Operation Research and Transport
Braess's Paradox

V. Leclère (ENPC)

April 24th, 2017

What will this course be about ?

- Understanding how people choose their way through a transportation network.
- having an idea on how to compute efficiently :
  - the shortest path on a network
  - the equilibrium on a network
- A practical work to compute this equilibrium on a computer
- Snapshots of other problems :
  - SNCF (03/05)
  - Google (17/05)
  - Air France (31/05)

Contents

Transportation Planning Process

1. Organization and definition
2. Base year inventory
3. Model analysis
    1. trip generation
    2. trip distribution
    3. modal split
    4. traffic assignement
4. Travel forecast
5. Network evaluation

Urban Transportation Network Analysis

Input of the analysis:

- transportation infrastructure and services (street, intersections...)
- transportation system and control policies
- demand for travel.

Two stage analysis:

- First stage: determining the congestion, i.e. calculating the flow through each component of the network.
- Second stage : computing measure of interests according to the flow.
    - travel time and costs,
    - revenue and profit of ancilliary services,
    - welfare measures (accessibility, equity),
    - flow by-products (pollution, change in land-value)...

Why do we need a system approach ?

- Some decision could be taken according to local measure. For example traffic light can be timed according to data on current usual traffic at the intersection.
- However most decision will impact the travel time / confort. Hence, some people will adapt their usual transit route.
- Consequently, the congestion on the network will change, changing time / confort of other part of the system and inducing other people to adapt their path...
- After some time these ripple effect will lessen, and the system will reach a new equilibrium.

Equilibrium in Markets

- For a given product, in a perfectly competitive market we have:
  - a production function giving the number of product companies are ready to make for a given price;
  - a demand function giving the number of product consumer are ready to buy for a given price.
- In some cases, especially in transportation, the price is not the only determinant factor. Regularity, fiability, ease of use, comfort are other determinant factor.
- In the remaining of the course we will be speaking of cost of each path, the cost factoring in all of this factors.

Nash Equilibrium : Prisonner's Dilemna

Two guys got caught while dealing chocolate. As he is missing hard evidence the judge offer them a deal.

- If both deny their implication they will get 2 month each.
- If one speak, and the other deny, the first will get 1 month while the other will get 5 months.
- If both speak they get 4 month each.

Question : what is the equilibrium ?

Nash Equilibrium

- In game theory we consider multiple agents $a \in \mathcal{A}$, each having a set of possible action $u_a \in \mathcal{U}_a$.
- Each agent earn a reward $r_a(u)$ depending on his action, as well as the other actions.
- A Nash equilibrium is a set of actions $\{u_a\}_{a\in\mathcal{A}}$, such that no player can increase his reward by changing is action if the other keep these actions :

$$\forall a \in \mathcal{A}, \quad \forall u'_a \in \mathcal{U}_a, \qquad r_a(u'_a, u_{-a}) \leq r_a(u_a, u_{-a}).$$

- A recommandation can be followed only if is a Nash Equilibrium.

Game Theory : a few classes

- Number of player
  - 2 (most results)
  - $n > 2$ (hard, even with 3)
  - an infinity.
- Objective
  - zero-sum game (e.g. chess)
  - cooperative : everybody share the same objective (e.g. pandemia)
  - generic (e.g. Prisonner dilemna)

Game theory : a few definitions

Definition

A Nash equilibrium is a set of action such that no player can unilaterally improve its pay-off by changing is action.

Definition

A Pareto efficient solution is a set of action such that no other set of actions can strictly improve at least one player pay-off without decreasing at least another.

Definition

A social optimum is a set of action minimizing the pay-off average.

Exercises :

- what about Prisonner's Dilemma ?
- what about Zero Sum games ?

Exercise: A beautiful mind

A beautiful mind : 19'

- Is the solution proposed by Nash a Nash equilibrium ?
- Is the solution proposed by Nash a Pareto Optimum ?
- Is the solution proposed by "Smith" a Nash equilibrium ?
- Is the solution proposed by "Smith" a Pareto Optimum ?
- Any other suggestion ?

