Algorithmic Game Theory

Autumn 2021
Exercise Set 3

Elisabetta Fedele

Contributors
Elisabetta Fedele, Massimo Albarello, Julian Gleissner, Ravi Srinivasan

Exercise 1

Part 1

Let s be a pure Nash equilibrium and s^* be an optimum solution which maximizes the social welfare. Then:

$$\begin{aligned} SW(s) &= \sum_i u_i(s) \quad \text{(definition of social welfare)} \\ &\geq \sum_i u(s_i^*, s_{-i}) \quad \text{(as } s \text{ is a PNE)} \\ &\geq \lambda SW(s^*) - \mu SW(s) \quad \text{(by smoothness)} \end{aligned}$$

and by rearranging the terms we get

$$\frac{SW(s^*)}{SW(s)} \leq \frac{(1+\mu)}{\lambda}$$

for any pure Nash equilibrium s and any social optimum s^*. That is, $PoA_{PNE} \leq \frac{(1+\mu)}{\lambda}$.

Part 2

In order to prove that $PoA \leq n$ it is sufficient to use the theorem of the exercise 1.1 with $\mu = 0$ and $\lambda = \frac{1}{n}$. Thus, if we are able to demonstrate that a technology game is $\left(\frac{1}{n}, 0\right)$-smooth, then our initial thesis is proved.
Let α_i be the quality parameter of the resource chosen by the player i. Since every player has to choose exactly one resource, his utility function when he switch from a strategy s to another strategy S^* is at least the quality parameter of the technology he has chosen in the strategy $S*$, as he were the only player to choose that technology.
Hence, if we call R the set of all the resources that are used by the players in the strategy S*, we have

$$\sum_i u(s_i^*, s_{-i}) \geq \sum_i \alpha_i = \sum_{r \in R} n_r(s^*) \alpha_r$$

Now we multiply the nominator and the denominator of every term of the summation for $n_r(s^*)$ and, since $n_r(s^*) \leq n$ is always true, we obtain:

$$\sum_{r \in R} n_r(s^*) \alpha_r = \sum_{r \in R} \frac{n_r(s^*)^2 \alpha_r}{n_r(s^*)} \geq \sum_{r \in R} \frac{n_r(s^*)^2 \alpha_r}{n} = \frac{\sum_{r \in R} n_r(s^*)^2 \alpha_r}{n}$$

Further, in a technology game we can define $SW(S^*) = \sum_{r \in R} n_r(s^*)^2 \alpha_r$.
Hence, rearranging the terms, we get that

$$\sum_i u(s_i^*, s_{-i}) \geq \frac{SW(S^*)}{n}$$

So technology games, by definition of smoothness, are $\left(\frac{1}{n}, 0\right)$-smooth and by the theorem of the exercise 1.1, we get that $PoA_{PNE} \leq n$.

Part 3

In order to have $PoA_{PNE} = n$ it is sufficient to have a technology game in which there is an $\alpha' = \alpha_{max}/n$, where α_{max} is the maximum α among all the quality parameters of the game. Let us show an example for $n = 2$ and $\alpha_{max} = 2$.
In particular there are two players and two different technologies available where the first one has a quality parameter $\alpha_1 = 1$ and the second one has a quality parameter $\alpha_2 = 2$.
Let u_i be the utility function of the technology i and n_i the number of players using that technology. Then, from the definition of technology game, we know that:

- $u_1 = n_1$
- $u_2 = 2 * n_2$

In this particular game we note by inspection the optimal utility of 8, realized when both the players use the second technology.
We can also notice that the PNE with the minimum cost is realized when both the players choose the first technology and thus they achieve a social welfare of 4.
Thus, from the definition of PoA_{PNE} we obtain that:

$$PoA_{PNE} = \frac{max_{s \in S} SW(s)}{min_{s \in PNE} SW(s)} = \frac{8}{4} = 2$$

As 2 is the number of players of that game, we have proved that exists at least one case in which $PoA \geq n$ and, thus, the limit of the exercise 1.2 is tight.

Exercise 2

We will take into consideration a game with two players and two possible strategy, that can be easily visualized through the following utility matrix

		x	$1-x$
		s_1	s_2
y	s_1	a, b	c, d
$1-y$	s_2	e, f	g, h

In order to prove that there is always a mixed Nash equilibrium, we need to find a probability distribution over the strategies of the two players such no player has a reason to switch to another strategy. Let x be the probability that the column player chooses to adopt the strategy s_1 and y the probability that the row player chooses the strategy s_1. Let us assume that 1-x and 1-y are respectively the probability that the row and the column player chooses the strategy s_2.
First, we will compute the values that should be assumed by x in order to guarantee that the row player has no reason to switch strategy.
We mainly distinguish three different cases.

