Decision aid methodologies in transportation

Lecture 1: Introduction

Prem Kumar

prem.viswanathan@epfl.ch

Transport and Mobility Laboratory

* This course is an extension of the same course taught last year by Dr Niklaus Eggenberg.
A few slides are inspired from the material used by him

Summary

- This session will review the basic concepts in Operations Research
- We will also get introduced to the idea and need for discrete optimization (integer programming)
- An introduction to airline business

Decision-aid optimization

- In the first part of the course, you have learnt about the demand side models that aid in forecasting and evaluating the demand for a product or service
- This part of the course deals with supply side models wherein we will aim to optimize the utilization of supply to maximize revenues
- Context of airlines in this part of the course is only for illustration of the application of the concepts to a particular business

Decision-aid tools

- A decision aid tool aims at facilitating the work of the decision taker
- Its goal is NOT to replace the decision taker
- The decision taker must know how to exploit the decision aid tool in order to be efficient

Decision-aid tools used in this course

- APM suite for aircraft scheduling and crew scheduling. Real dataset from FlyBaboo (now Darwin)
  - Baboo data is confidential and hence the APM suite must be uninstalled after every session
- Optimization with Spreadsheet
- Optimization with MATHPROG

Optimization problems

- All optimization problems are characterized by the presence of the following:
  - An objective function
  - A set of boundary conditions (or constraints)
- Optimization problem are solved using mathematical models
- Models are representation of the actual scenario or problem and having three important features:
  - Formulation (mathematical or otherwise, of the objective and constraints)
  - Parameters (known in advance)
  - Decision variables (to be determined)

Optimization problems

- Optimization problems can have
  - Linear objective function and linear constraints
  - Non-linear objective function and linear constraints
  - Linear objective function and non-linear constraints
  - Non-linear objective function and non-linear constraints
- Why is the categorization between linearity and non-linearity?
- We will consider linear objective function and constraints in most of this course
- Such problems are solved by linear programming techniques

An Example of Linear Program

- A shipping company plans to acquire an aircraft and is designing a customized interior to carry thermally insulated and normal products
- Temperature controlled products are sold in the market for a profit of CHF 7 per unit, while normal ones yield a profits of CHF 5 per unit
- Temperature controlled products require 2 KW-h electric power and 3 cu m space for carrying one unit
- Normal products requires 1 KW-h power and 4 cu m space per unit
- Total power and space availability are 1000 KW-h and 2400 cu m
- Assuming that aircraft will always fly full capacity, how many units of temp controlled and normal products should it be designed for?

An Example of Linear Program

Max $7X_1 + 5X_2$ (Profit)

subject to

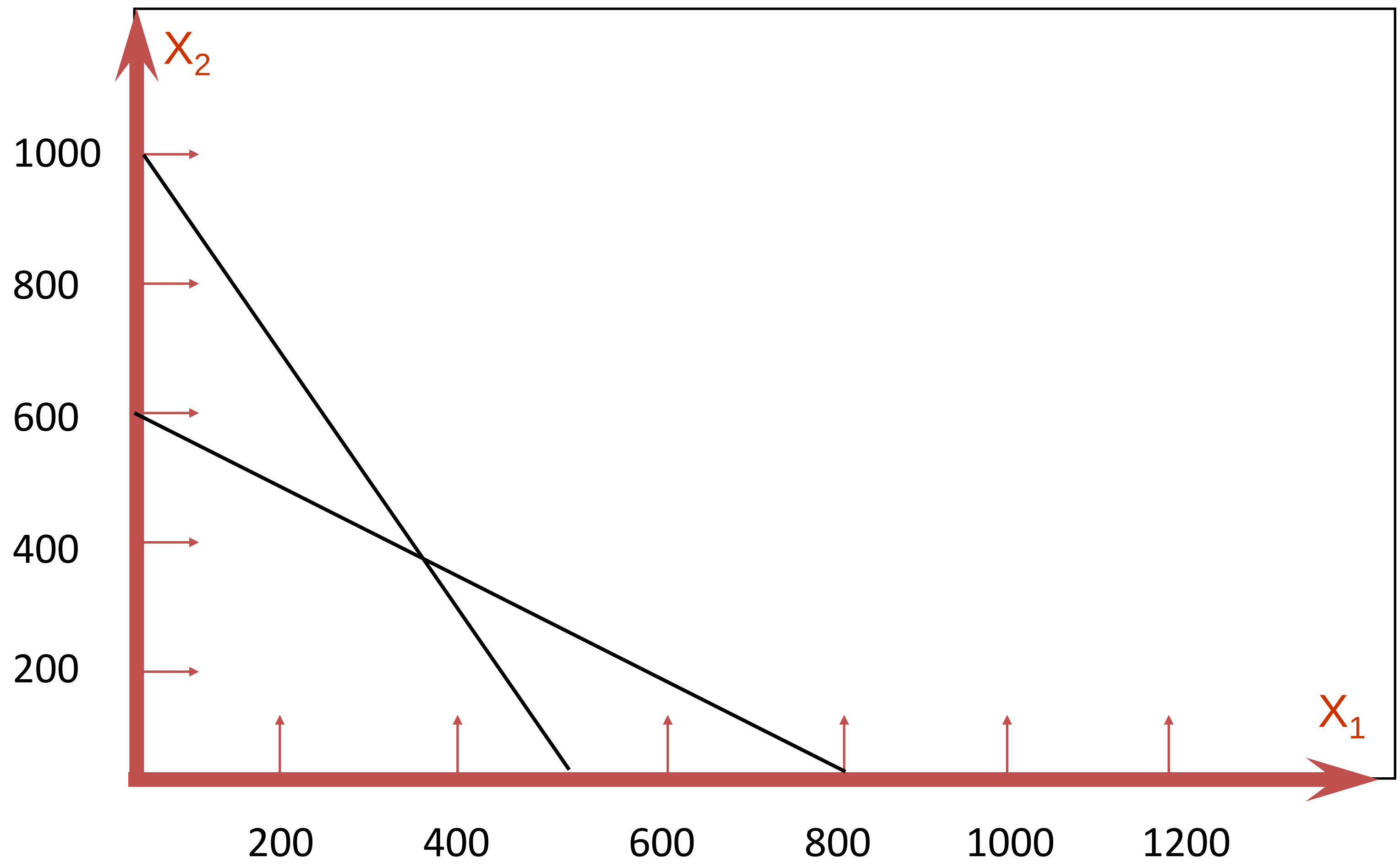
$2X_1 + 1X_2 \leq 1000$ (Power)

$3X_1 + 4X_2 \leq 2400$ (Space)

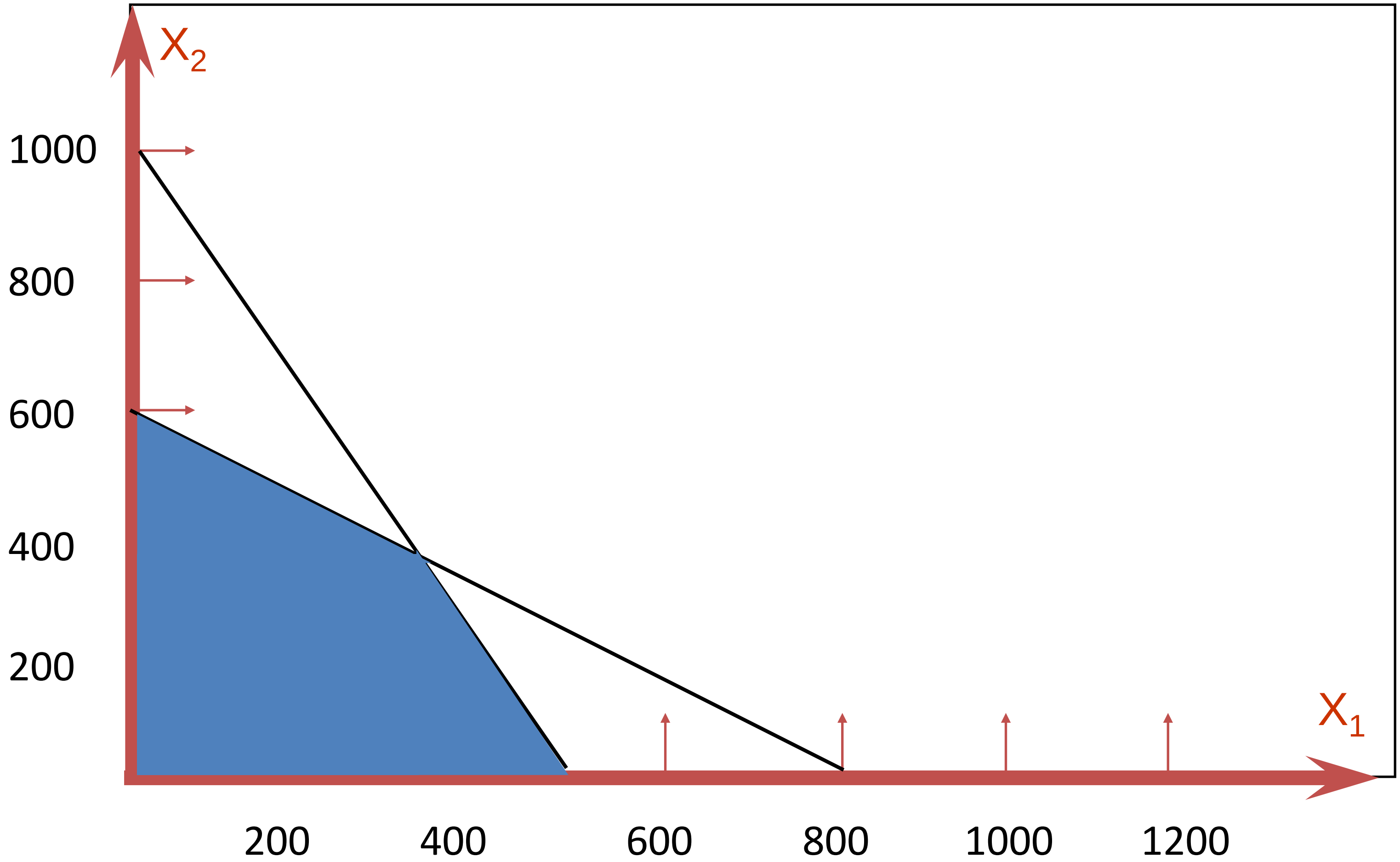
$X_j \geq 0, \ j = 1,2$ (Non-negativity)

* Note that this can be represented as a vector of variables, constraint matrix and RHS vector

An Example of Linear Program: Graph Analysis



An Example of Linear Program: Graph Analysis



Linear Program: Solution

- Corner point theorem: Optimal solution for a linear program always lies on one of the corner points
- Simplex algorithm visits corner points in sequence
- Optimal solution for the example problem has to be one of (0,0), (500,0), (0,600) or (320,360)
- Optimal design for temperature controlled products is 320 units and non-temperature controlled ones is 360 units. Total optimal profits is CHF (7x320 + 5x360) = CHF 4040
- Interior point algorithm is another method to solve LP

Dual of a Linear Program

Max $7X_1 + 5X_2$ (Profit)

subject to

$2X_1 + 1X_2 \leq 1000$ (Power) π_1

$3X_1 + 4X_2 \leq 2400$ (Space) π_2

$X_j \geq 0, \ j = 1,2$ (Non-negativity)

Min $1000\pi_1 + 2400\pi_2$ (Cost)

subject to

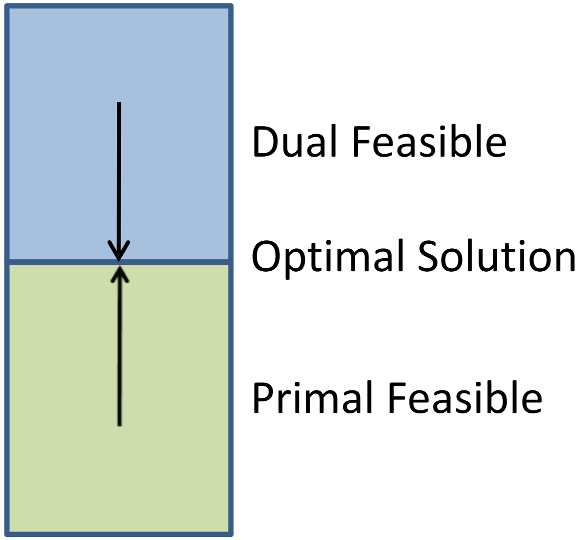
$2\pi_1 + 3\pi_2 \geq 7$ (Temp controlled) X_1

$\pi_1 + 4\pi_2 \geq 5$ (Normal) X_2

$\pi_i \geq 0, \ i = 1,2$ (Non-negativity)

Linear Program: Duality

- Note that the optimal solution value of the primal as well as dual is the same at CHF 4040
- Even though the feasible regions of the primal and dual are exclusive, they tend to meet at optimality



Linear Program: Duality Theory

- Note that the optimal solution value of the dual variables are CHF 2.60 and CHF 0.60 respectively
- These variables are referred to as the shadow prices of the constraints in the primal
- Thus one KW-h increase in the availability of power would result in an incremental profit of CHF 2.60
- Example, if the constraint $2X_1 + 1X_2 \leq 1000$ is updated as $2X_1 + 1X_2 \leq 1200$, the new profit would be CHF 4040 + CHF 2.60 x (1200-1000) = CHF 4560
- Obviously shadow prices are valid only for a specific range

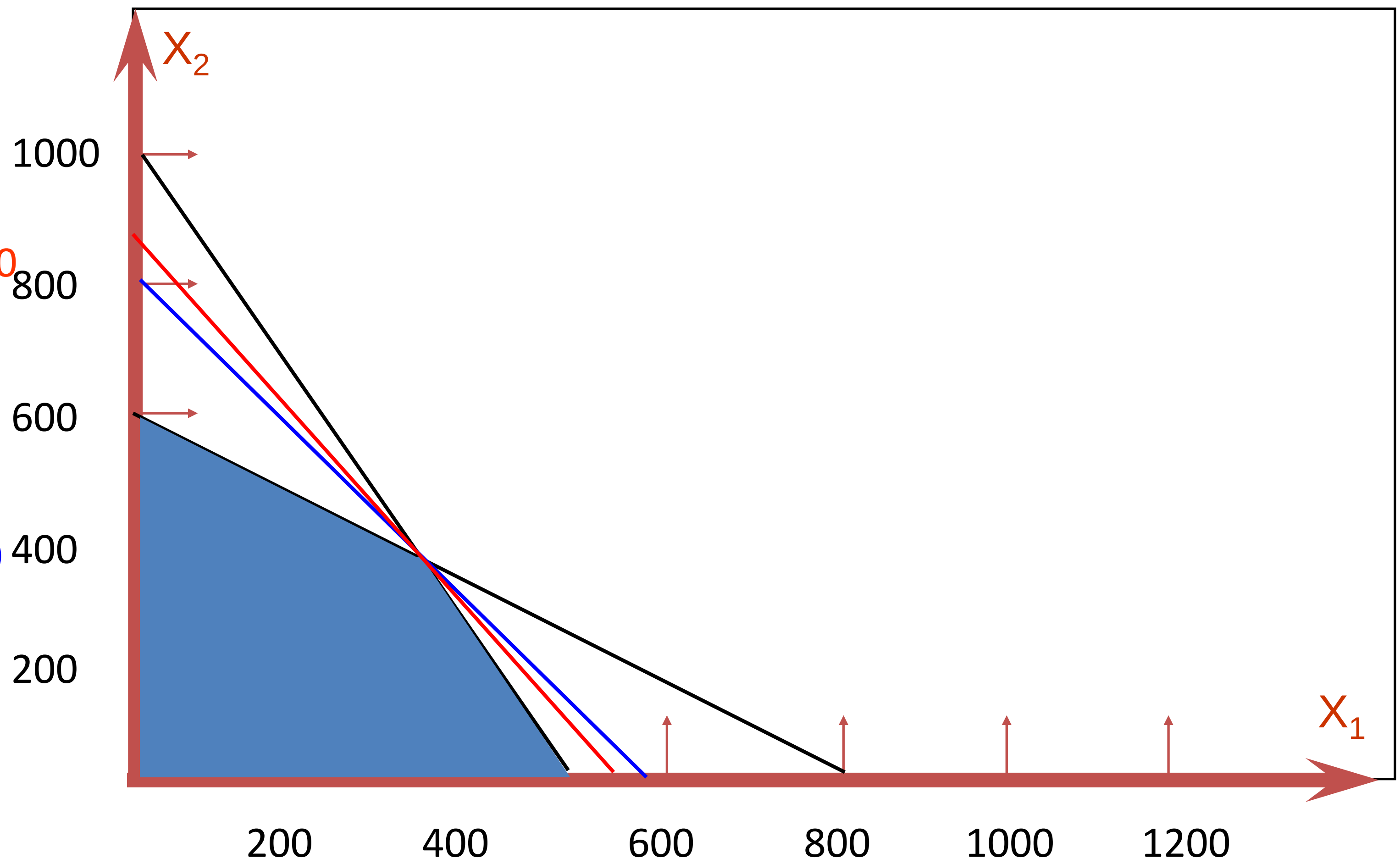
Linear Program: Sensitivity Analysis

- What happens if the profit contribution of temp controlled product changes from CHF 7 to CHF 8 per unit?
- Max $\overset{8}{7}\,X_1 + 5\,X_2$ (profit contribution)
- Total profit certainly goes up
- But would this change also mean if the amount of temp controlled product should be carried more?
- Will it change the optimal solution?