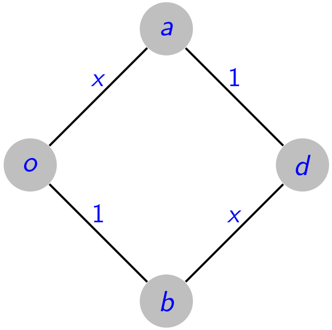
Contents

Game theory in road network

- People choose their means of transport (e.g. car versus public transport), their time of departure, their itinerary.
- Each user choose in its own interest (mainly the shortest time / lowest cost).
- The time depends on the congestion, which means on the choice of other users.
- Hence, we are in a game framework : users interact with conflicting interest.

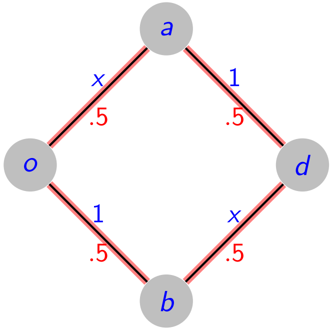
A very simple framework

- Consider a large group of person want to go from the same origin o to the destination d, at the same time, with the same car.
- We look at a very simple graph with two roads, each composed of two edges.
- The time on each edges of the road is given as a function of the number of person taking the given edge.



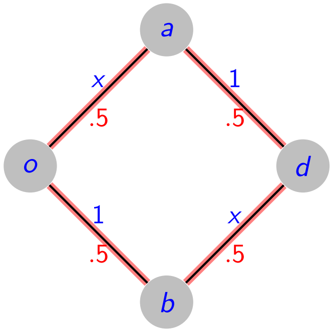
A very simple framework

- Consider a large group of person want to go from the same origin o to the destination d, at the same time, with the same car.
- We look at a very simple graph with two roads, each composed of two edges.
- The time on each edges of the road is given as a function of the number of person taking the given edge.



A very simple framework

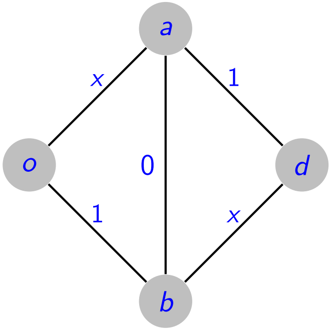
- Consider a large group of person want to go from the same origin o to the destination d, at the same time, with the same car.
- We look at a very simple graph with two roads, each composed of two edges.
- The time on each edges of the road is given as a function of the number of person taking the given edge.



Total time : 1.5

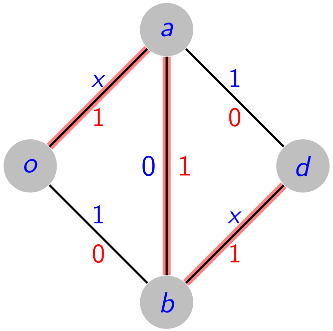
Adding a road

- Now someone decide to construct a new, very efficient road with cost 0.
- What is the new equilibrium ?



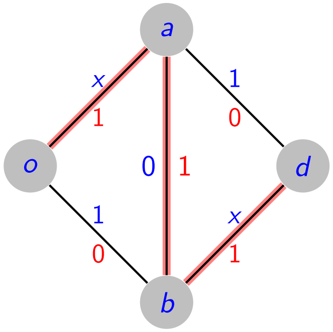
Adding a road

- Now someone decide to construct a new, very efficient road with cost 0.
- What is the new equilibrium ?



Adding a road

- Now someone decide to construct a new, very efficient road with cost 0.
- What is the new equilibrium ?
- Notice that the time for every user as increased ! This is the cost of anarchy.



Total time : 2

Definitions snapshot

On this example we can compare :

- User Equilibrium (UE), with global cost 2
- System Optimum (SO), with global cost 1.5
- price of anarchy : $4/3$.

Definition

A Wardrop (User) Equilibrium, is a repartition of flow such that no single user can improve its cost (travel time) by unilaterally changing routes.

