Tutorial 2 : Shortest path in graphs

Operations research, 3rd semester.

2022

Exercice 1 — *Routing table in a network*

In a packet switching network, we often need a table indicating, for each node, where an entering packet must be send in order to reach its destination, whatever its source is. This exercice explains how to build such a table.

Let be a network with 4 nodes, describe by the following adjacency matrix of a graph :

	1	**2**	**3**	**4**
1	-	3	-	3
2	2	-	2	2
3	1	-	-	1
4	-	4	4	-

1. Give two algorithms that can be used to compute the shortest path from the node **1** to every other node.
2. In fact, instead of one source, we need the paths from every possible source to every possible destination. Apply the Floyd-Warshall algorithm to do so. Give the matrices π and P at each step of the algorithm.
3. Using P, give the optimal routing table T in that network, *i.e.* for each couple (u, v), give the successor of u in a shortest path from u to v.

Algorithme 1 Floyd-warshall ($G = (X, U)$)

$\pi_{ij}^{(0)} :=$ weight of (i, j)
// if the arc (i, j) does not exist, the weight is $+\infty$; unless $i = j$ where the weight is 0

$P^{(0)}(i, j) := i$ //At the end of the algorithm, $P^{(n)}(i, j)$ contains the predecessor of j in a shortest path from i to j

for $m = 1, 2, \ldots, n$ **do**
  for $i = 1, 2, \ldots, n$ **do**
    for $j = 1, 2, \ldots, n$ **do**
      // Is is shorter to go from i to j by going through m ?
      if $(\pi_{im}^{(m-1)} + \pi_{mj}^{(m-1)} < \pi_{ij}^{(m-1)})$ **then**
        $\pi_{ij}^{(m)} := \pi_{im}^{(m-1)} + \pi_{mj}^{(m-1)}$
        $P_{ij}^{(m)} := P_{mj}^{(m-1)}$
      else
        $\pi_{ij}^{(m)} := \pi_{ij}^{(m-1)}$
        $P_{ij}^{(m)} := P_{ij}^{(m-1)}$

Exercice 2 — *Optimality condition*

1. Let $G = (X, U)$ be a directed graph with a node s, and where each arc (u, v) is weighted with $\omega(u, v) \in \mathbb{R}$. We associate each node u with a number d_u. Demonstrate the following theorem : The graph has no absorbing circuit if and only if the system of inequations $\{ d_v - d_u \leq \omega(u, v)$ for every arc $(u, v) \in U \}$ has a solution. In that case, setting each number $d_u \in \mathbb{R}$ to the value of a shortest path from s to u is a solution.
2. Transform the following system into a shortest path problem in a graph and solve it :

$$\begin{aligned} x_3 - x_4 &\leq 5 \\ x_4 - x_1 &\leq -10 \\ x_1 - x_3 &\leq 8 \\ x_2 - x_1 &\leq -11 \\ x_3 - x_2 &\leq 2 \end{aligned}$$

3. Same question for this system :

$$\begin{aligned} x_3 - x_2 &\leq 5 \\ x_4 - x_3 &\leq -2 \\ x_4 - x_2 &\leq 4 \\ x_1 - x_2 &\leq 3 \\ x_5 - x_1 &\leq 2 \\ x_3 - x_5 &\leq 7 \\ x_5 - x_4 &\leq -1 \end{aligned}$$

Exercice 3 — *Disjoint cuts and shortest paths*

Let $G = (V, A)$ be a directed graph and x and y be two nodes of G. A cut separating x from y is a subset of arcs of A such that removing those arcs from G destroy every directed path from x to y.

1. Show that, after removing every arc of a cut C separating x from y, V can be partionned into two sets X and $Y = V \backslash X$ such that every arc of G linking X to Y is in C.
2. Show that the maximum number of disjoint cut separating x from y equals the number of arcs in a shortest path from x to y.

Exercice 4 — *Currency converter*

Let G be a graph where each node is a currency and where an arc (u, v) exists if and only if it is possible to convert the currency u into the currency v. In that case, the arc is weighted with the exchange rate.

What condition must this graph satisfy so that someone can become as rich as he wants ? Transform this graph into a weighted graph H such that searching for a shortest path in H is equivalent to deciding if someone can become as rich as he wants. Is the Dijkstra algorithm useful to answer this question ?