Linear Program: Sensitivity Analysis

Original
Objective Function
$7\,X_1 + 5\,X_2 = \text{CHF}4040$

Updated
Objective Function
$8\,X_1 + 5\,X_2 = \text{CHF}4360$



Linear Program: Sensitivity Analysis

If the OFC of variable X_1 became higher or lower beyond a certain range

$11 X_1 + 5 X_2 = \$5500$

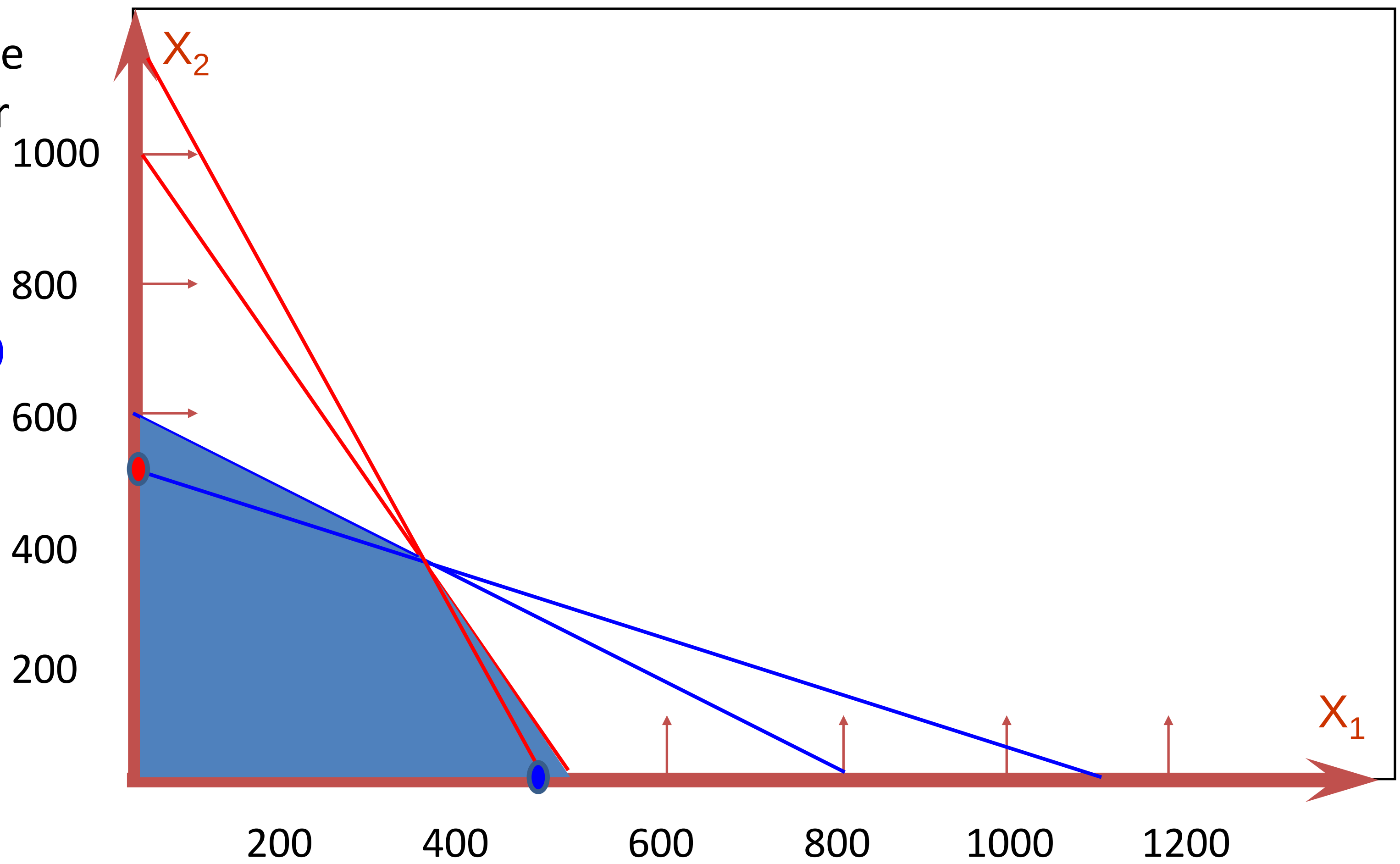
Optimal Solution

(X_1 =500, X_2 =0)

$3 X_1 + 5 X_2 = \$2850$

Optimal Solution

(X_1 =200, X_2 =450)



Linear Program: Reduced Costs

- Note that when the coefficient of profit for temp controlled is increased from 7 to 11, while making no change in other coefficients, the optimal solution for normal products is 0
- Reduced cost is defined as the minimum amount by which the objective function coefficient (OFC) of a variable should change to cause that variable to become non-zero (0 to 1 in case of integrality)
- Note that the reduced cost of the variable vector is computed using

$$c - y^T A$$

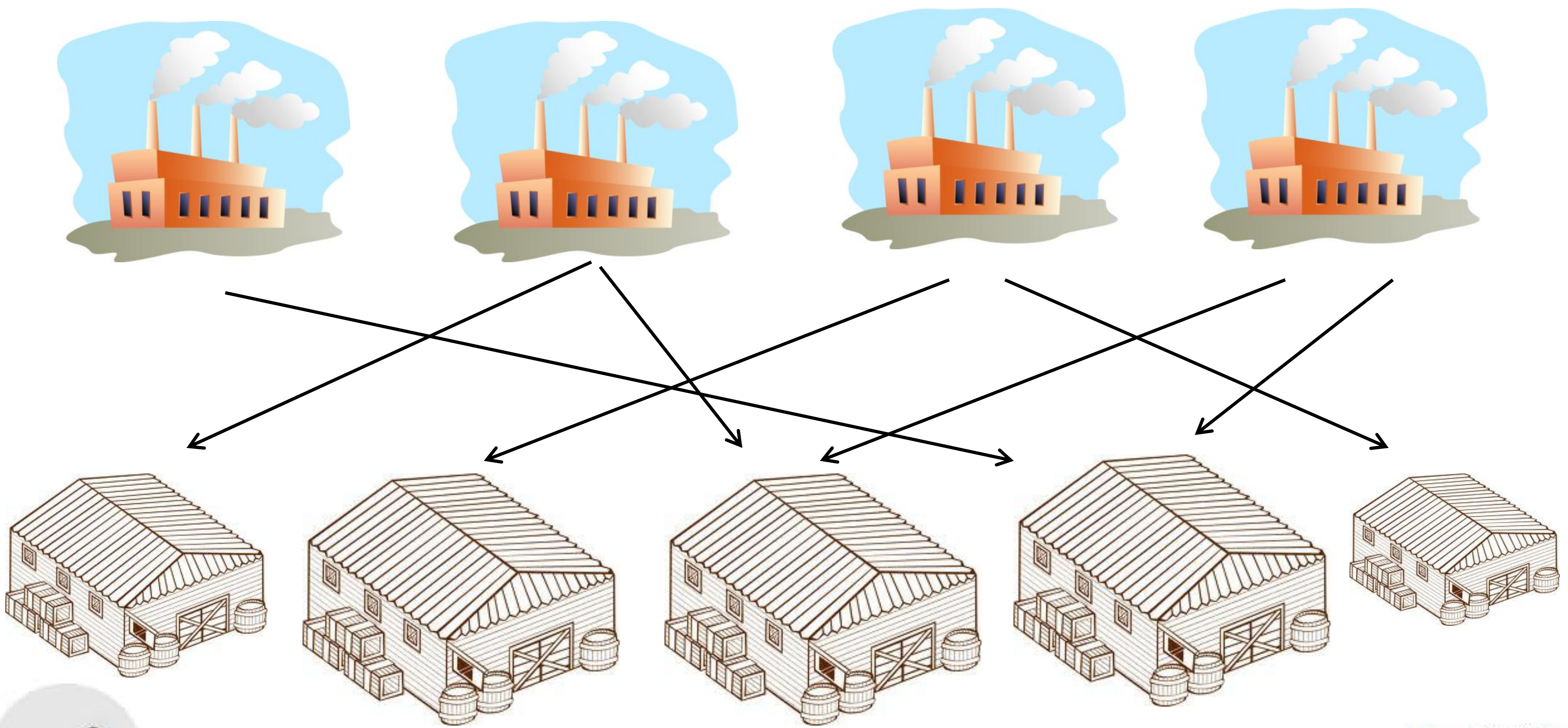
where c^T is the OFC vector, y is the dual optimal solution and A is the constraint matrix

Integer Program: Introduction

- So that brings us to Integer Programming
- Why do we need integer programming if all problems in the world (subject to linearity) can be solved with linear programs?
- Assume a situation where we need to determine the number of cars transported from factory location F_m ($m = 1, 2, \ldots, M$) to retail outlet R_n ($n = 1, 2, \ldots, N$).
- After solving the linear program, we end up with the optimal solution involving transporting 15.37 cars from F_4 to R_{11}
- Obviously the number of transported cars cannot be anything but integral

Integer Program: Examples

- Transportation problem



Transportation Problem: Math Formulation

- Let us consider a factory-warehouse cost minimization formulation
- Sets
  - i is the set of factories where $i = \{1, 2, \ldots, I\}$
  - j is the set of warehouses where $j = \{1, 2, \ldots, J\}$
- Parameters
  - c_{ij} be the cost of transporting one unit from factory i to warehouse j
  - s_i be the total production at factory i (supply constraint)
  - d_j be the total demand at warehouse j (demand constraint)
- Decision variables
  - x_{ij} = number of units transported from factory i to warehouse j

Assignment Problem: Mathematical Formulation

- Objective Function (minimize costs)

$$\text{Minimize} \sum_i \sum_j c_{ij} x_{ij}$$

- Total supply cannot exceed the factory production capacity and has to be more than the demand at each warehouse

$$\sum_i x_{ij} \geq d_j, \ \forall j$$

$$\sum_j x_{ij} \leq s_i, \ \forall i$$

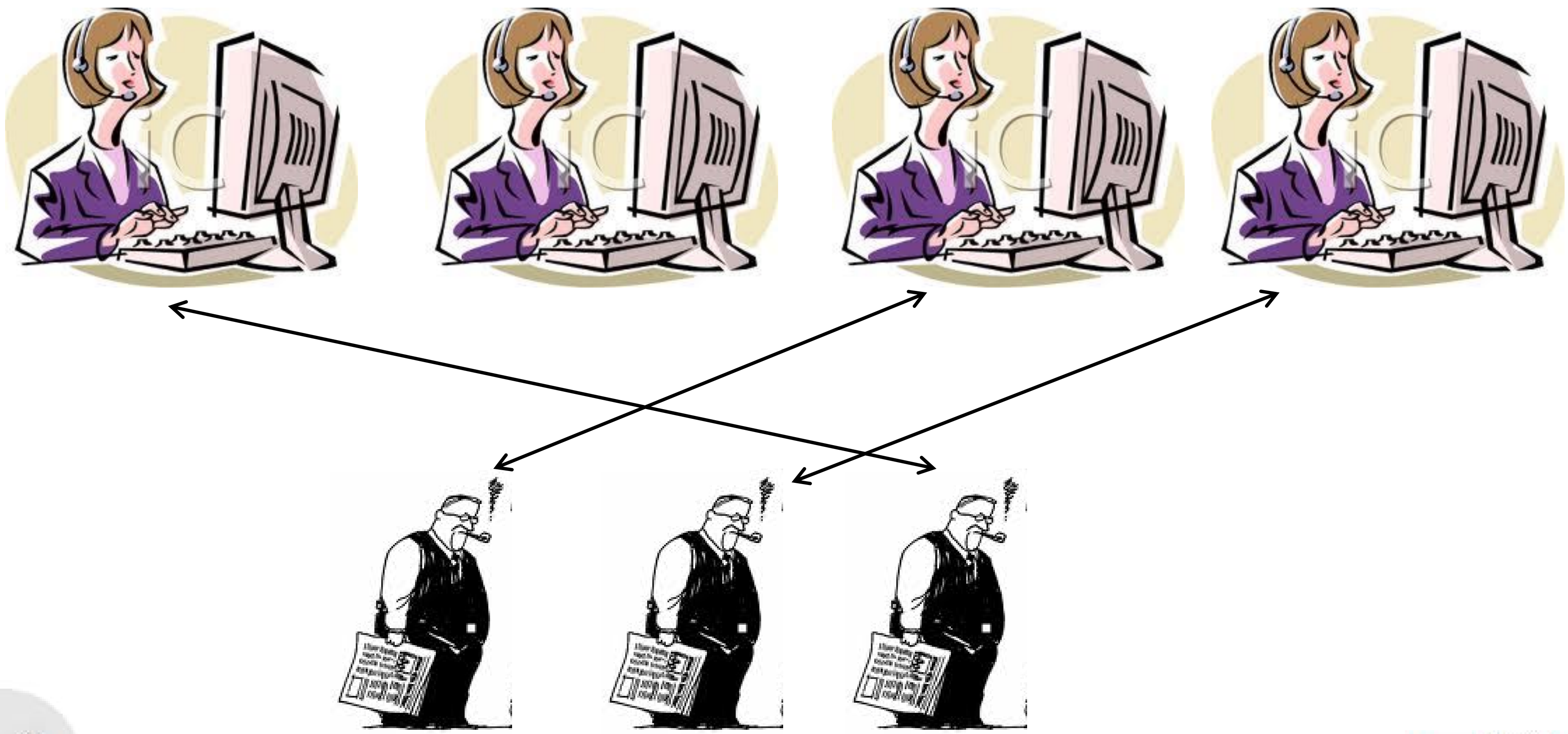
- Bounds

$$x_{ij} \geq 0$$

$$x_{ij} \in Integer$$

Integer Program: Examples

- Assignment problems



Assignment Problem: More Examples

- Man-Machine assignment (one-to-one)
- Job-Machine assignment (one-to-one, one-to-many, many-to-one)
- Factory-Retailer assignment (one-many, many-one, many-many)
- Train-Platform or Flight-Slots (one-to-one, many-to-one)
- Demand-Supply
- Teacher-Class
- Course-Room
- Man-Woman
- ...

Assignment Problem: Mathematical Formulation

- Let us consider a man-machine assignment formulation
- Sets
  - i is the set of men where $i = \{1, 2, \ldots, I\}$
  - j is the set of machines where $j = \{1, 2, \ldots, J\}$, where $J \leq I$
- Parameters
  - p_{ij} represents the productivity when man i works on machine j
- Decision variables
  - $x_{ij} = 1$ when man i is assigned to machine j, 0 otherwise

Assignment Problem: Mathematical Formulation

- Objective Function (maximize productivity)

$$\text{Maximize} \sum_i \sum_j p_{ij} x_{ij}$$

- Every machine gets exactly one man assigned to it, not all men are assigned to some machine, though

$$\sum_i x_{ij} = 1, \ \forall j$$

$$\sum_j x_{ij} \leq 1, \ \forall i$$

- Bounds

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

Integer Program: Examples

- Knapsack Problem: Problem originated in the context of mountaineers who need to pack necessary items for their expedition, but have a finite weight limitation

Knapsack Problem: Math Formulation

- Sets
  - i is the set of items that are contenders for space in the knapsack where i ={1, 2, …, I}
- Parameters
  - v_i be the perceived value of item i
  - w_i be the weight of item i
  - W be the maximum allowable weight in the knapsack
- Decision variables
  - x_i = 1 if item i makes it to the knapsack, 0 otherwise

Knapsack Problem: Mathematical Formulation

- Objective Function (maximize value)

$$\text{Maximize} \sum_i v_i x_i$$

- The total weight of items packed into the knapsack cannot exceed its capacity

$$\sum_i w_i x_i \le W$$

- Bounds

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

Knapsack Problem: Algorithm

- How do we solve the knapsack problem?
- Let us say that all the item weights are the same and one unit each
- $\lfloor W \rfloor$ would represent the number of items that will go into the knapsack
- Value will be maximized if all items are sorted by value (highest to lowest) and we pick the first $\lfloor W \rfloor$ items
- Do you agree?
- This algorithm is referred to as **GREEDY ALGORITHM** because it tends to optimize that way

Knapsack Problem: Algorithm

- Is the greedy algorithm optimal?
- How to use the greedy algorithm if the weights are not identical?
- Sort the items by $\left(\frac{v_i}{w_i}\right)$ - largest to smallest
- Pick the items from the sorted list in a greedy manner till the weight limit of the knapsack is not exceeded
- Is the greedy algorithm in this example optimal?

Knapsack Problem: Example

- Let the value and weight of the items be as given below:

Value (v)	6	7	2	12	3	3	14	2
Weight (w)	3	8	1	5	4	2	6	2
v/w	2	0.875	2	2.4	0.75	1.5	2.333333	1

- How to apply greedy algorithm here? Will it be optimal?

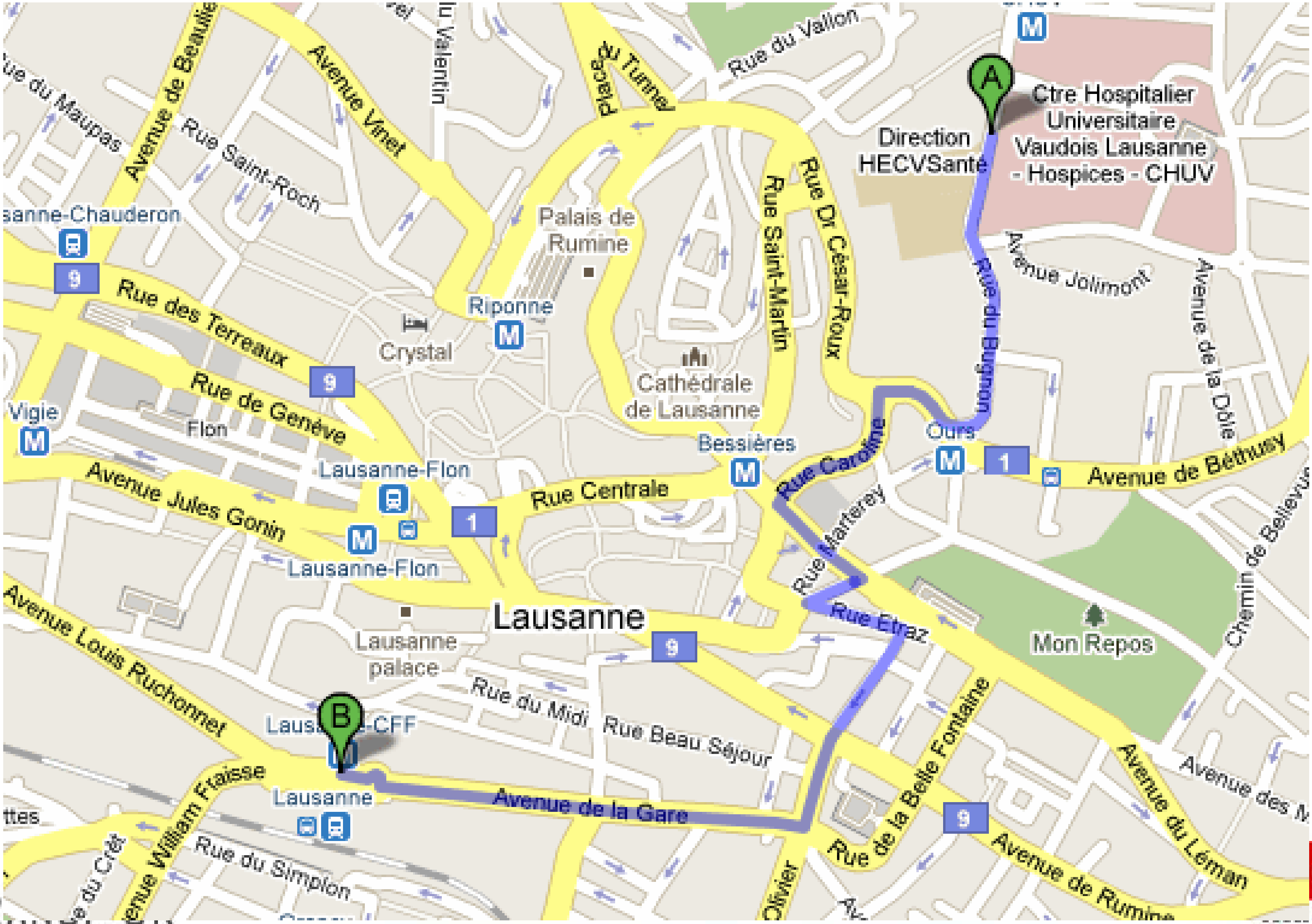
Greedy Algorithm

- A greedy algorithm picks up a choice that appears to be the most beneficial at any decision making stage
- Typical examples:
  - Driving from Lausanne to Morges (motorway or route cantonale?)
  - Investment in high risk instruments
  - Choice of subject and university
  - Playing chess, cards etc.
- For some problem, such as continuous knapsack, it works
- Greedy algorithms are easier to implement and test

Greedy Algorithm

- Don’t worry though. Greedy algorithm is not a *deadwood* if it cannot be applied to integral knapsack problems
- Greedy algorithm gives optimal (sometimes, sub-optimal but good) solutions to different integer program problems such as:
  - Minimal (maximal) spanning tree
  - Bin packing
  - Graph coloring
- Home work: Write the mathematical formulation of the bin packing problem