Real case examples

- 42d Street of New York. (New York Times, 25/12/1990).
- Stuttgart 1969 (a newly built road was closed again), Séoul 2003 (6 lanes highway was turned into a park).
- New York 2009 (closed some places with success)
- In 2008, researcher found road in Boston and NYC that should be closed to diminish traffic.
- Steinberg and Zangwill showed that Braess paradox is more or less as likely to occur as not.
- Rapoport's experiment (2009):
  - A group of 18 students is presented with the problem of repetively (40 times) choosing its road on the graph, earning money for the experiment : fastest meaning more money.
  - Then the graph is modified (either by adding the 0 cost road, or retiring it).
  - Conclusion : after a few iteration the observed repartition is close to the theoretical one with some oscillations.
  - Then tested on a bigger network.

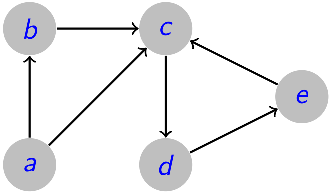
Exercises

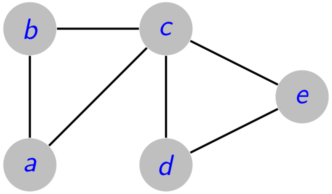
- Two nodes : a and b
- Two edges : (from a to b): 1 and 2
- Total number of trips : 1000
- Costs : $c_1(x_1) = 5 + 2x_1$, $c_2(x_2) = 10 + x_2$.
- Question : what is the repartition of the trips along the two edges ?
- Same question with $c_1(x_1) = 15(1 + 0.15(\frac{x_1}{1000})^4)$, $c_2(x_2) = 20(1 + 0.15(\frac{x_2}{3000})^4)$?

Contents

What is a Graph ?

- A graph is one of the elementary modelisation tools of Operation Research.
- A directed graph (V, A) is defined by
  - A finite set of n vertices V
  - A finite set of m arrows each linked to an origin and a destination.
- A graph is said to be undirected if we do not distinguish between the origin and the destination.





A few definitions

Consider a directed graph (V, A).

- If $(u, v) \in A$, u is a predecessor of v, and v is a successor of u.
- A path is a sequence of arrows $\{a_k\}_{k \in [\![1,n]\!]}$, such that the destination of one arrow is the origin of the next. The origin of the first arrow is the origin of the path, and the destination of the last arrow is the destination of the path.
- A (directed) graph is connected if for all $u, v \in V$, there is a u-v-path.
- A cycle is a path where the destination vertex is the origin.

A weighted graph

- A weighted (directed) graph is a (directed) graph (V, A) with a weight function $c : A \to \mathbb{R}$.
- The weight of a $s-t-$path p is sum of the weights of the arrows contained in the path :

$$c(p) := \sum_{a \in p} c(a).$$

- The shortest path from o to d is the path of minimal weight with origin o and destination d.
- An absorbing cycle is a cycle of strictly negative weight.

Contents

An optimality condition

The methods we are going to present are based on a label function over the vertices. This function should be understood as an estimate of the cost of the shortest path between the origin and the current vertex.

Theorem

Suppose that there exists a function $\lambda : V \mapsto \mathbb{R} \cup \{+\infty\}$, *such that*

$$\forall (i,j) \in A, \qquad \lambda_j \leq \lambda_i + c(i,j).$$

Let P *be a path joigning* o *to* i_k, *such that*

$$\forall (i,j) \in P, \qquad \lambda_j = \lambda_i + c(i,j).$$

Then P *is a shortest path from* o *to* i_k.

A generic algorithm

We keep a list of candidates vertices $U \subset V$, and a label function $\lambda : V \mapsto \mathbb{R} \cup \{+\infty\}$.

```
$U := \{o\}$ ;
$\lambda(o) := 0$ ;
$\forall v \neq o, \quad \lambda(v) = +\infty$ ;

while $U \neq \emptyset$ do
    choose $u \in U$ ;
    for $v$ successor of $u$ do
        if $\lambda(v) > \lambda(u) + c((u, v))$ then
            $\lambda(v) := \lambda(u) + c((u, v))\}$;
            $U := U \cup \{v\}$;
    $U := U \setminus \{u\}$ ;
```

Algorithm properties

- If $\lambda(u) < \infty$ then $\lambda(u)$ is the cost of a o-u-path.
- If $u \notin U$ then
  - either $\lambda(i) = \infty$ (never visited)
  - or

$$\text{for all successor } v \text{ of } u, \qquad \lambda(v) \leq \lambda(u) + c(u,v).$$

