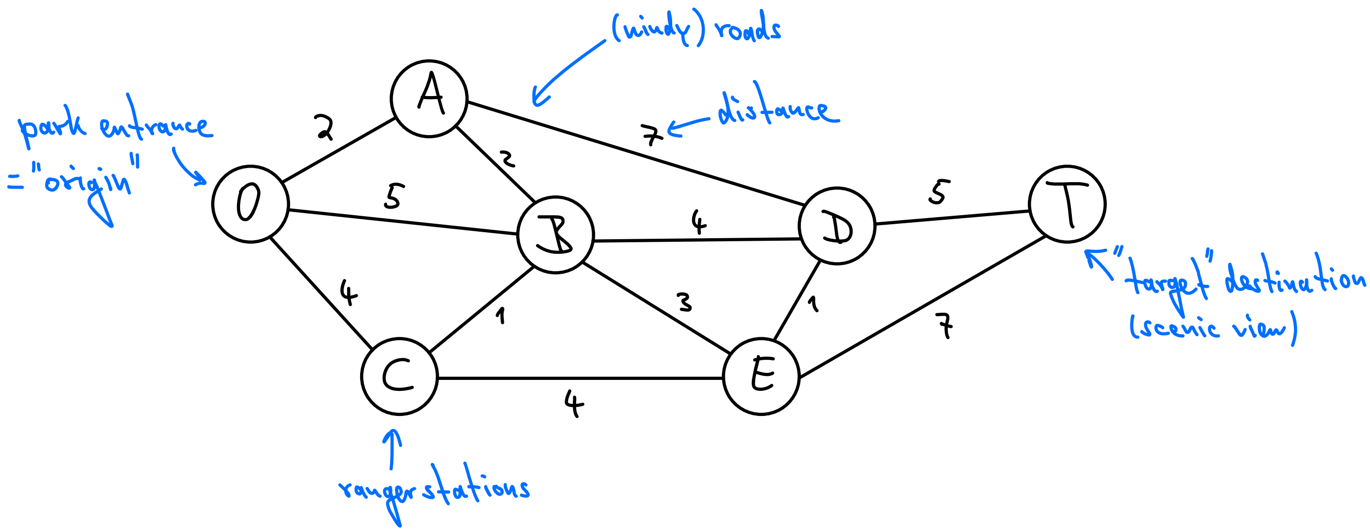
2.6 Network Optimization

Networks are everywhere: transportation, electricity, communication, ...

Network optimization problems are often special types of LP problems (as it was for the transportation problem).

Example to illustrate problem types:

Seervada Park (Hillier, Liebermann Chapter 9)



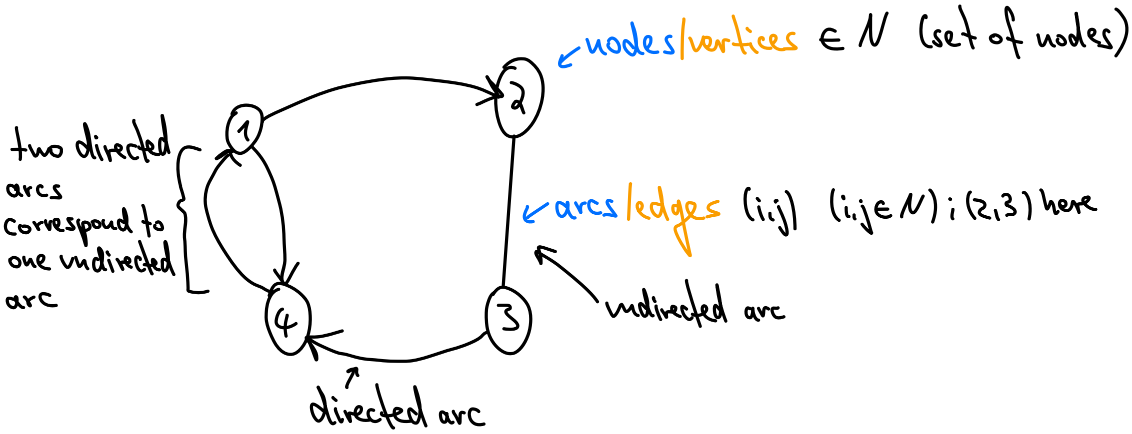
Types of problems:

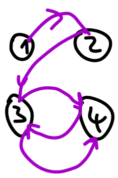
- **Shortest path problem**: Which route from O to T has smallest distance?
- **Minimum spanning tree problem**: Install communication lines under roads so every pair of stations is connected, while minimizing the construction costs.
- **Maximum flow problem**: Limits are set on transportation via each road. Maximize number of trips ("visitor flow") from O to T.

First: some network terminology

OR language ↓   math language ↙

Network / Graph

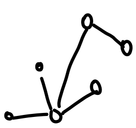
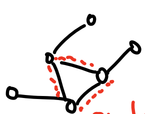


- **Path:** sequence of matching arcs, e.g.  $(1,2),(2,3),(3,4),(4,3),(3,4)$

- **Cycle:** path that begins and ends at same node, e.g., $(1,2),(2,3),(3,4),(4,1)$

- A network is **connected** if there is an undirected path between any two nodes; e.g., Seervada park above is connected,  is not connected.

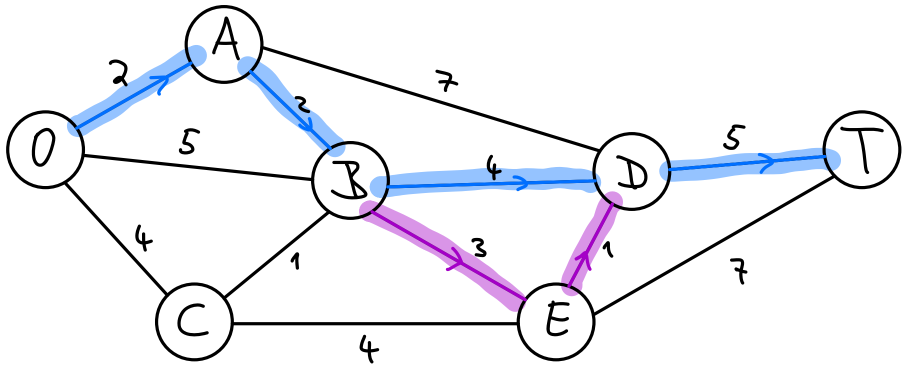
- A **tree** is a connected network that has no cycles.

E.g.:  tree    not a tree

Next, we briefly discuss some special algorithms for the problem types above.

Afterwards, we discuss their LP formulation in a more general context.

- **Shortest Path problem:**



Goal: Find shortest path from O to T.

Algorithm:

- Start with O as a "solved node"; all others are "unsolved nodes".
- List "candidates": unsolved nodes with shortest connecting link to solved nodes.
- Candidate with shortest total distance from O becomes a new solved node.
- Repeat with new set of solved nodes until T is reached.

Table for our example:

Iteration step n (total # of solved nodes)	Solved nodes (directly connected to unsolved nodes)	Closest unsolved node	Total distance	n-th nearest node (= new solved node in next step)	Min. distance	Last connection
1	O	A	2	A	2	O-A
2,3	O A	C B	4 2+2=4	C B	4 4	O-C A-B
4 O already excluded (no connection to unsolved node)	A B C	D E E	2+7=9 4+3=7 4+4=8	E	7	B-E
5	A B E	D D D	2+7=9 4+4=8 7+1=8	D D	8 8	B-D E-D
6	D E	T T	8+5=13 7+7=14	T	13	D-T

⇒ Shortest paths: O-A-B-E-D-T and O-A-B-D-T with total distance 13