Integer Program: More Examples



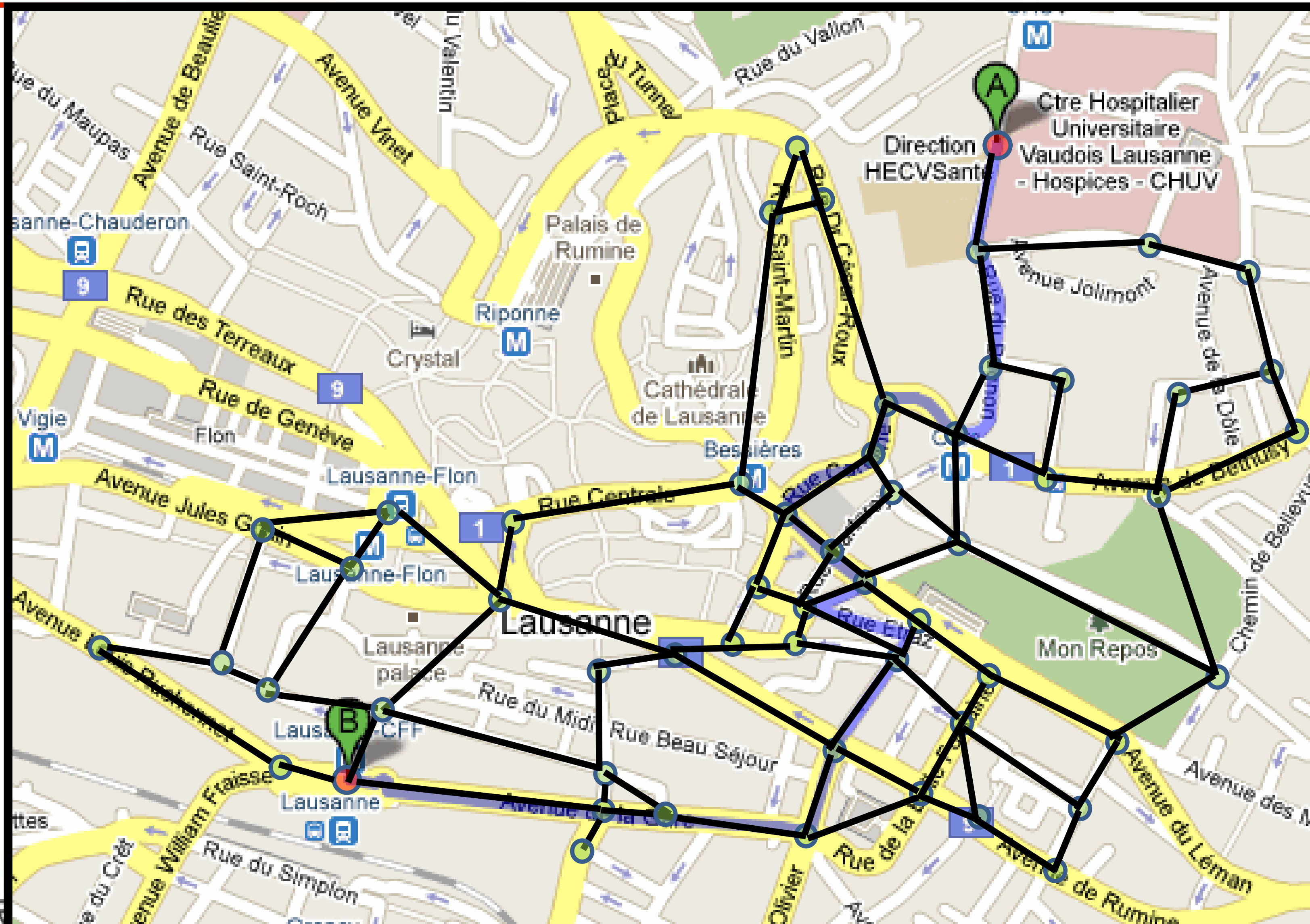
Integer Program: Shortest Path Problem

- What is the shortest path from CHUV (A) to Lausanne Gare (B)?
- So far, we had looked at mathematical formulations
- Now let us change track and look at network formulation for this integer program problem. Consider the following representation:
  - 1 node for the origin (A)
  - 1 node for the destination (B)
  - 1 node for each crossing
  - 1 arc between each crossing nodes if they are directly connected with a road segment

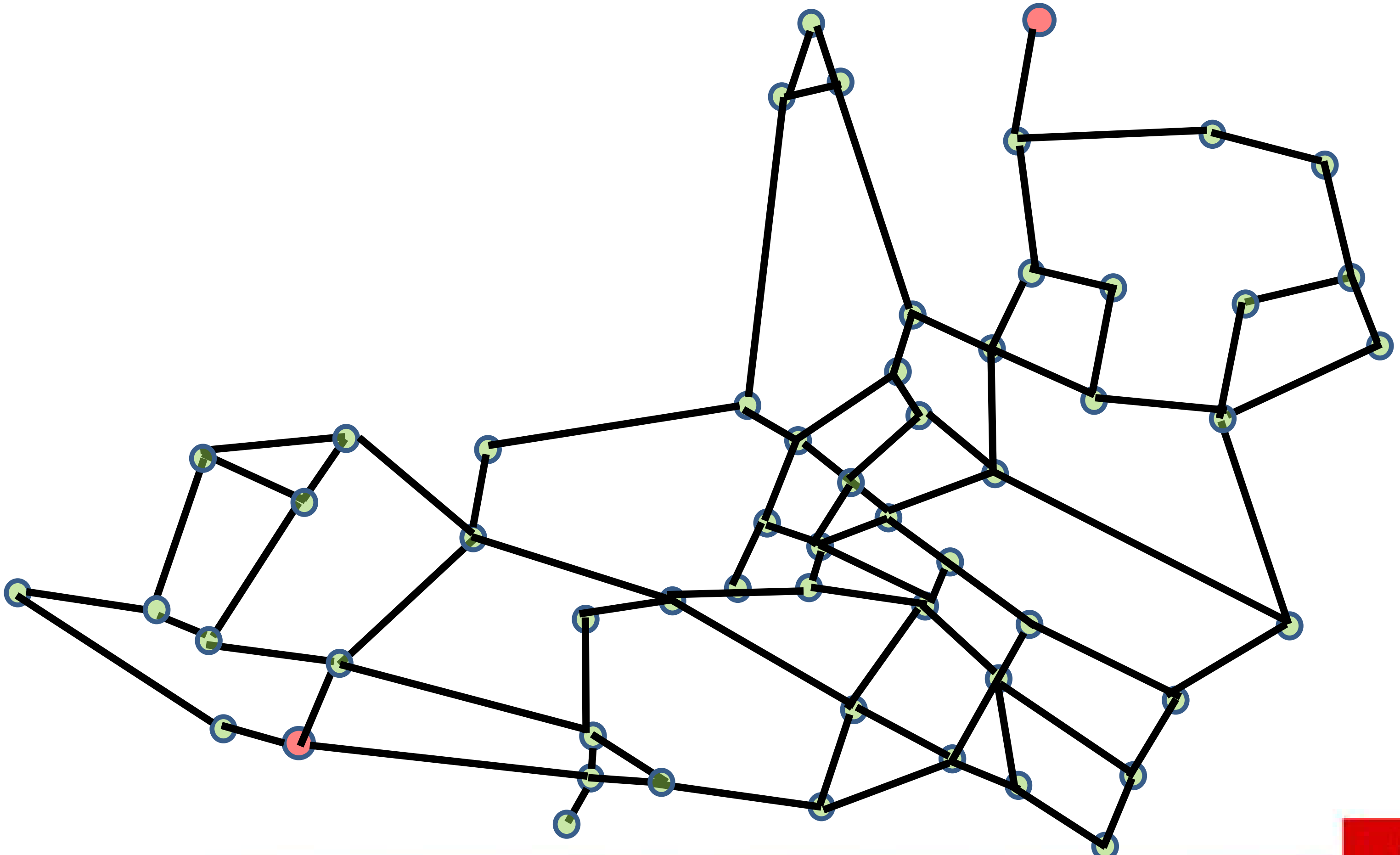
Integer Program: Shortest Path Problem

- Objective function
  - Quickest path / Shortest Path / Minimum cost
- Parameters
  - Road network to build the graph
  - Deterministic travel times / costs
- Decision variable
  - Which arcs should be used and which ones not?

Shortest Path: Graphical Representation



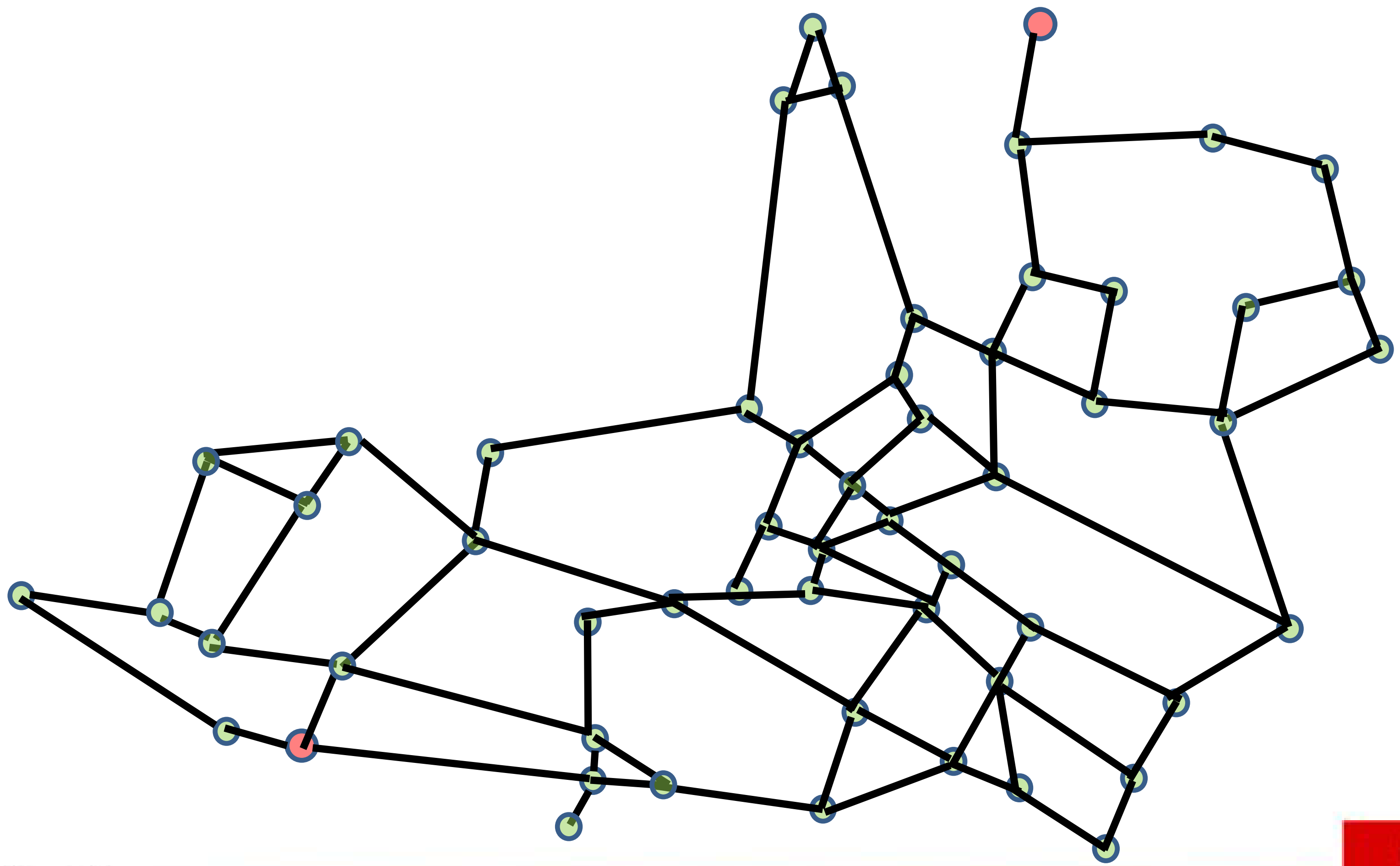
Shortest Path: How to Solve?



Shortest Path Problem: Dijkstra's Algorithm

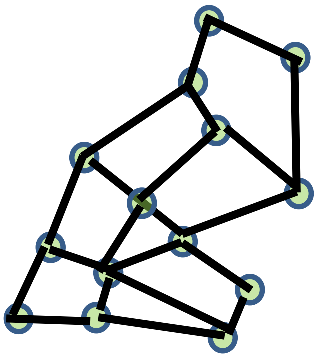
- Let the starting node (CHUV here) be called initial node
- Assign to every node a distance value: set zero for initial node and infinity for all other nodes
- Mark all nodes as unvisited. Set initial node as current.
- For current node, consider all its unvisited neighbors and calculate their *tentative* distance. If this distance is less than the previously recorded distance, overwrite the distance.
- When we are done considering all neighbors of the current node, mark it as visited. A visited node will not be checked ever again; its distance recorded now is final and minimal.
- If all nodes have been visited, finish. Otherwise, set the unvisited node with the smallest distance (from the initial node, considering all nodes in graph) as the next "current node" and continue from step 3.

Dijkstra’s Algorithm: Illustration

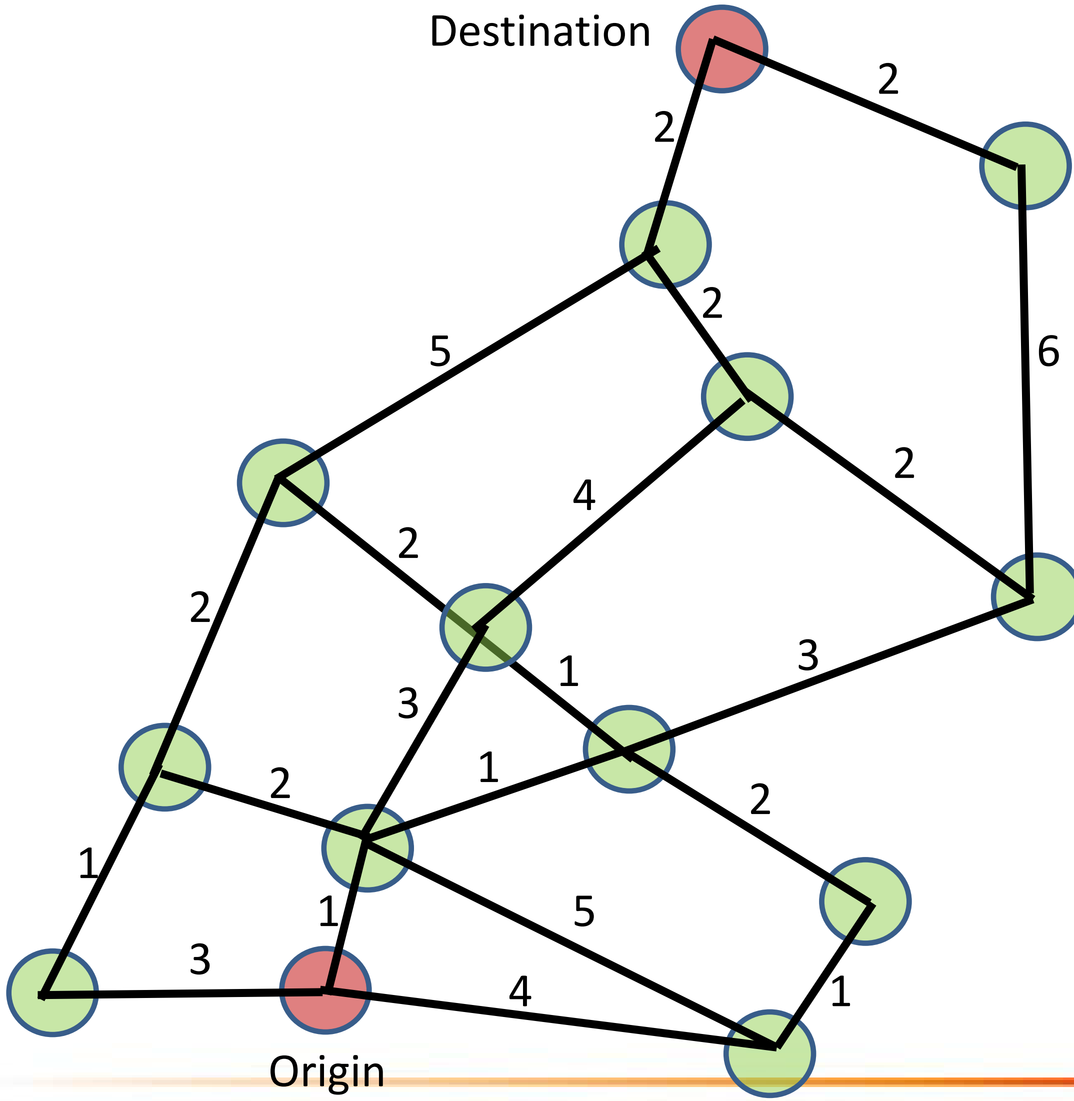


Dijkstra's Algorithm: Illustration

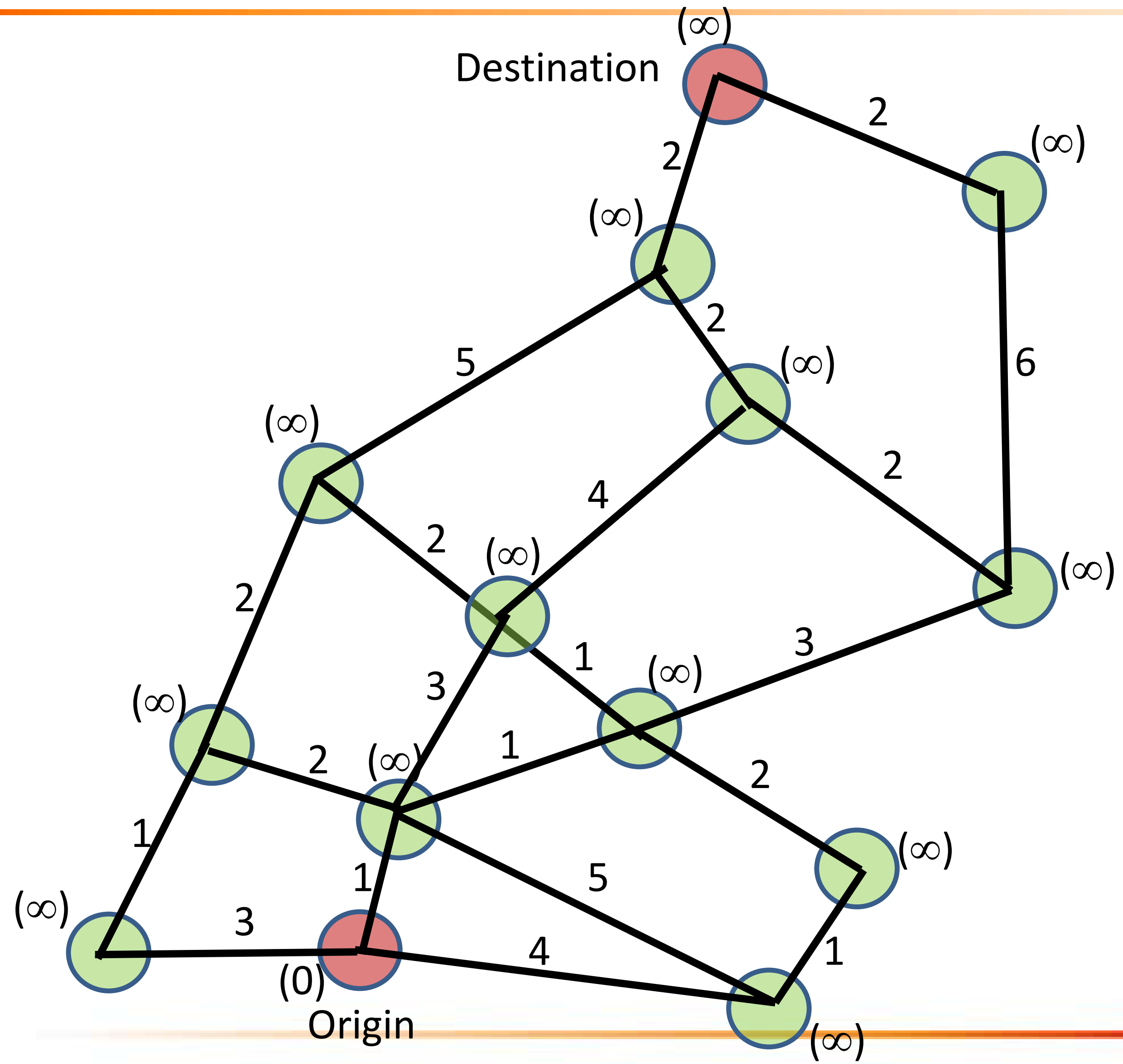
- Let us illustrate on a small network instead



