ENPC - Operations Research and Transport - 2017

You have 2 hours for the exam. Exercises are independent. Computer, phones, tablets and every connected objects are forbidden. Every note is allowed.

Exercice 1 (2pts)**.** *Consider a game where rewards (to be maximized) are given by the following table where actions of player* 1 *correspond to the lines, actions of player* 2 *to the columns, rewards being given in the order of player. For example, if player* 1 *play* a*, and player* 2 *play* c*, then player* 1 *gains* 0 *and player* 2 *gains* 1*.*

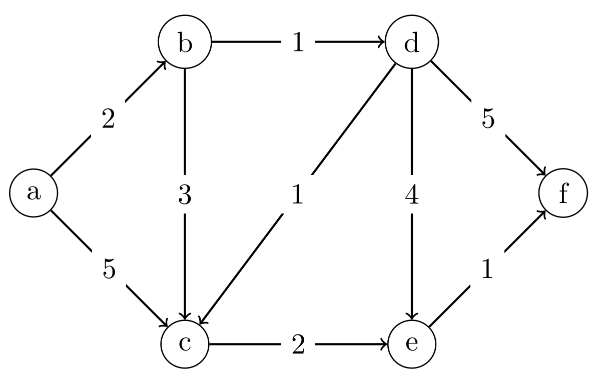
	a	b	c	d
a	(1,-1)	(0,0)	(0,1)	(-1,4)
b	(-1,2)	(2,3)	(3,2)	(-2,3)

1. *Find the Nash equilibrium(s)*
2. *Find the social optimum(s)*
3. *Find the Pareto optimum(s)*

Solution. 1. *(0.75pt) NE :* (b, b) *and* (a, d)

2. *(0.5pt) SO :* (b, b) *and* (b, c)
3. *(0.75pt) Pareto :* (b, b), (b, c), (a, d).

Exercice 2 (5pts)**.** *Consider the following weighted graph.*



1. *Use Dijkstra's algorithm to find the cost of the shortest path between node* a *and node* f*. The results can be presented in a table of the labels where each column corresponds to a node of the graph, and each line to an iteration of the Dijkstra algorithm.*
2. *Find a topological ordering for the graph. Use the topological ordering to compute the cost of the shortest path from* a *to every nodes by Dynamic Programming.*

3. *Give the shortest path from a to f.*

Solution. 1. *We have (2 pt)*

a	b	c	d	e	f
(0)	∞	∞	∞	∞	∞
0	(2)	(5)	∞	∞	∞
0	2	(5)	(3)	∞	∞
0	2	(4)	3	(7)	(8)
0	2	4	3	(6)	(8)
0	2	4	3	6	(7)
0	2	4	3	6	7

Hence the shortest path from a to f as cost 7*.*

2. *(1pt) topological ordering : a-b-d-c-e-f. By dynamic programming we have: (1pt)*
   (a) $\lambda(b) = 2$
   (b) $\lambda(d) = 2 + 1 = 3$
   (c) $\lambda(c) = \min\{3 + 2, 3 + 1\} = 4$
   (d) $\lambda(e) = \min\{4 + 2, 3 + 4\} = 6$
   (e) $\lambda(f) = \min\{6 + 1, 3 + 5\} = 7$
3. *(1pt) Shortest path : a-b-d-c-e-f.*

Exercice 3 (6pts)**.** *Consider a (finite) directed, strongly connected, graph $G = (V, A)$. We consider K origin-destination vertex pair $\{o^k, d^k\}_{k\in[\![1,K]\!]}$, such that there exists at least one path from o^k to d^k. Let denote by*

1. *r^k the intensity of the flow of users entering in o^k and exiting in d^k;*
2. *$\mathcal{P}_k$ the set of all simple (i.e. without cycle) paths from o^k to d^k, and by $\mathcal{P} = \bigcup_{k=1}^{K} \mathcal{P}_k$;*
3. *f_p the number of users taking path $p \in \mathcal{P}$ per hour (intensity);*
4. *$f = \{f_p\}_{p\in\mathcal{P}}$ the vector of path intensity;*
5. *$x_a = \sum_{p\ni a} f_p$ the flux of user taking the arc $a \in A$;*
6. *$x = \{x_a\}_{a\in A}$ the vector of arc intensity;*
7. *$\ell_a : \mathbb{R} \to \mathbb{R}^+$ the cost incurred by a given user to take arc a, if the arc-intensity is x_a;*
8. *$L_a(x_a) := \int_0^{x_a} \ell_a(u)du$.*

We want to find bounds on the price of anarchy, assuming that, for each arc a, $\ell_a : \mathbb{R}^+ \to \mathbb{R}^+$ is non-decreasing, and that we have

$$x\ell_a(x) \leq \gamma L_a(x), \qquad \forall x \in \mathbb{R}^+$$

1. *Recall which optimization problems solves the social optimum x^{SO} and the user equilibrium x^{UE}. We will denote $W(x)$ the objectif function of the user equilibrium problem and $C(x)$ the objective function of the social optimum problem.*

2. *Let x be a feasable vector of arc-intensity. Show that $W(x) \leq C(x) \leq \gamma W(x)$.*