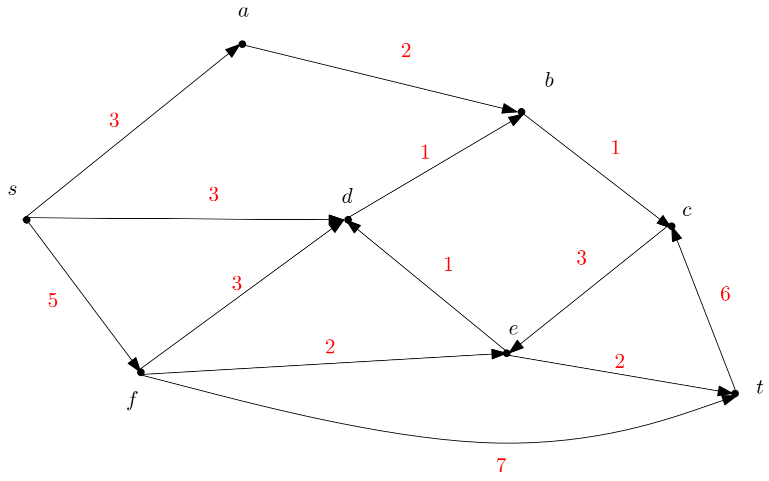
- If the algorithm end $\lambda(u)$ is the smallest cost to go from o to u.
- Algorithm end iff there is no path starting at o and containing an absorbing circuit.

Dijkstra's algorithm

Assume that all cost are non-negative.

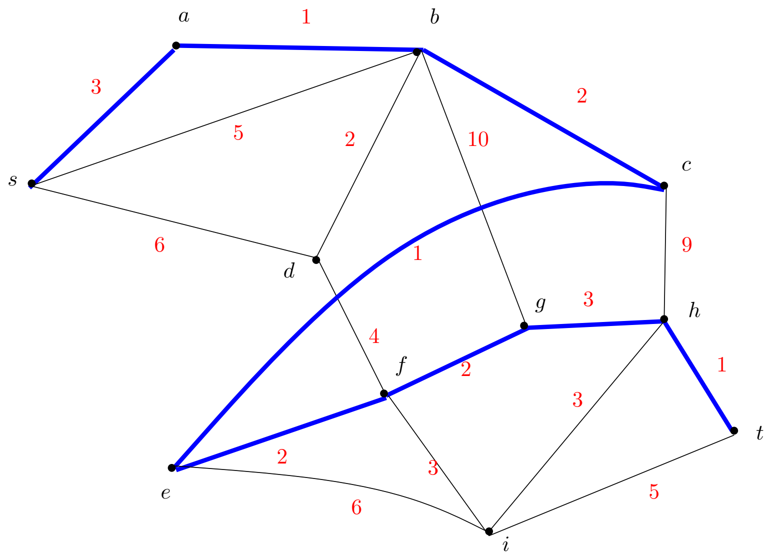
```
$U := \{s\}$ ;
$\lambda(s) := 0$ ;
$\forall v \neq s, \quad \lambda(v) = +\infty$ ;

while $U \neq \emptyset$ do
    choose $u \in \arg\min_{u' \in U} \lambda(u')$ ;
    for $v$ successor of $u$ do
        if $\lambda(v) > \lambda(u) + c((u, v)$ then
            $\lambda(v) := \lambda(u) + c((u, v)\}$;
            $U := U \cup \{v\}$;
    $U := U \setminus \{u\}$ ;
```

Algorithm 1: Dijkstra algorithm


a
b
c
d
e
f
s
t
3
2
1
1
3
3
5
1
3
6
2
2
7

Application example

s	a	b	c	d	e	f	t
(0)	(∞)	(∞)	(∞)	(∞)	(∞)	(∞)	(∞)
0	(3)	(∞)	(∞)	(3)	(∞)	(5)	(∞)
0	3	(5)	(∞)	(3)	(∞)	(5)	(∞)
0	3	(4)	(∞)	3	(∞)	(5)	(∞)
0	3	4	(5)	3	(∞)	(5)	(∞)
0	3	4	5	3	(8)	(5)	(∞)
0	3	4	5	3	(7)	5	(12)
0	3	4	5	3	7	5	(9)
0	3	4	5	3	7	5	9