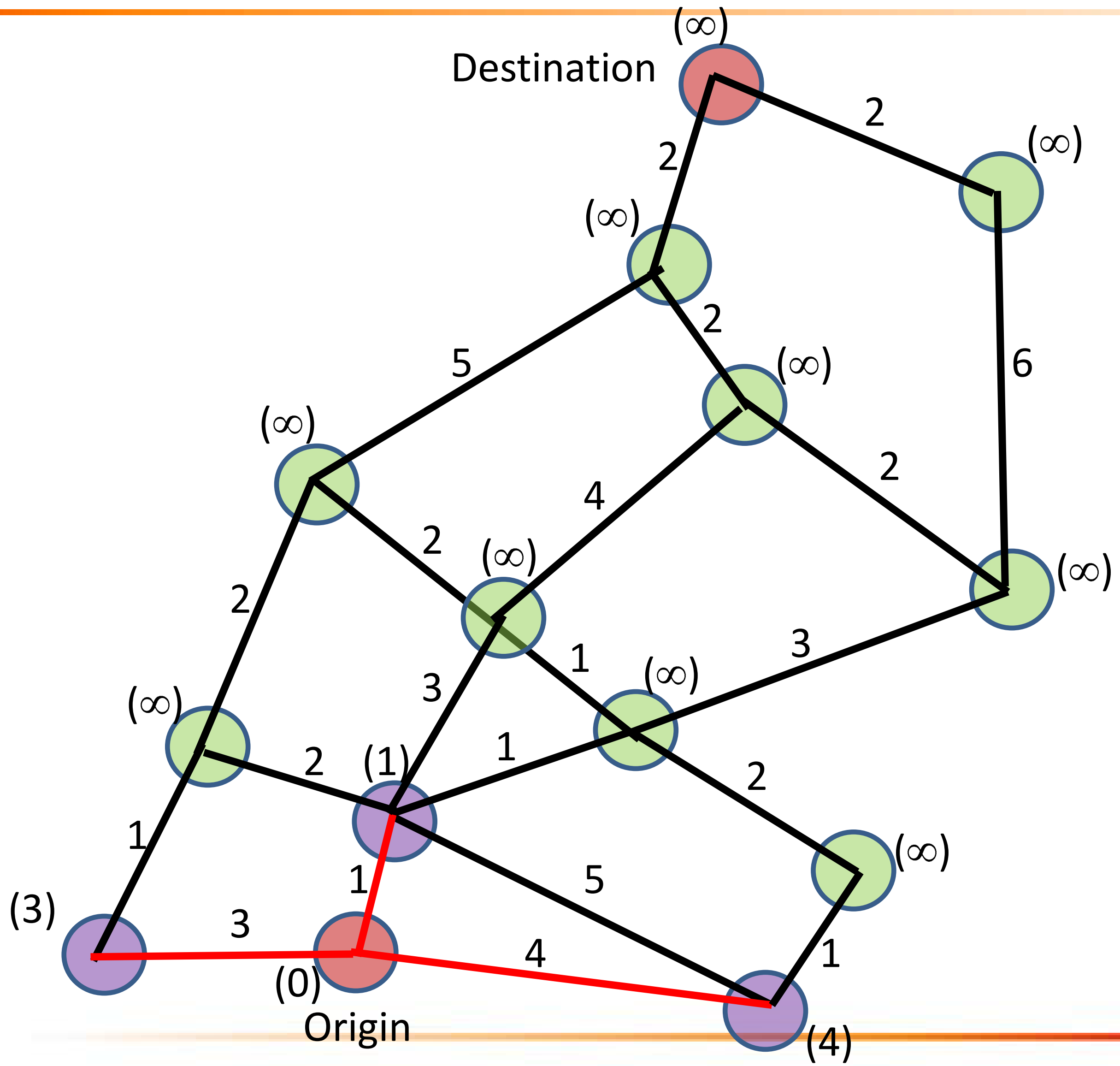
Dijkstra's Algorithm: Illustration



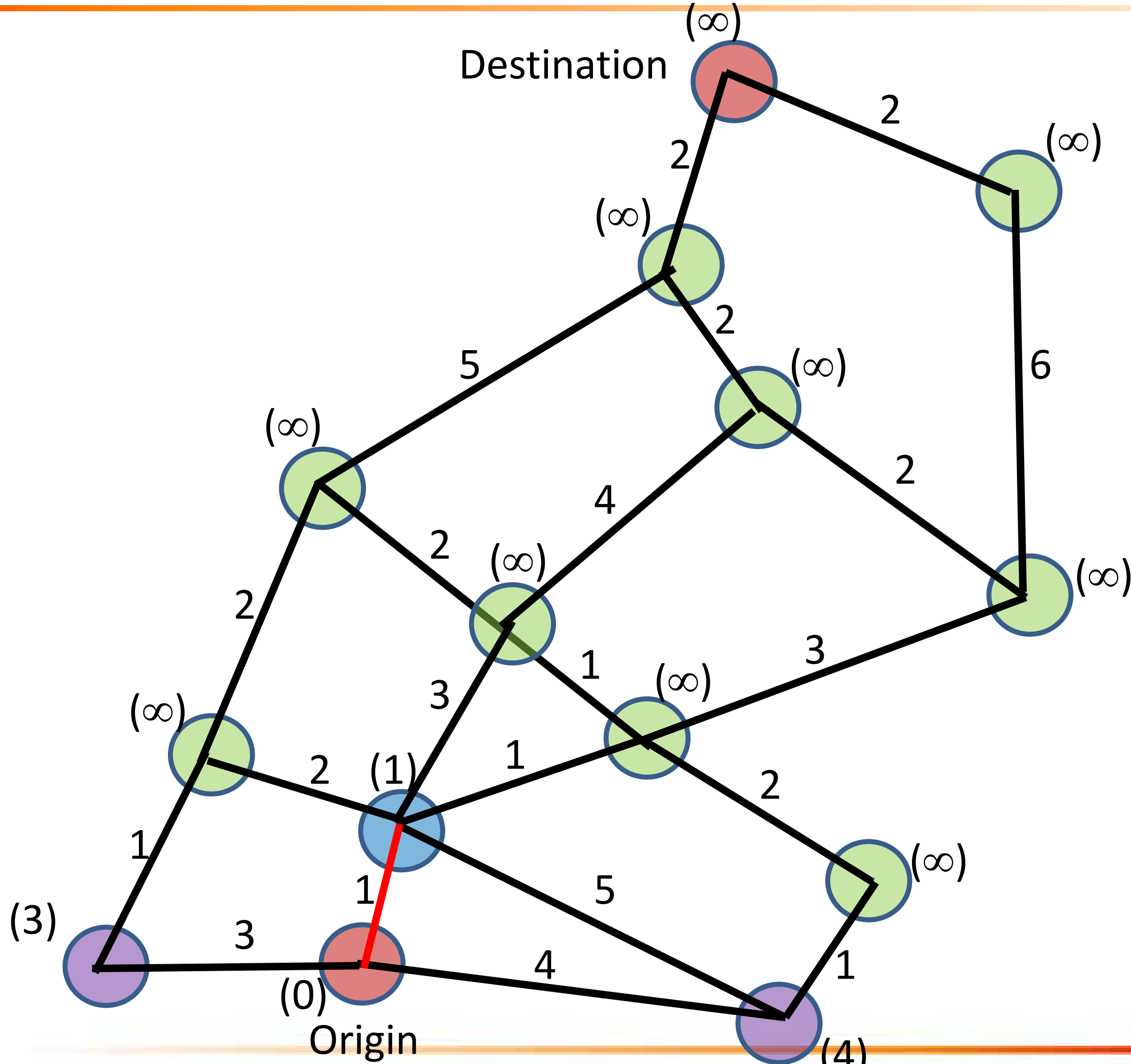
Dijkstra’s Algorithm: Illustration



Dijkstra’s Algorithm: Illustration



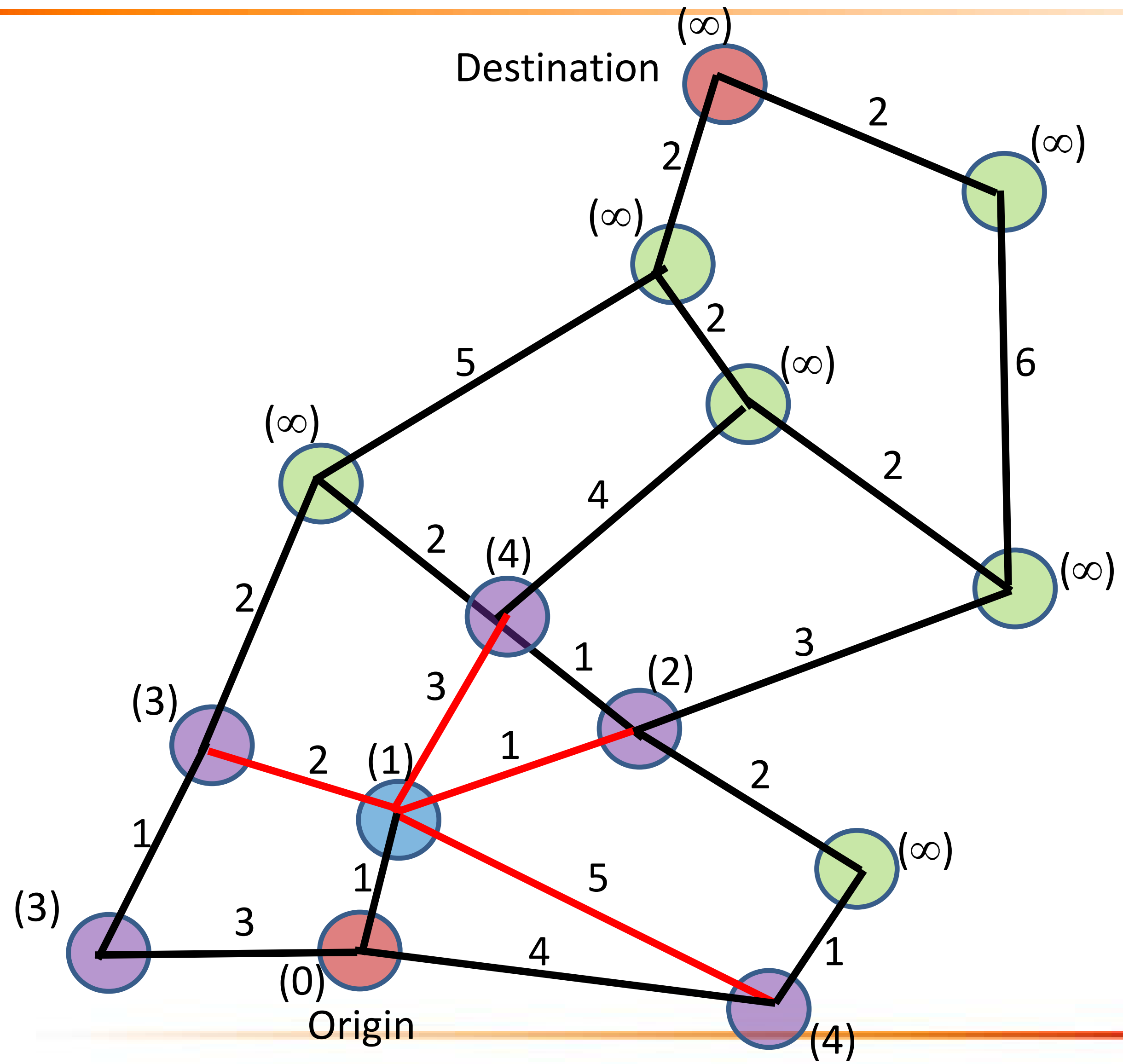
Dijkstra's Algorithm: Illustration



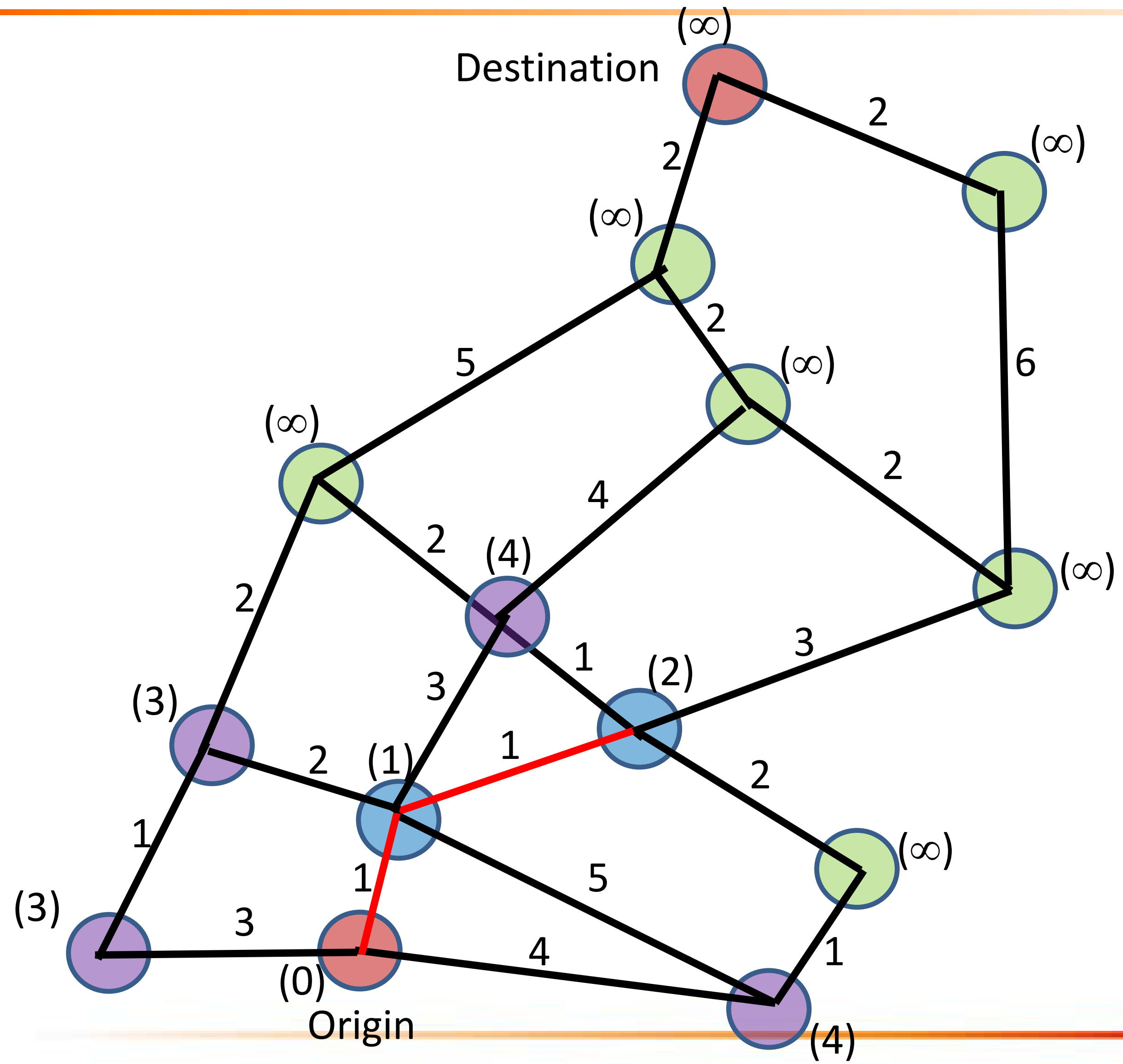
TRANSP-OR

EPFL
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Dijkstra’s Algorithm: Illustration