Case 1: $a > e \land c > g$

In this case the row player will always maximize his/her utility by choosing the strategy s_1, regardless of the choice taken by the column player. Consequently, x can be any probability distribution in this case.

Case 2: $a < e \land c < g$

In this case the row player will always maximize his/her utility by choosing the strategy S_2, regardless of the choice taken by the column player. Consequently, x can be any probability distribution also in this case.

Case 3: $(a \geq e \land c \leq g) \lor (a \leq e \land c \geq g)$

In this case the choice of the row player depends on the probability distribution of the choice of the other one. In this situation we can compute the values of x which guarantee a mixed equilibrium solving the following equation, which make equals the expected values of the utilities in both the strategies.
In particular we have that:

$$ax + c(1-x) = ex + g(1-x)$$

If $a = e$

$$x = \frac{g-c}{g-c+a-e} \tag{1}$$

Furthermore, we can demonstrate that the one assumed by x are values of a probability function, considering the two following situations included in case 3:

Case 3a: $(a \geq e \land c \leq g)$

The numerator and the denominator of (1) are always positive, so x is always greater than 0. Furthermore, as $a - e > 0$ the denominator is always greater than the nominator, thus x is always smaller than 1.

Case 3b: $(a < e \land c > g)$

Symmetrically, the numerator and the denominator of (1) are always both negative, so x is always greater than 0. Furthermore, as $a - e > 0$ the denominator is always greater than the nominator, thus x is always smaller than 1.
It must be noticed that in the case in which we have $a = e \land c = g$ the first equation obtained would

have infinite solutions, thus we will take any value of the range $[0, 1]$.

Symmetrically, we can compute the probability y which, once fixed, will guarantee that exists a strategy for the column player such that he has no reason to switch strategy.
In particular, as we noticed for the previous player, if $b > d \wedge f > h$ the column player choice will not depend on the row player's one, since he would will always maximize his/her utility by choosing the strategy s_1 (if $b > d \wedge f > h$) or s_2 (if $b < d \wedge f < h$).
As we have already said before if $(b > d \wedge f < h) \vee (b < d \wedge f > h)$ we need to compute the values of y which guarantees a mixed equilibrium, since the choice of the column player will no more be independent from the choice taken by the row player. In particular we get

$$y = \frac{h - f}{h - f + b - d}$$

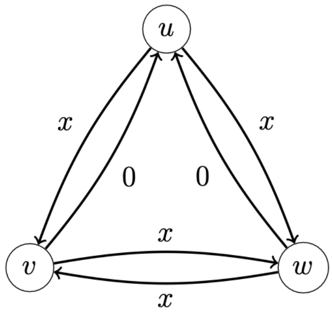
Thus, since it does exist a probability distribution over the strategies of the two players such no player has a reason to switch to another strategy, this game has always a mixed Nash equilibrium.

Exercise 3

We inspire our solution on the congestion game given in the lecture notes of week 2, in the Theorem 7. We consider a game with 4 players, who have a source and a sink each. The source and sink for each player are presented in the table below. Further, we consider the (asymmetric) network presented in the figure below. We consider the delay function $x(0) = 0, x(1) = 1, x(2) = M$. Hence, in general the delay function takes the form

$$x(k) = M_r \cdot k - (M_r - 1)$$

where k represents the number of players who are using the edge.



We note by inspection the optimal cost is 4. Now, we want to prove that the strategy summarized in table below is the pure Nash equilibrium with the maximum cost. We do this by considering the scenario of each player individually. We note that for each source and sink, there are two paths one could take to get from the source to the sink. One of them is through the direct edge connecting the source and the sink (this edge exists for any source and any sink), or via the path going through the third node which is not the source nor the sink. So we can call for each player the strategy of taking the "direct path" s_1 and the strategy of taking the path visiting the third node strategy s_2. The strategies of the four players are summed up in the table below, but we will analyze their choices one by one to make everything clearer.

player	source	sink	strategy in OPT	cost in OPT	strategy in PNE	cost in PNE
1	u	v	u$\rightarrow v$	1	u$\rightarrow w \rightarrow v$	M
2	u	w	u$\rightarrow w$	1	u$\rightarrow v \rightarrow w$	M
3	v	w	v$\rightarrow w$	1	v$\rightarrow u \rightarrow w$	M
4	w	v	w$\rightarrow v$	1	w$\rightarrow u \rightarrow v$	M

Player 1

We note that player 1 in the strategy s_2 uses the path (u,w,v), which has a cost of M. Consider switching to strategy s_1, i.e. using path (u,v). We note that that edge is already occupied by players 2 and 3, hence the cost for player 1 would then become 2M instead of M, hence she will stick with strategy s_2.