a
1
b
3
5
2
10
2
s
c
6
1
9
d
g
3
h
4
f
2
1
3
t
2
3
e
6
5
i

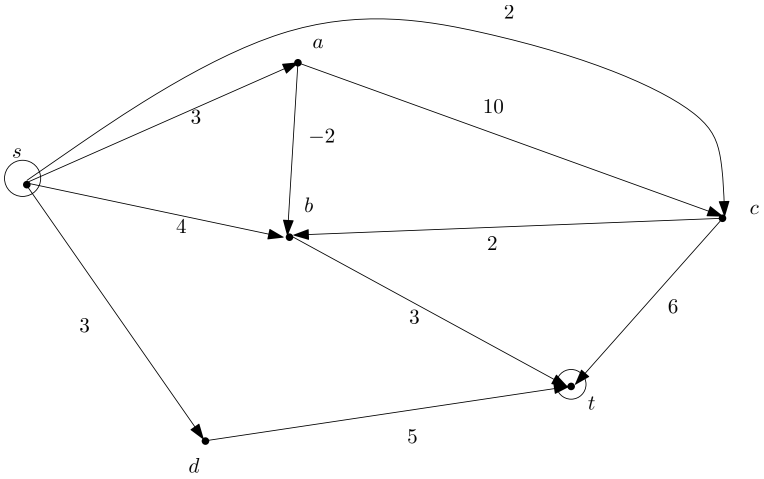
Shortest path complexity with positive cost

Theorem

Let $D = (V, A)$ be a directed graph, $s \in V$ and a cost function $c : A \to \mathbb{R}_+$. Shortest path from s to any vertex v can be found in $O(n^2)$.

Note that with specific implementation (e.g. in binary tree of nodes) we can obtain a complexity in $O(n + mloglog(m))$.

Acircuitic graph



Bellman's idea

A part of an optimal path is still optimal.

$\lambda(v) :=$ minimum cost of o-v-path, with $\lambda(v) := \infty$ if such a path doesn't exist.

Bellman's equation

$$\lambda(v) = \min_{(u,v)\in A} \left(\lambda(u) + c(u,v)\right)$$

The shortest path between o and v is the shortest path between o and u (a predecessor of v) adding the arrow (u,v).

Bellman's idea

A part of an optimal path is still optimal.

$\lambda(v) :=$ minimum cost of o-v-path, with $\lambda(v) := \infty$ if such a path doesn't exist.

Bellman's equation

$$\lambda(v) = \min_{(u,v)\in A} \big(\lambda(u) + c(u,v)\big)$$

The shortest path between o and v is the shortest path between o and u (a predecessor of v) adding the arrow (u,v).

Bellman's idea

A part of an optimal path is still optimal.

$\lambda(v) :=$ minimum cost of o-v-path, with $\lambda(v) := \infty$ if such a path doesn't exist.

Bellman's equation

$$\lambda(v) = \min_{(u,v)\in A} \big(\lambda(u) + c(u,v)\big)$$

> *The shortest path between o and v is the shortest path between o and u (a predecessor of v) adding the arrow (u,v).*

Dynamic Programming algorithm

Assume that the graph is connected and without cycle.

```
$\lambda(s) := 0$ ;
$\forall v \neq s, \quad \lambda(v) = +\infty$ ;

while $\exists v \in V, \quad \lambda(u) = \infty$ do
    choose a vertex $v$ such that all predecessors $u$ have a finite
     label ;
    $\lambda(v) := \min\{\lambda(u) + c(u,v) | (u,v) \in E\}$;
```

Algorithm 2: Bellman Forward algorithm

Graphe without cycle $\Longrightarrow$ there exists a vertex v such that we already visited all predecessors (topological order).

Algorithm

Theorem

Let $D = (V, A)$ be a directed graph without cycle, and $w : A \to \mathbb{R}$ a cost function. The shortest path from o to any vertex $v \in V$ can be computed in $O(m)$.

Note that we do not require the costs to be positive for the Bellman-Ford algorithm. In particular we can also compute the longest path.