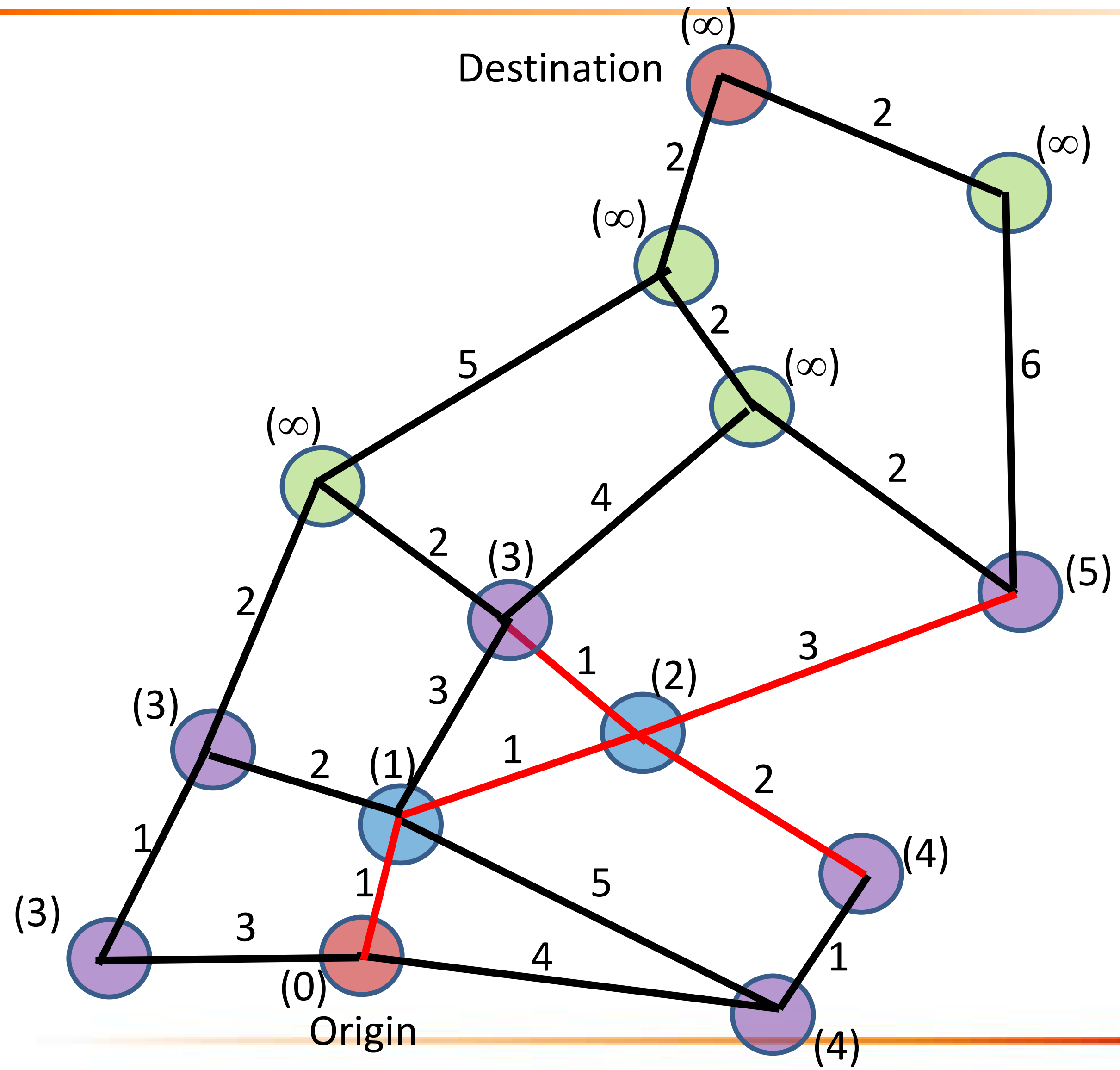
Dijkstra’s Algorithm: Illustration



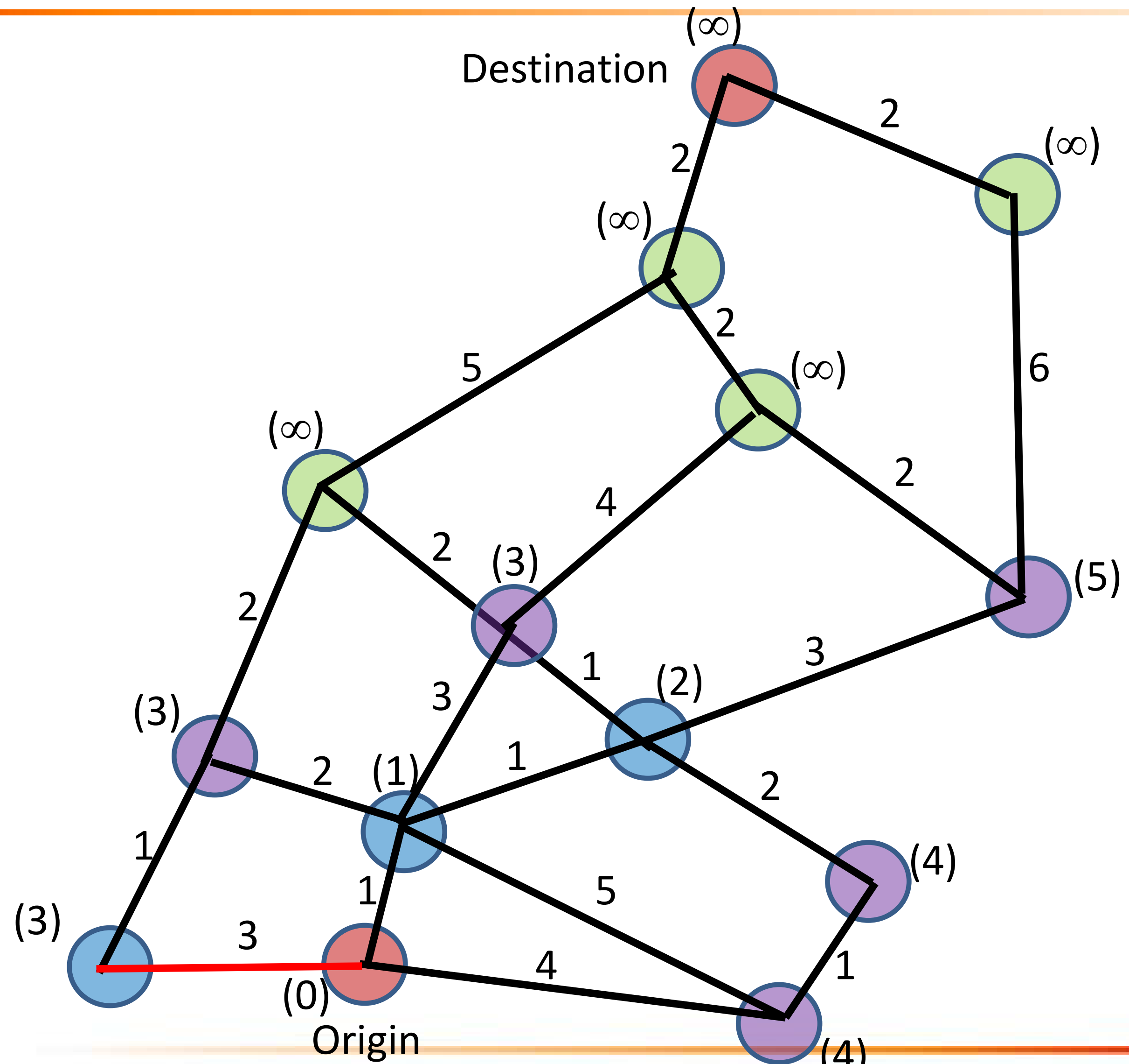
TRANSP-OR

EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

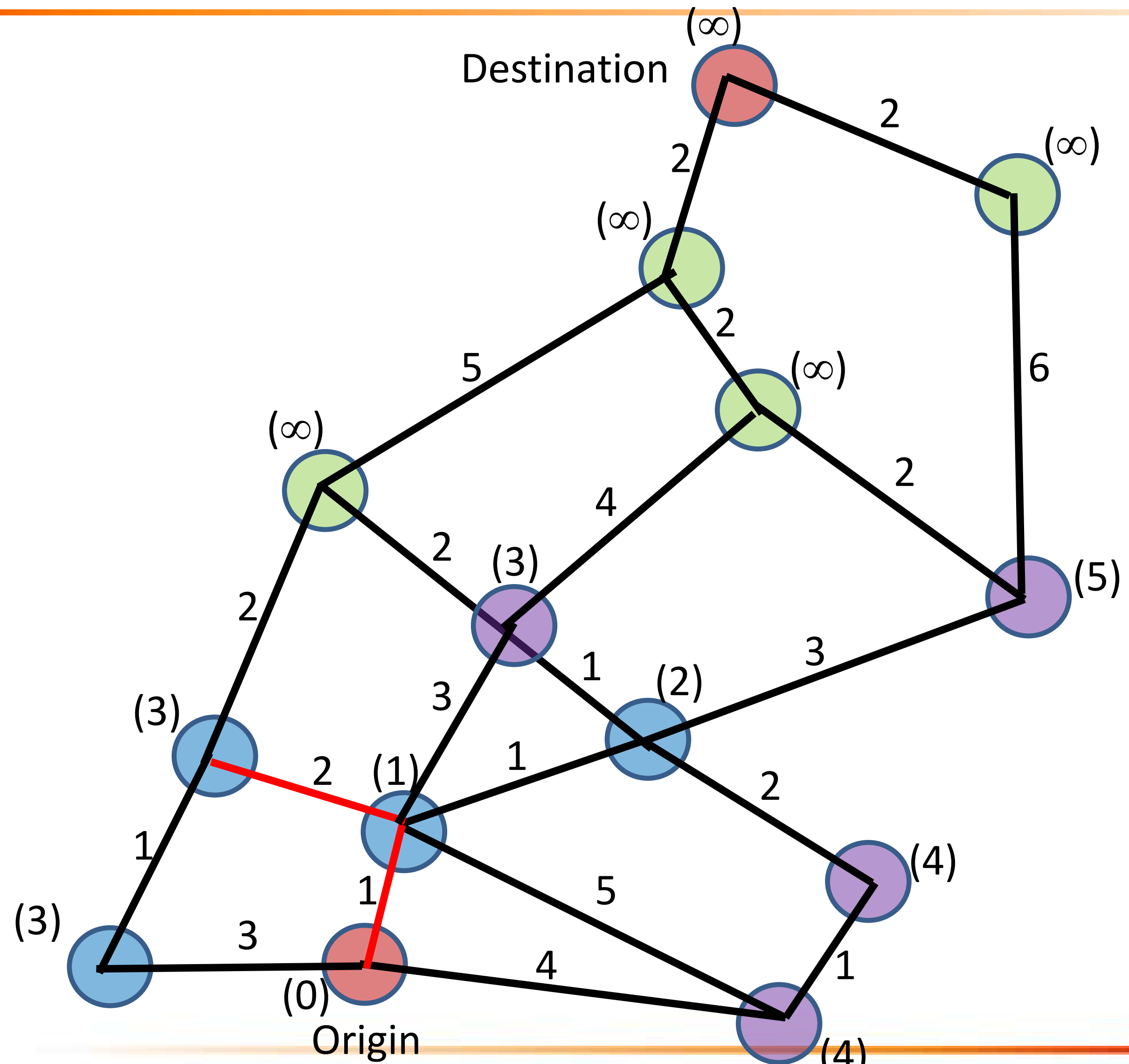
Dijkstra’s Algorithm: Illustration



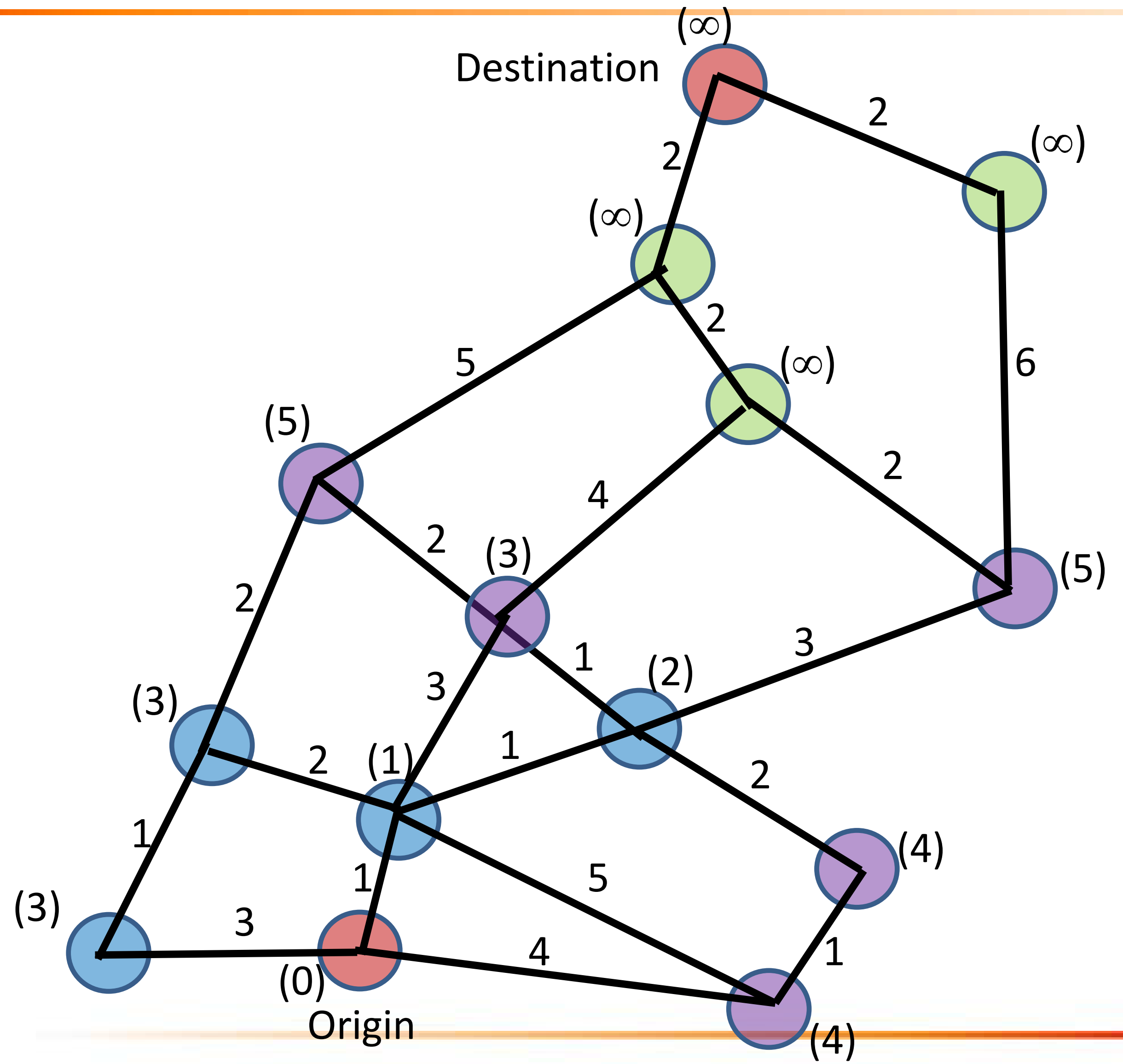
Dijkstra’s Algorithm: Illustration



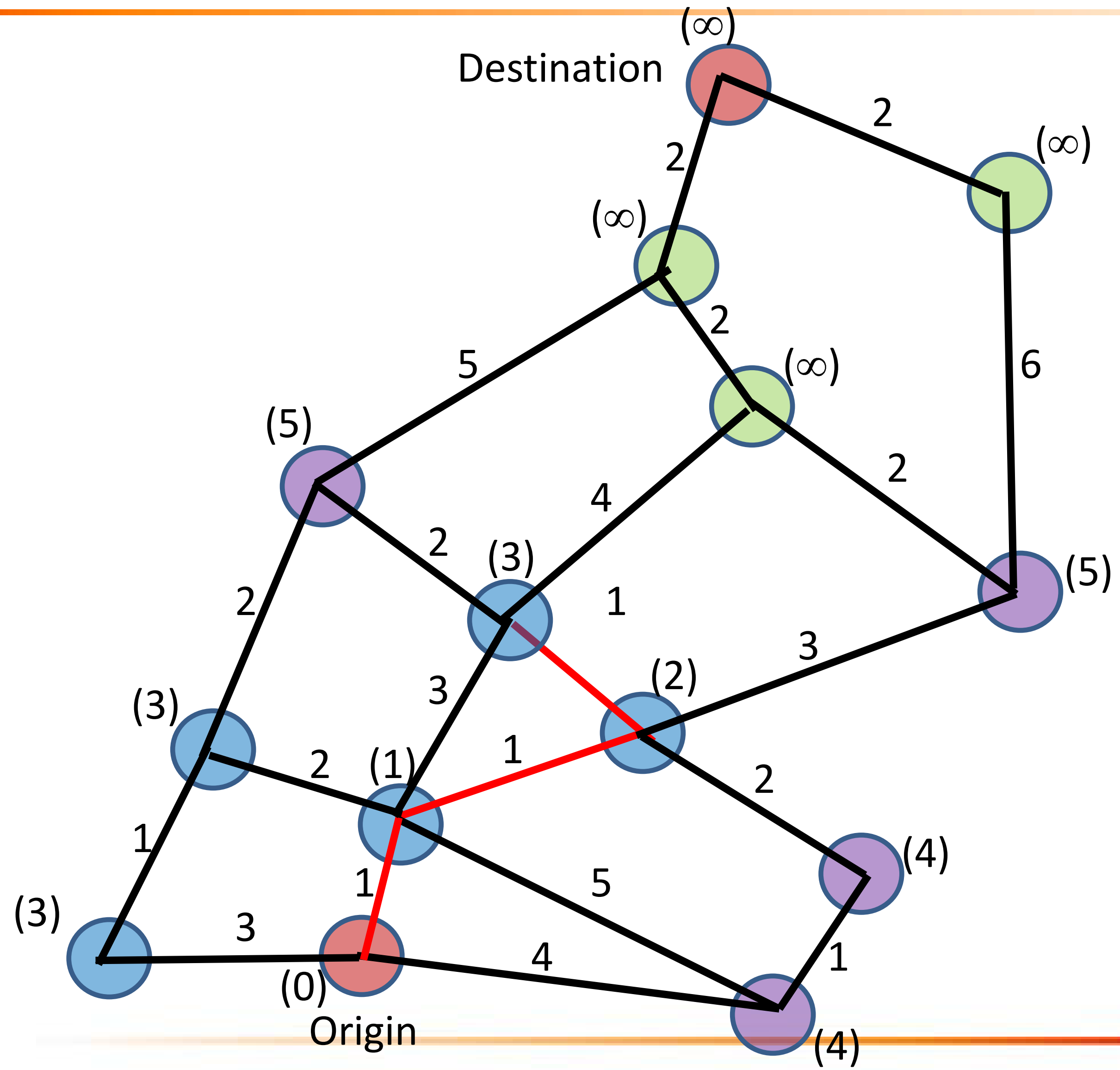
Dijkstra’s Algorithm: Illustration



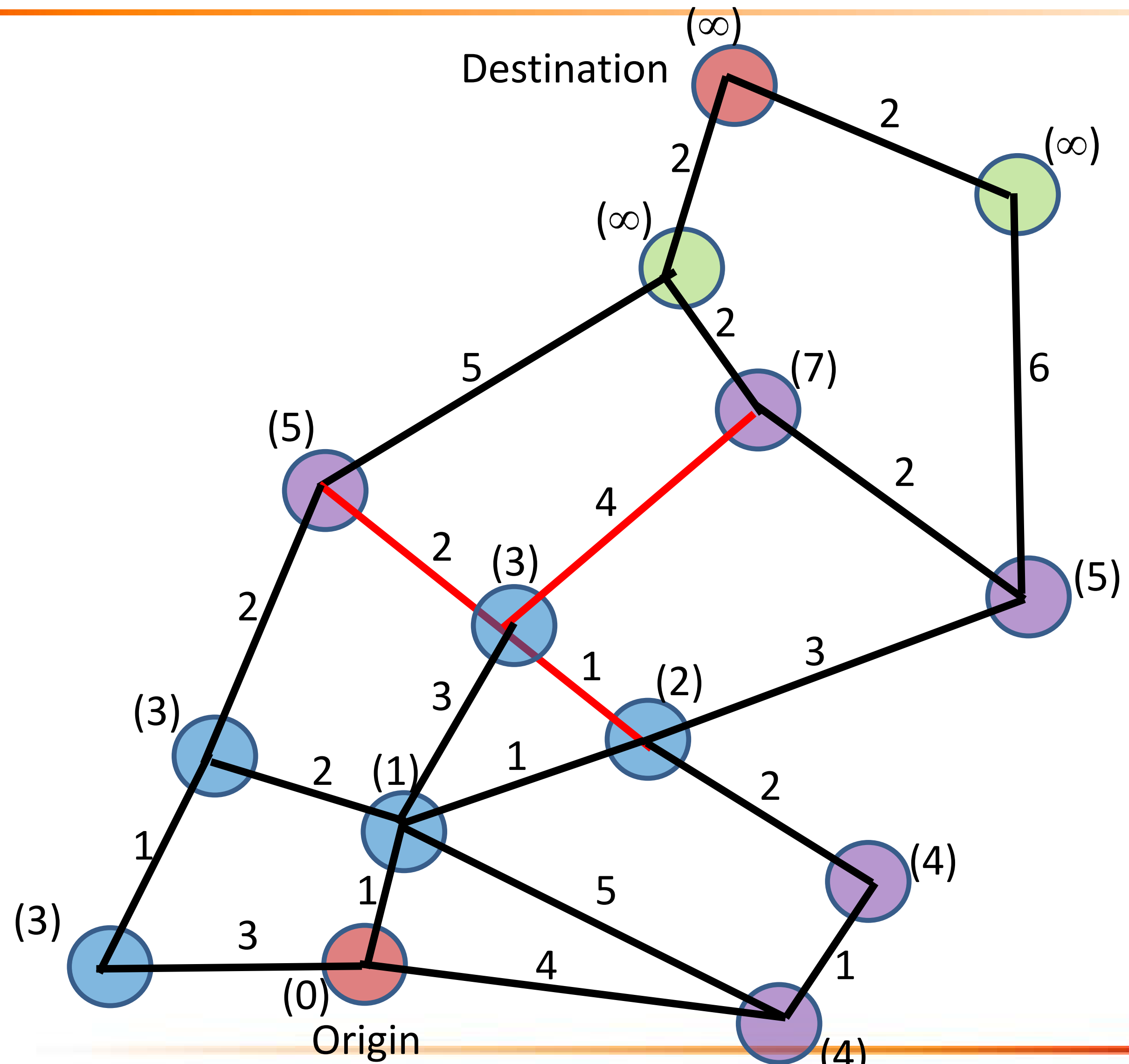
Dijkstra's Algorithm: Illustration



Dijkstra’s Algorithm: Illustration