3. *Show that the price of anarchy $C(x^{UE})/C(x^{SO})$ is lower than γ.*

4. *If the cost per arc ℓ_a are polynomial of order at most p with non-negative coefficient, find a bound on the price of anarchy. Is this bound sharp ?*

Solution. 1. *(0.5pts) The user equilibrium and social optimum problem are of the following form*

$$\min_{x,f} \quad J(x) \tag{1}$$

$$s.t. \quad r_k = \sum_{p \in \mathcal{P}_k} f_p \qquad k \in [\![1, K]\!] \tag{2}$$

$$x_a = \sum_{p \ni a} f_p \qquad a \in A \tag{3}$$

$$f_p \geq 0 \qquad p \in \mathcal{P} \tag{4}$$

where $J(x) = W(x) = \sum_{a \in A} L_a(x_a)$ for the user equilibrium, and $J(x) = C(x) = \sum_{a \in A} x_a \ell_a(x_a)$ for the social optimum.

2. *(2pts) As ℓ_a is non-decreasing and $x_a \geq 0$ we have $x_a \ell_a(x_a) \geq \int_0^{x_a} \ell_a(x_a) = L_a(x_a)$. Furthermore, by assumption we have $x\ell_a(x) \leq \gamma L_a(x)$. Summing over $a \in A$ gives the result.*

3. *(2pts) We have*
$$C(x^{UE}) \leq \gamma W(x^{UE}) \leq \gamma W(x^{SO}) \leq \gamma C(x^{SO}).$$

4. *(1.5pts) If ℓ_a is a polynomial function of order at most p with non-negative coefficient, then we have $x\ell_a(x) \leq (p+1)L_a(x)$. Hence, the price of anarchy is lower than $p+1$. For $p = 1$ we have affine function in which case the price of anarchy is at most $4/3 < 2$, so the bound is not sharp.*

Exercice 4 (7pts). *Consider a set of tasks $T = \{T_i\}_{i \in [\![1,n]\!]}$. Each task t_i is a triplet $t_i = (\tau_i, l_i, r_i)$, where $\tau_i \in \mathbb{R}^+$ is the date at which the task should be started, $l_i \in \mathbb{R}^+$ is the length of task i and $r_i \in \mathbb{R}^+$ is the revenue obtained by doing task i. Assume that each task can be done by any single employee.*

1. *Construct a graph with nodes $T \cup \{o, d\}$, where o and d are artificial nodes such that any $o - d$ path is a feasible set of tasks for one employee.*

2. *Show that if you have only one employee, finding the set of tasks that maximizes your revenue is a shortest path problem (don't forget to specify the cost associated to each arc).*

3. *What shortest path algorithm would be adapted here ? Why ?*

4. *Write a Linear Programming problem with binary (with value in $\{0, 1\}$) variables that aims at maximizing revenue with $N \in \mathbb{N}$ employee (a task can be done only once).*

5. *Explain why this problem can be solved with the simplex algorithm.*

6. *Now assume that you want to treat all tasks with the minimum number of employees. Write this problem as a linear problem with binary and continuous variables.*

Solution. 1. *(1pts) For all pair of tasks (t_i, t_j) add an arc to the graph iff $\tau_i + l_i \leq \tau_j$.*

2. *(1pts) Attach to each arc (t_i, t_j), and (t_i, d) the cost $-r_i \leq 0$. Finding the shortest path is finding the best set of task for one employee.*

3. *(1pts) The graph is acyclic, then Dynamic Programming can be applied. (Note : with negative cost Djikstra can not be applied directly).*

4. *(2pts) We denote by x_a the binary variable that has 1 for value if the arc a is selected.*

$$
\begin{aligned}
\max_{\{x_a\}_{a\in A}} \quad & \sum_{(i,j)\in A} x_{(i,j)} r_i & \\
s.t. \quad & \sum_{a\in\delta^-(i)} x_a = \sum_{a\in\delta^+(i)} x_a & \forall i \in [\![1, n]\!] \\
& \sum_{a\in\delta^+(i)} x_a \leq 1 & \forall i \in [\![1, n]\!] \\
& \sum_{a\in\delta^+(o)} x_a \leq N & \\
& x_a \in \{0, 1\} & \forall a \in A
\end{aligned}
$$

5. *(1pts) The constraint matrix being a flow matrix is TU, hence the continuous relaxation has an optimal integer solution found by the simplex algorithm.*

6. *(1pts) We have*

$$
\begin{aligned}
\min_{t,\{x_a\}_{a\in A}} \quad & t & \\
s.t. \quad & \sum_{a\in\delta^-(i)} x_a = \sum_{a\in\delta^+(i)} x_a & \forall i \in [\![1, n]\!] \\
& \sum_{a\in\delta^+(i)} x_a = 1 & \forall i \in [\![1, n]\!] \\
& \sum_{a\in\delta^+(o)} x_a \leq t & \\
& x_a \in \{0, 1\} & \forall a \in A
\end{aligned}
$$

Exercice 5 (bonus)**.** *Give three ideas that allow to speed-up the shortest path algorithm in Google Maps.*