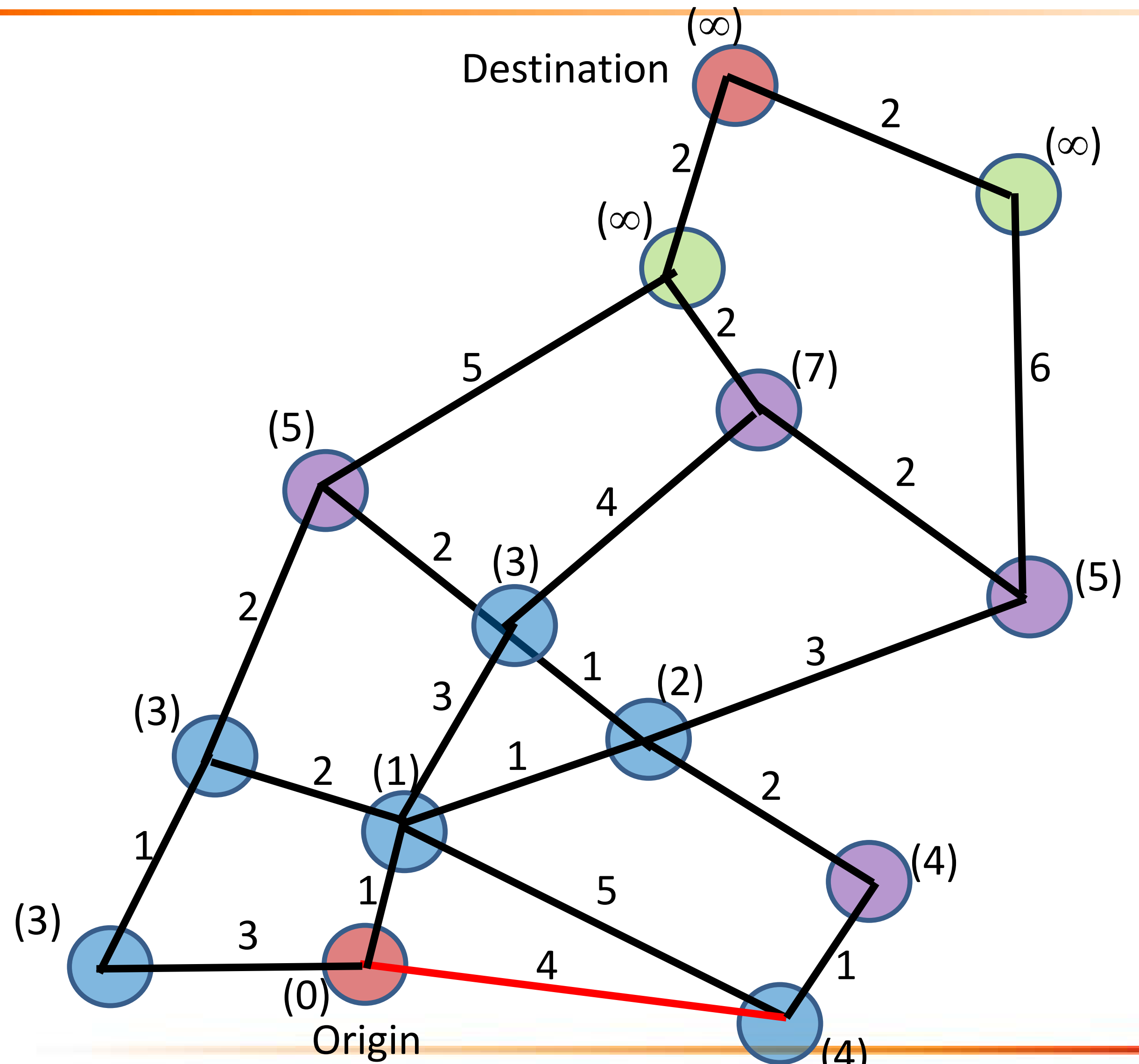
Dijkstra’s Algorithm: Illustration



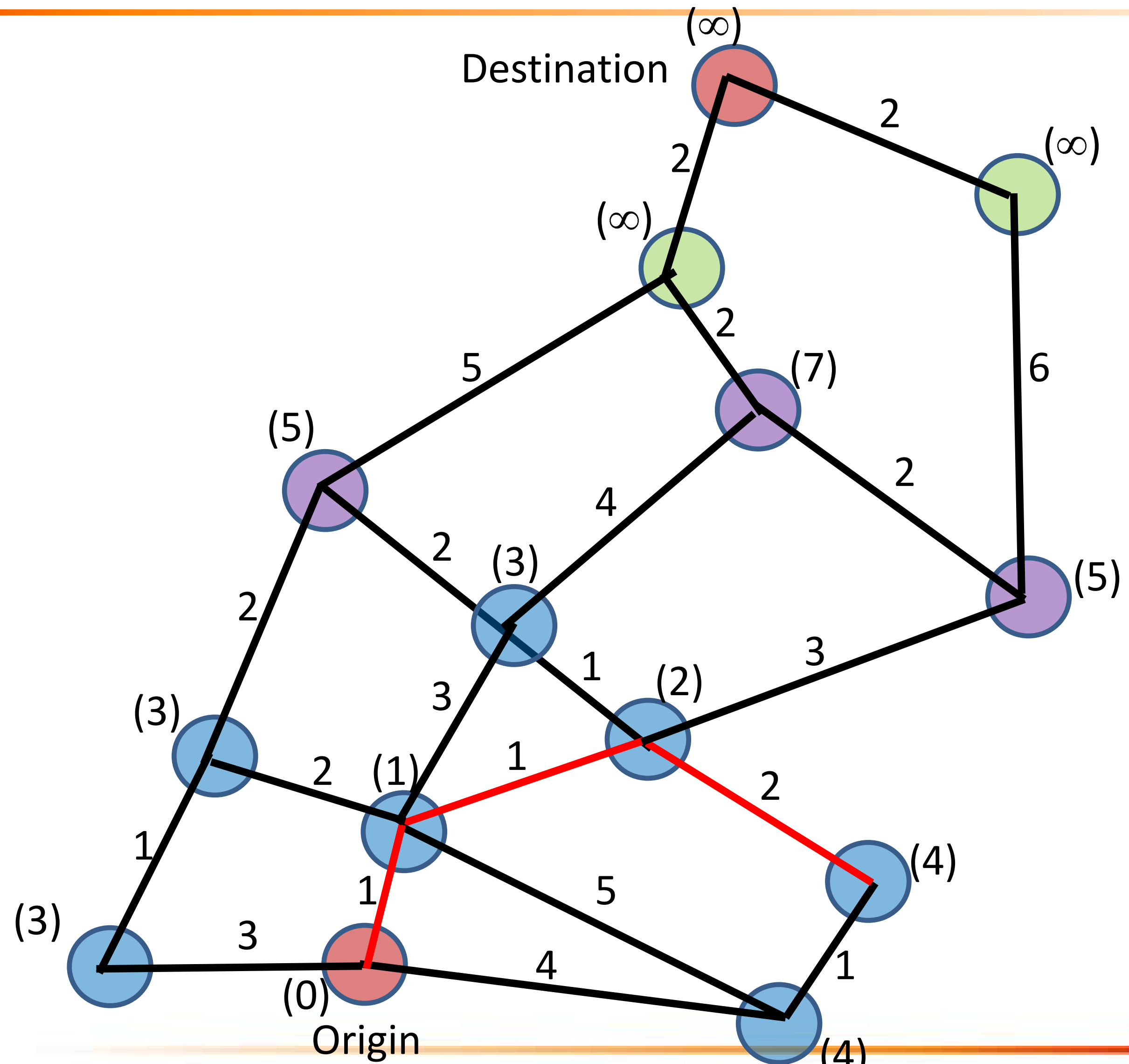
TRANSP-OR

EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

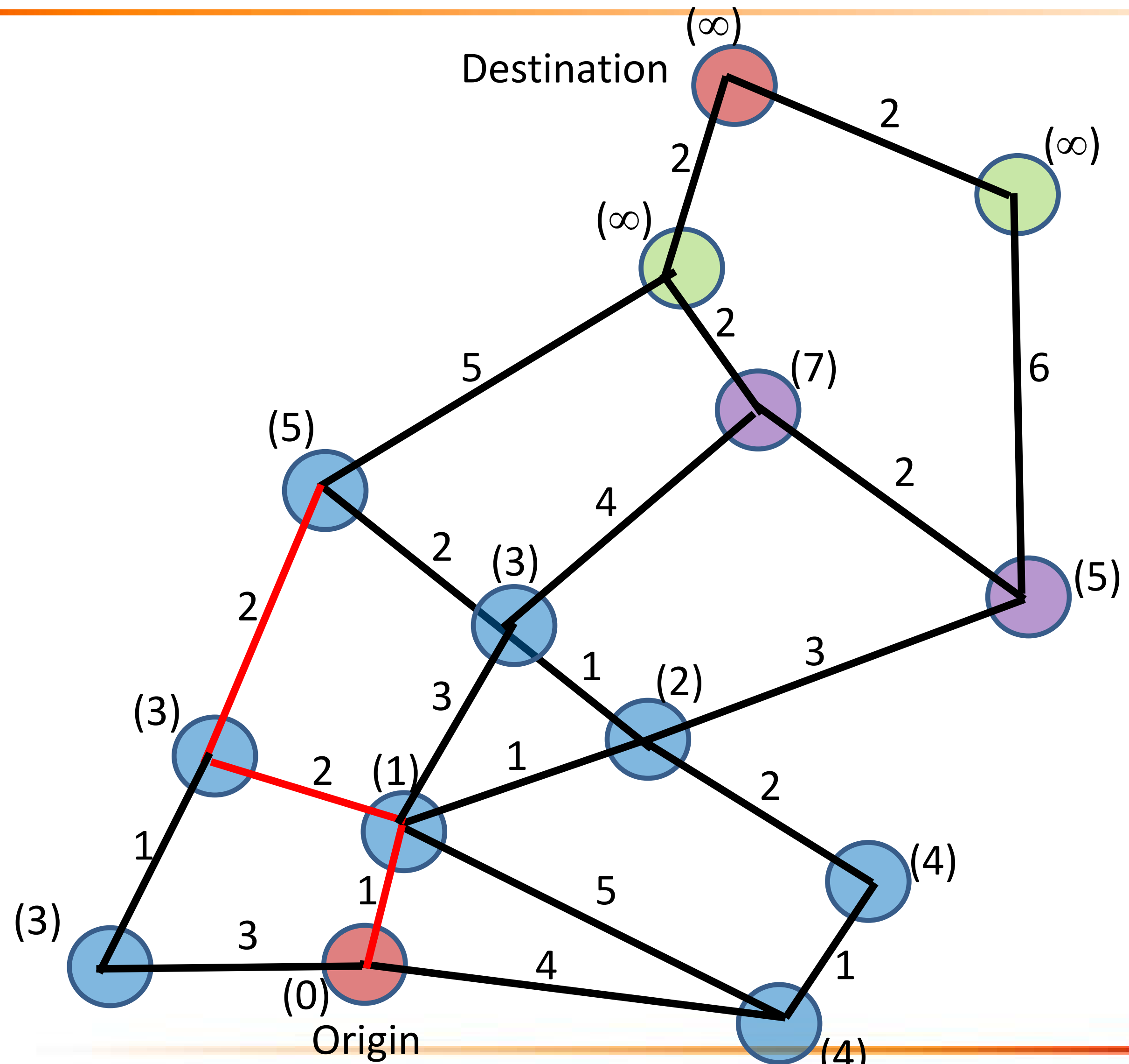
Dijkstra’s Algorithm: Illustration



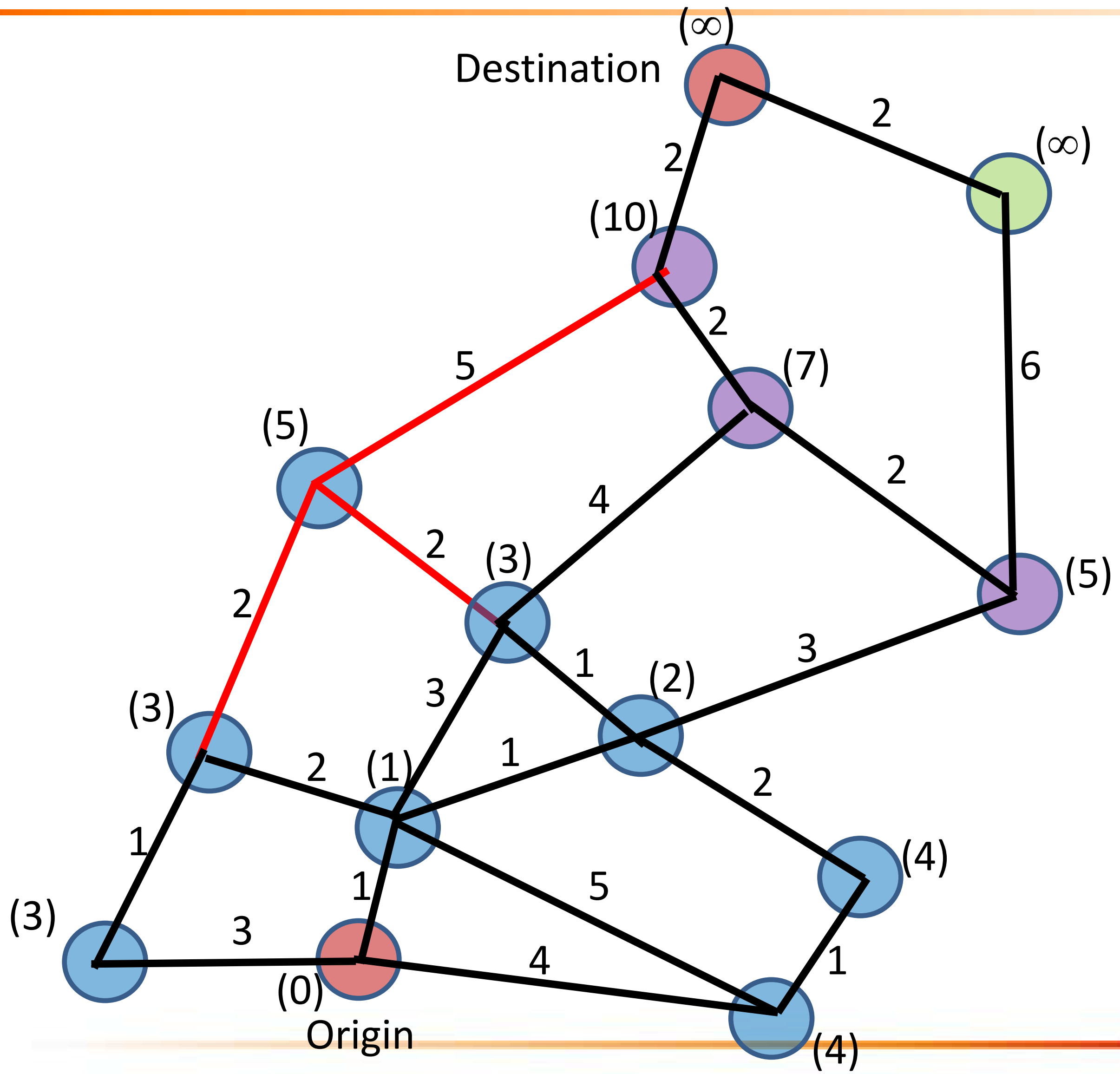
Dijkstra’s Algorithm: Illustration



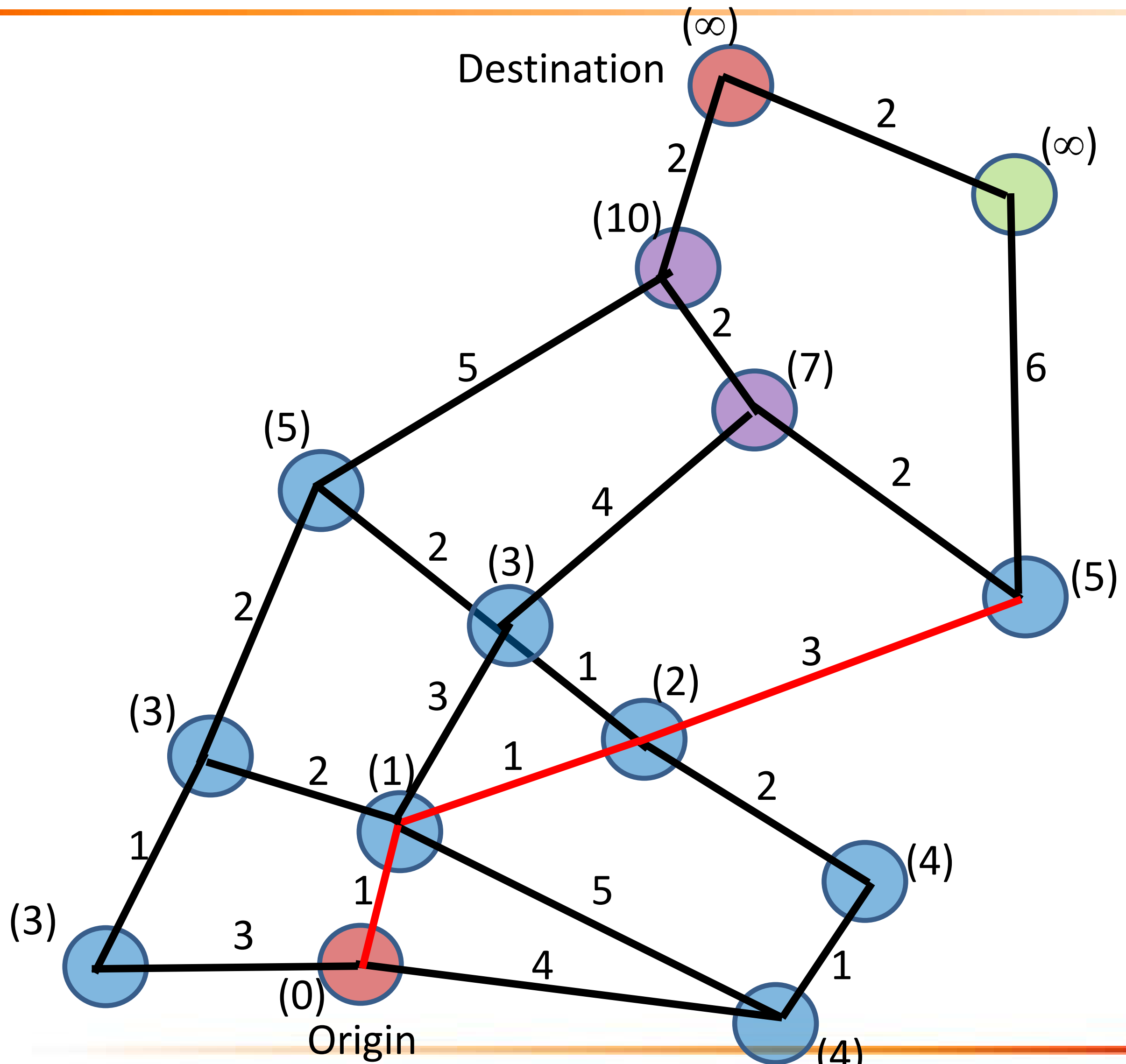
Dijkstra’s Algorithm: Illustration



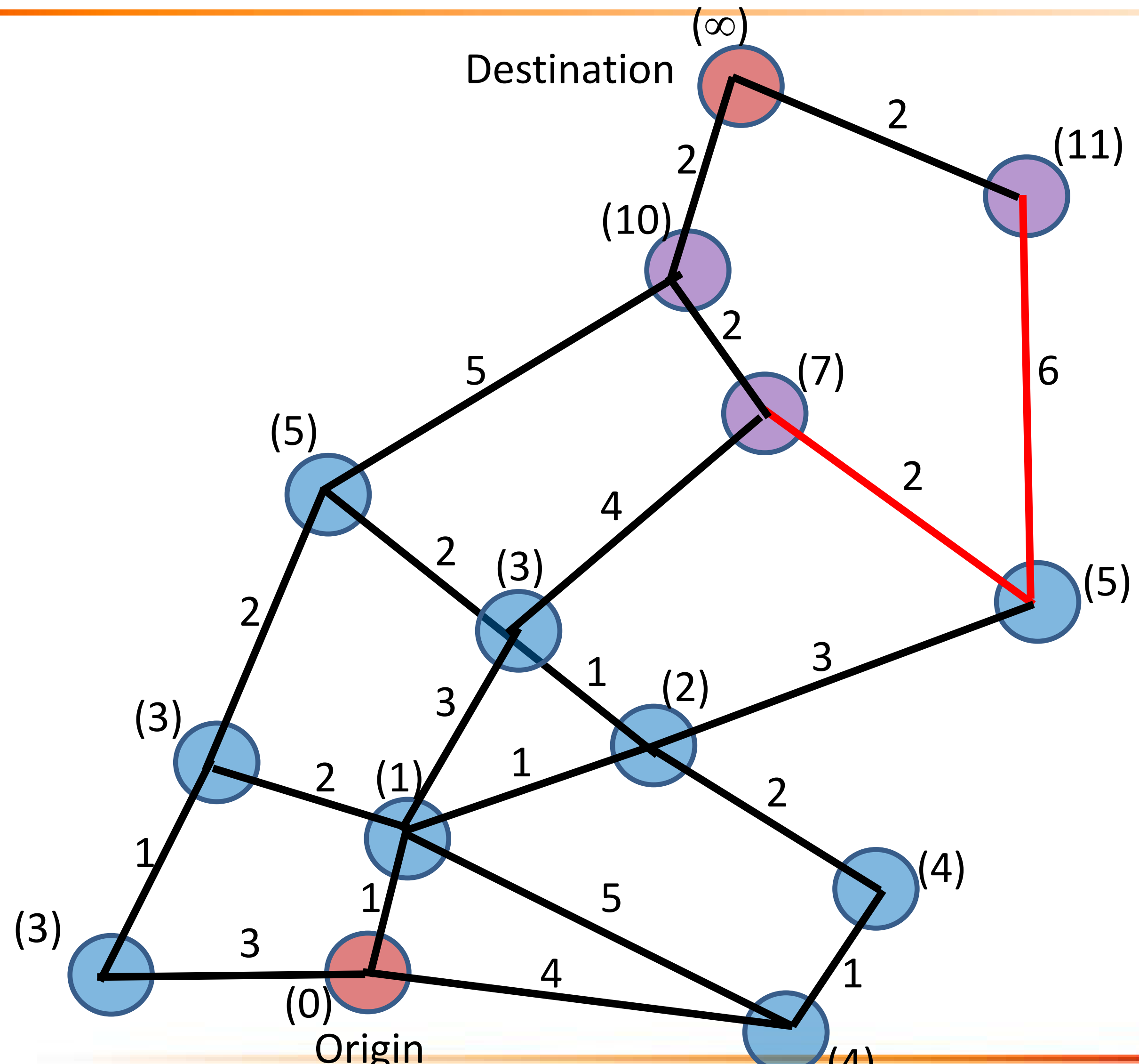
Dijkstra’s Algorithm: Illustration



Dijkstra's Algorithm: Illustration



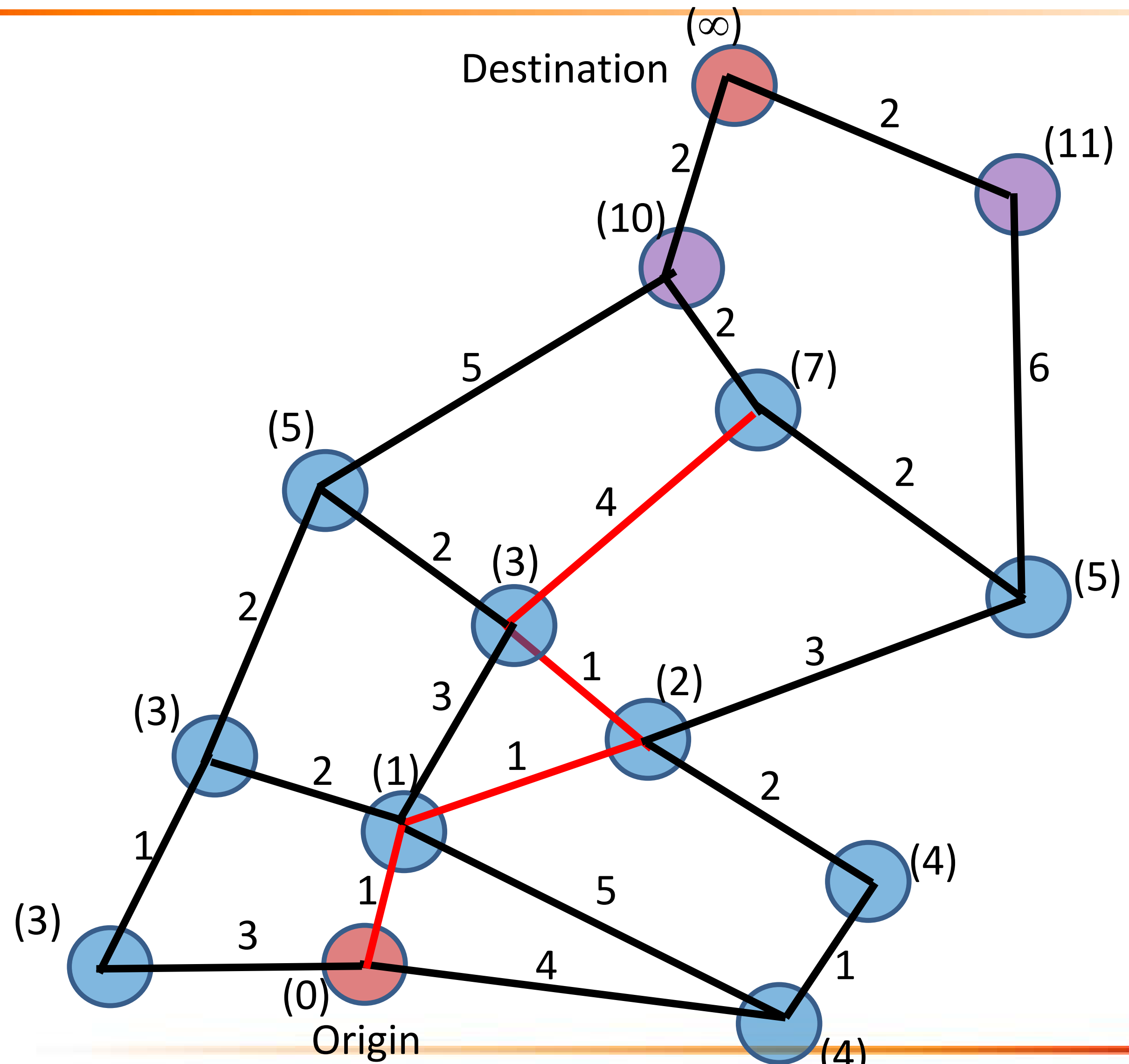
Dijkstra’s Algorithm: Illustration



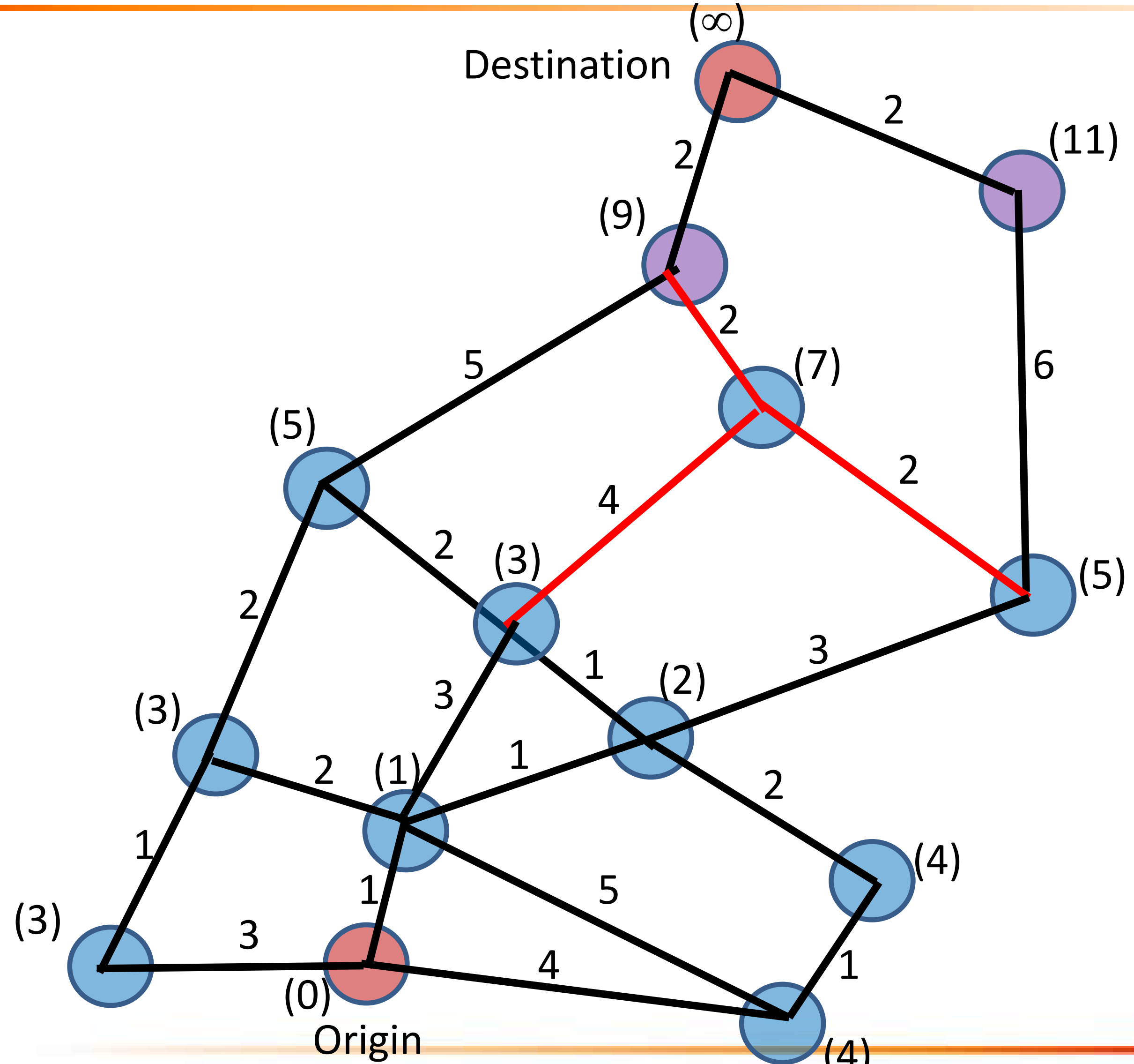
TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

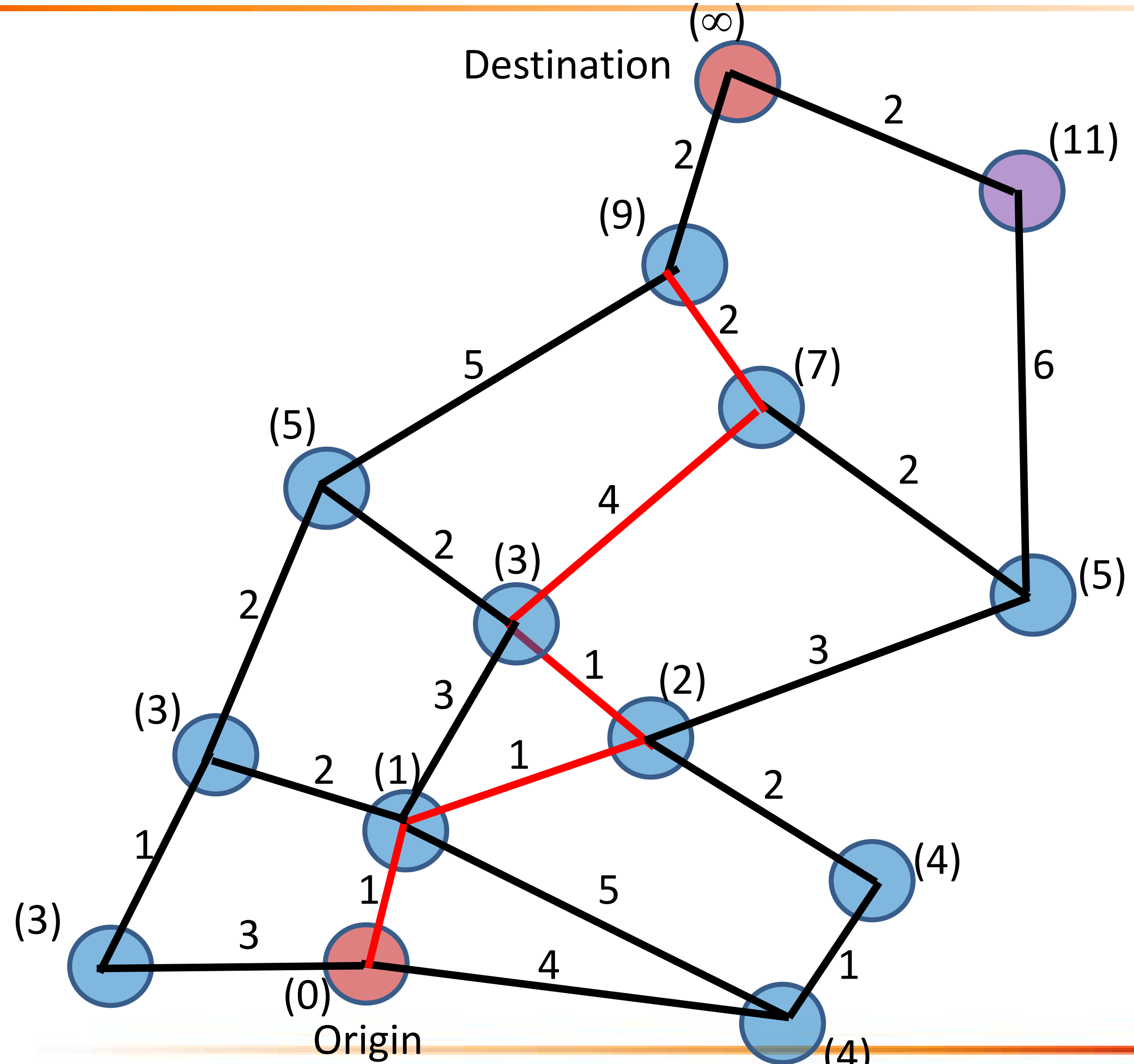
Dijkstra’s Algorithm: Illustration



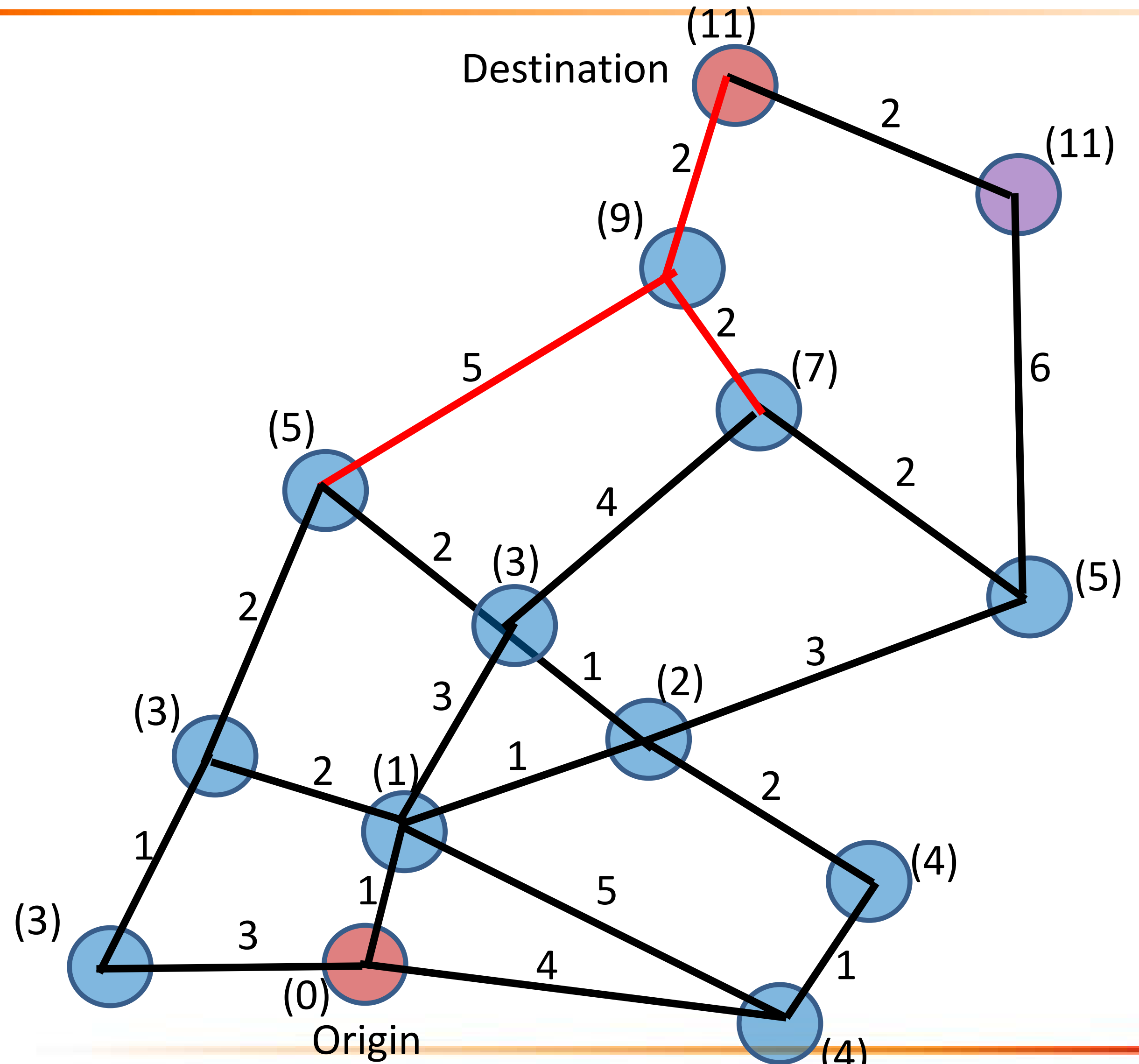
Dijkstra's Algorithm: Illustration



Dijkstra’s Algorithm: Illustration



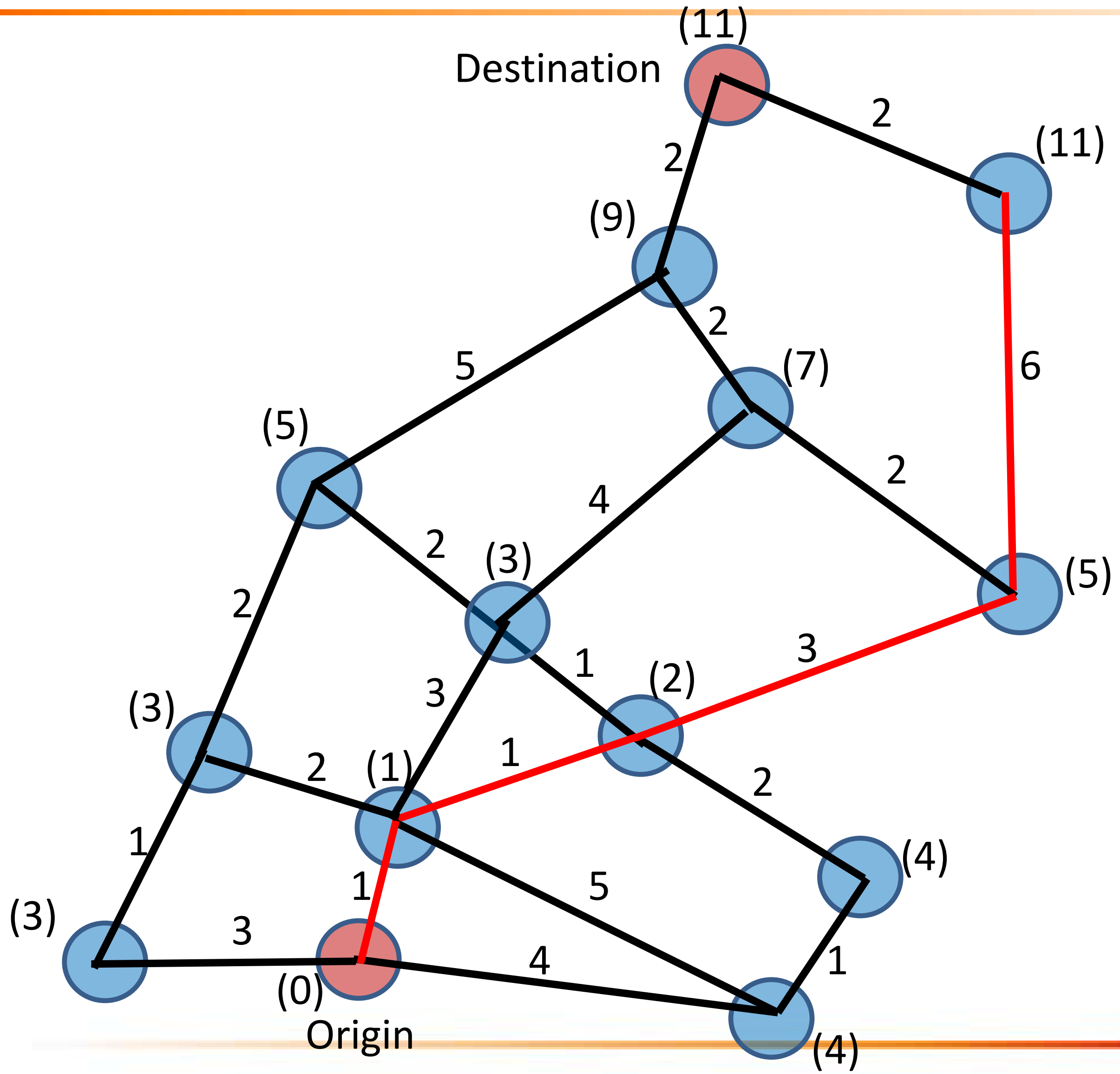
Dijkstra’s Algorithm: Illustration



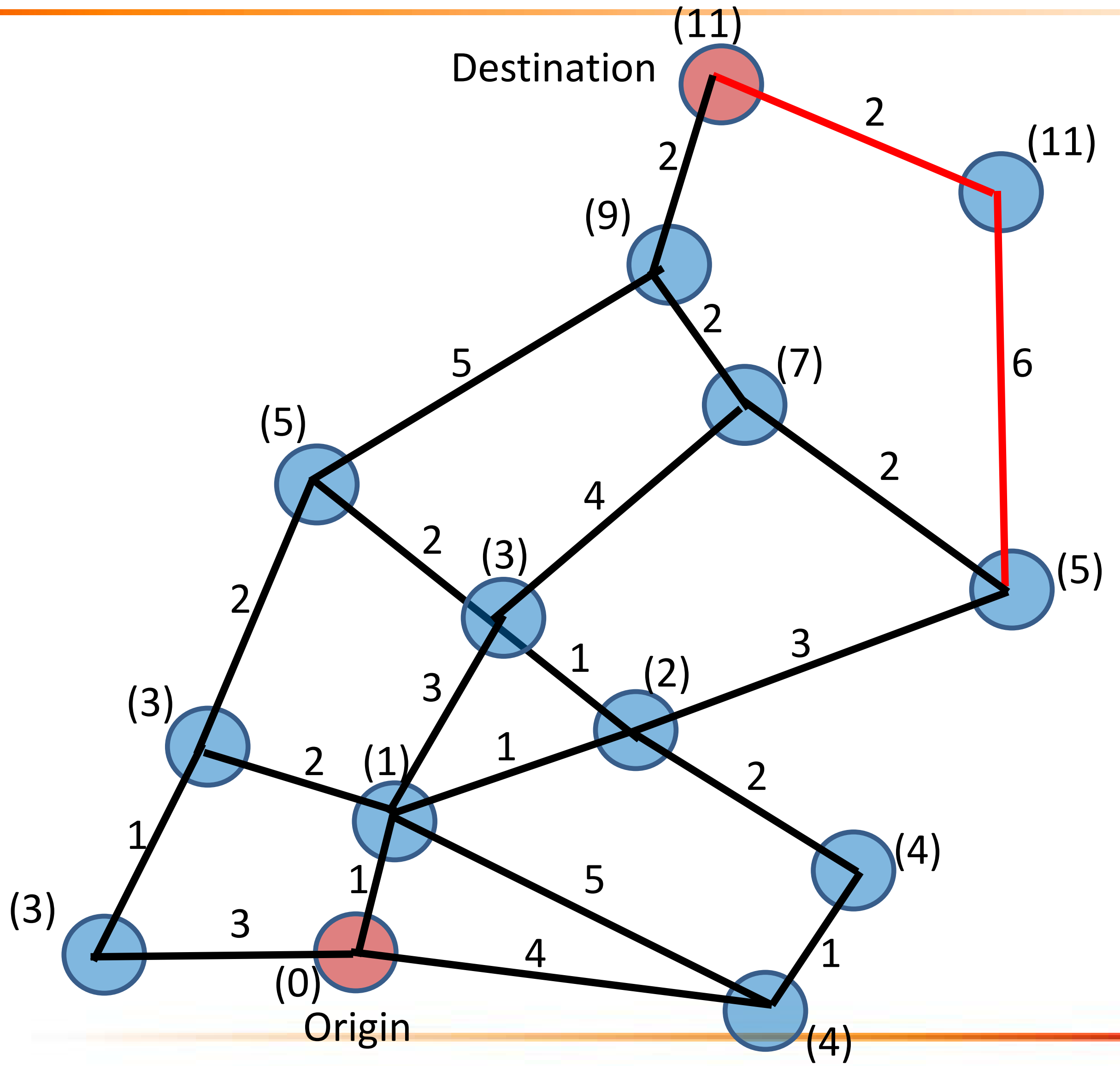
TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

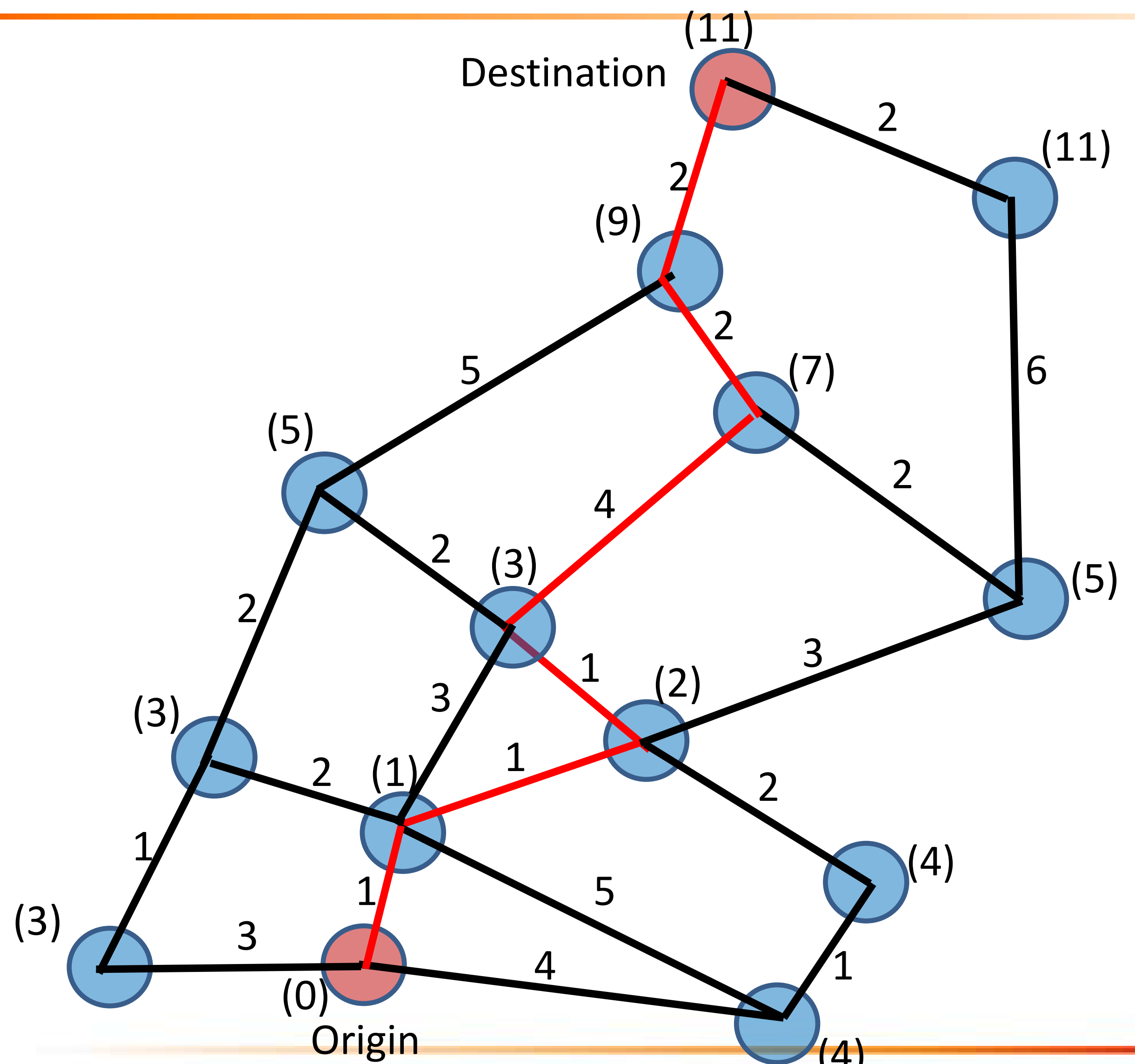
Dijkstra’s Algorithm: Illustration



Dijkstra's Algorithm: Illustration



Dijkstra's Algorithm: Illustration



Integer Program: Concluding Notes

- Any integer program can be formulated as a mathematical program or as a graph (network model)
- Choice depends on convenience of representation
- We have learnt about greedy algorithm and Dijkstra’s algorithm
- During the later sessions, we will learn about solving other formulations of integer programming

Integer Program: Exercise

- N-Queens problem
  - Anybody familiar with chess? Or chess board?
  - Consider a n x n chessboard (a generalized version of a 8 x 8 chessboard)
  - What is the maximum number of queens that can be placed on the board such that no queen is in a confronting position with any other
  - A confronting position is defined if two queens are either in the same row, or same column, or the same diagnol
- This problem can be formulated as a 0-1 integer program

Integer Program: Exercise

- Farmer and his pets
  - A farmer cuts fresh grass from his field and wants to go back home
  - He owns a sheep and a tiger (yes, a tiger!) as pets and takes them to his farm as well
  - As long as he is around, sheep and tiger are well behaved – neither the sheep eats the grass, nor the tiger attacks the sheep
  - But now they arrive at a river bank which that have to cross with a ***small*** boat
  - The boat can carry only two of the four – farmer, sheep, tiger and bundle of grass
  - Obviously only the farmer can steer the boat
  - What is the minimum number of trips it would take to cross the river if all the four are required to reach the other bank without any loss
- How would you formulate this problem?

Airline Industry: History

- Need for commercial aviation recognized as early as 1914 when a private operator flew between two cities in the vicinity of Tampa bay, barely at a height of 15 m (50 feet) above the ground
- In Europe, KLM was one of the first operators connecting London and Amsterdam in 1920 as chartered flight to ferry two British passengers
- First international airline with regular flights commenced between London and Paris in early 1920s
- France connected a mail service to Morocco and named it Aeropostale, but by 1927 it was bankrupt. Aeropostale and several other small airline companies merged to form Air France
- KLM also envisioned the concept of a hub network as the vast Dutch empire had shrunk after the wars

Airline Industry: Recent History

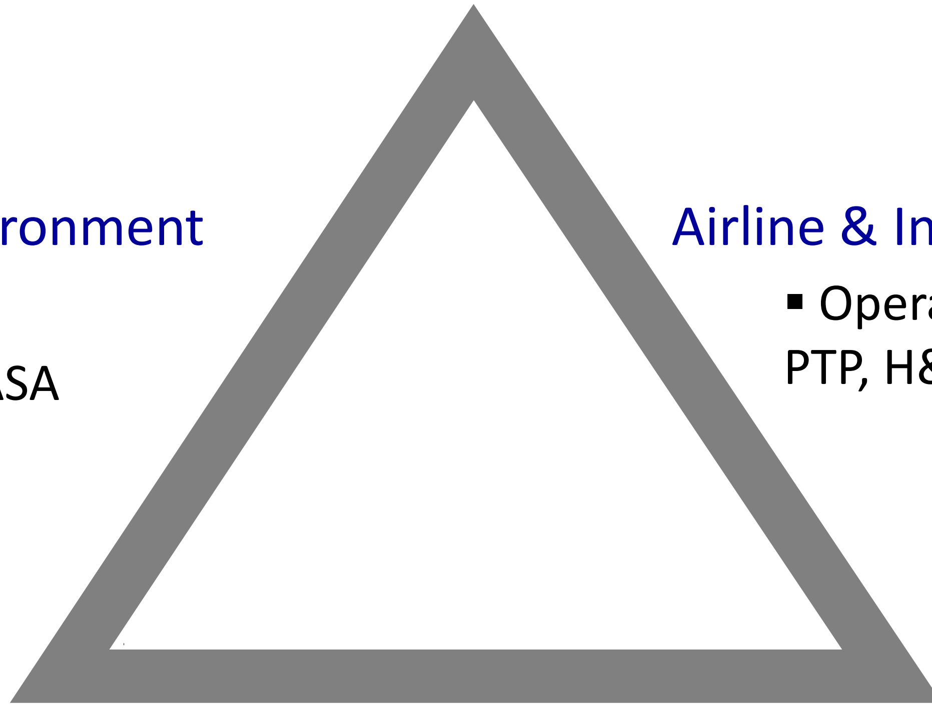
- After WW-II, governments sat together to form regulatory bodies for commercial aviation
- Several government regulated the aviation business to make the service reach to far flung areas
- In 1990s, the industry was deregulated in Europe and it gave birth to several low cost carriers such as Ryan Air and Easyjet
- By 2000s, several traditional airline companies posted heavy losses and consolidation was the only way forward
- KLM was merged with Air France and Swiss Air was rechristened after getting acquired by Lufthansa

Airline Industry: World's Largest Airlines*

Rank	Airline	2010	2009	2008	2007	2006	2005
1	Delta Air Lines	162,614,714	161,049,000	106,070,000	72,900,000	73,584,000	86,007,000
2	United Airlines	145,550,000	81,421,000	86,412,000	68,400,000	69,265,000	66,717,000
3	Southwest Airlines	130,948,747	101,339,000	101,921,000	101,911,000	96,277,000	88,380,000
4	American Airlines	105,163,576	85,719,000	92,772,000	98,162,000	99,835,000	98,038,000
5	Lufthansa	90,173,000	76,543,000	70,543,000	66,100,000	53,400,000	51,300,000
6	China Southern Airlines	76,500,000	66,280,000	57,961,000	56,900,000	48,512,000	43,228,000
7	Ryanair	72,719,666	65,300,000	57,647,000	49,030,000	40,532,000	33,368,585
8	Air France-KLM	70,750,000	71,394,000	73,844,000	74,795,000	73,484,000	70,015,000
9	China Eastern Airlines	64,877,800	44,042,990	37,231,480	39,161,400	35,039,700	24,290,500
10	US Airways	59,809,367	58,921,521	62,659,842	66,056,374	66,102,774	71,580,012

* Source: Wikipedia

Airline Industry: Market Forces



Regulatory Environment

- IATA, ICAO
- In Europe: EASA

Airline & Industry Strategies

- Operating model
PTP, H&S etc

Aircraft & Aerospace Capabilities

- Range and speed
- Maintenance & support
- Air traffic

Airline Industry: Aircraft Fleets

- Major aircraft manufacturers include Airbus, Boeing, Embraer, McDonald Douglas etc.

  Fleet B737

  - B737-300

  Sub-fleet

  - B737-500

- Aircrafts are categorized into fleets and sub-fleets
- Sub-fleets have different capacities, operating costs
- Crew is classified by the fleets they can operate

TRANSP-OR

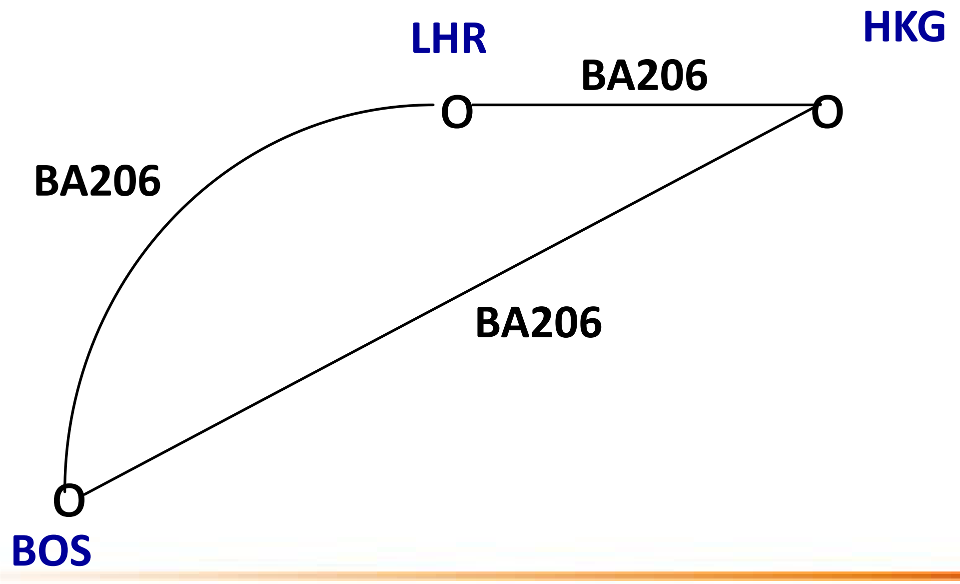
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Airline Industry: Aircraft Maintenance

- Maintenance checks
  - Overnight checks
  - Heavy maintenance checks (e.g. A, B, C checks)
- Every station need not have the required maintenance support for every sub-fleet
- Extended Range Twin Engine Operations (ETOPS)
  - Applies when a twin engine aircraft is flying over-water
  - Between two ocean crossings, the aircraft has to fly a stipulated number of non over-water routes
  - The same mechanic cannot attend both the engines

Airline Industry: Leg and Segment

- An origin-destination pair on which the flight flies non-stop is called a leg
- An origin-destination pair on which the flight might have stops in between is called a segment



Airline Industry: Hub Model

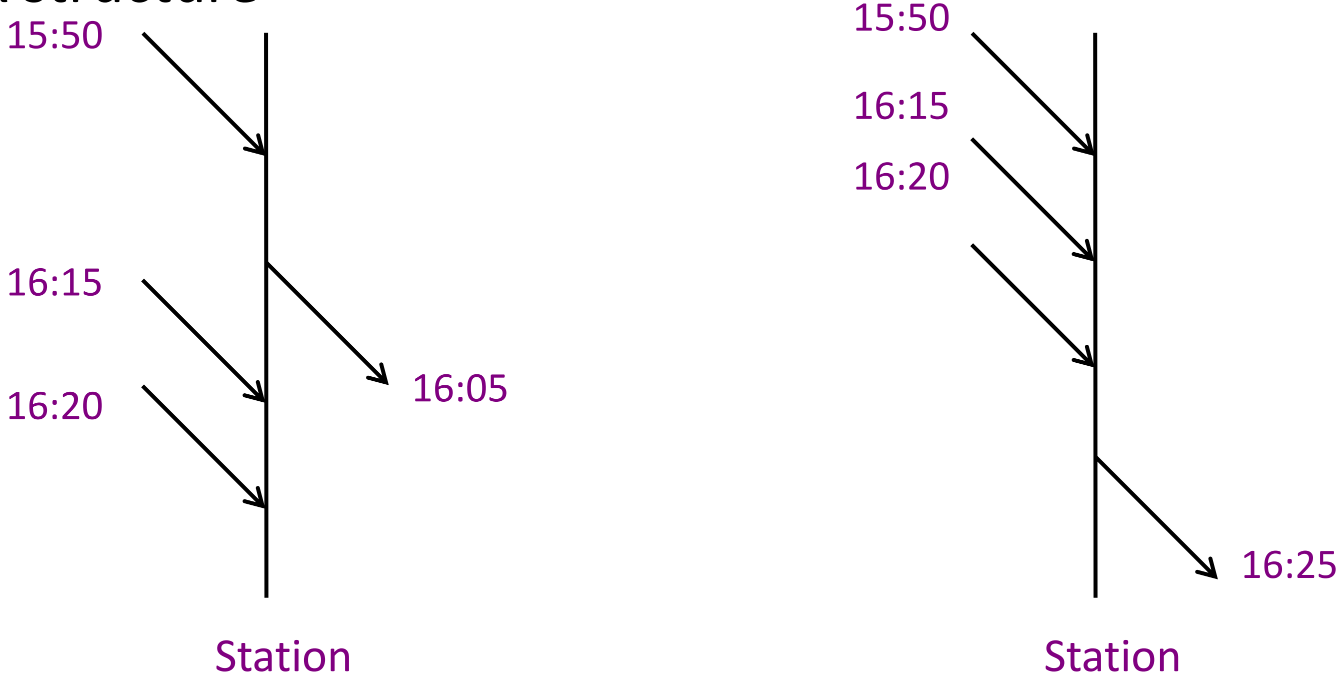
- Hub and Spoke structure gained prominence during the regulation years in North America and Europe

- A hub is a central airport that flights are routed through, and spokes are the routes that planes take out of the hub airport

- The purpose is to save airlines money and give passengers better routes to destinations

Airline Industry: Bank Structure

- Bank structure



- A bunch of arrivals followed by departures
- Directional consideration for bank arrivals/departures is critical
- The bank structure increases the number of connections

Airline Industry: Code Sharing

- Code sharing helps airline expand market share to regions where they don’t even operate
- More than one airlines share their airline codes on a particular route

ORD —— DL402 —— CDG —— DL1404 —— BOM

- Boarding ticket / flight information may show:
  - DL402 ORD to CDG
  - AF192 CDG to DEL

Airline Industry: Performance Metrics

- ASM= Available Seat Mile (#seats x #miles)
- ASK = Available Seat Kilometer (#seats x #km) (= ASM x 1.61)
- RPM= Revenue Passenger Mile (#passengers x #miles)
- RPK= Revenue Passenger Kilometer (#passengers x #km)
- Yield= Revenue per RPM (income / RPM)
- CASM = Cost per available seat mile (Operating cost / ASM)
- RASM = Revenue per available seat mile(Revenue / ASM)
- LF= Load Factor (RMP / ASM = [#passengers] / [#seats])

Airline Industry: Metrics Calculation

- Example – Aircraft with 175 seats flies 2,000 miles with 140 passengers
  - ASM= 175 seats x 2,000 miles = 350,000 seat miles
  - RPM= 140 passengers x 2000 miles = 280,000 passenger miles
- With 15,000€ revenue and 10,000€ operating costs
  - Yield = [15,000€] / [280,000 RPM] = 0.054 €per RPM
  - CASM = [10,000€] / [350,000] = 0.029 €per ASM
  - RASM = [15,000] / [350,000] = 0.043 €per RPM
  - LF = [280,000€] / [350,000] = 0.80 = 80%

Airline Industry: Operations Research

- Airline industry has been one of the first to successfully apply operations research techniques for revenue enhancement as well as driving cost efficiencies
- OR techniques have been widely used for
  - Planning
  - Aircraft Scheduling
  - Crew Scheduling
  - Revenue Management
  - Supply Chain Management
  - Operations