Player 2

We note that player 2 in the strategy s_2 uses the path (u,v,w), which has a cost of M. Consider switching to strategy s_1, i.e. using the path (u,w). We note that player 1 and 3 already are using this edge, hence the cost for player 2 would then become 2M instead of M, so player 2 will stick with strategy s_2.

Player 3

We note that now player 3 in the strategy s_2 uses the path (v,u,w), which has a cost of M. Consider player 3 switching to path (v,w). We note that player 2 is already using this edge, so the cost if player 3 would switch to this path would still be M, so she has no incentive to switch.

Player 4

We note that now player 4 in the strategy s_2 uses the path (w,u,v) of cost M. Let's consider player 4 switching to strategy s_1, i.e. using path (w, v). We noted that player 1 is already using this edge, hence the cost when using strategy s_1 would still be M, hence player 4 has no incentive to switch strategy. Hence, if every player plays the strategy s_2 we have a pure Nash equilibrium. We note that the total cost of the optimal strategy is 4, and the cost of the non optimal PNE is 4M. Hence, using the definition of price of anarchy, we get that:

$$PoA_{PNE} = \frac{\max_{s \in PNE} cost(s)}{\min_{s \in S} cost(s)} \geq \frac{4M}{4} = M$$

We note that for $M \geq \frac{5}{2}$ this proves the claim.

Exercise 4

A problem π^* in PLS is called PLS-complete if, for every problem π in PLS, it holds $\pi \leq_{PLS} \pi^*$.
In this case, π^* is the problem of computing a PNE in congestion games with affine delay functions of the form: $d_r(x) = a_r x + b_r$ with $a_r, b_r \geq 0$.

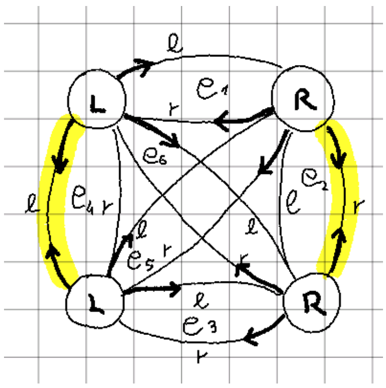
We know that Max-Cut is PLS-complete: $\pi \leq_{PLS}$ Max-Cut $\forall \pi \in PLS$. Therefore, if we show that Max-Cut $\leq_{PLS} \pi^*$, then it must hold that $\pi \leq_{PLS} \pi^*, \forall \pi \in PLS$, and so that π^* is PLS-complete.

To do that we inspire our solution on the proof of the Theorem 19, which demonstrate that the problem of computinng a PNE in a generic congestion game is PLS-complete.
In order to compute the reduction wanted in this particular situation, we have to map instances of Max-Cut to congestion games with affine delay functions. Given a graph $G = (V, E)$ with edge weights $w : E \to N$, players correspond to vertices V and for each edge $e \in E$ we add two resources r_e^{left} and r_e^{right}.

In case of affine delay function of the form: $d_r(x) = a_r x + b_r$ with $a_r, b_r \geq 0$, we can define the delay as:

$$d_r(k) = \begin{cases} 0, & \text{if } k = 0 \\ w_e, & \text{if } k = 1 \\ 2w_e & \text{if } k = 2 \end{cases}$$

Consider a clique of four nodes in which a cut is performed in order to split the two left nodes from the other two right nodes. The associated congestion game with affine delay function defined as above is shown in the following picture.



In this game, each player has two strategies, either to choose all the "left" resources (labeled by l in the picture) or all the "right" resources (labeled by r) for its incident edges. This way, cuts in the graph are in one-to-one correspondence to strategy profiles of the game. We can see this by noting that the cut W has a weight of: $W = w_1 + w_3 + w_5 + w_6$ While Rosenthal potential is a function of this cut W:

$$\Phi(s) = \sum_{r=1}^{m} \sum_{k=1}^{n_r(s)} d_r(k) =$$

$$= (w_{1,l}) + (w_{1,r}) + (0) + (w_{2,r} + 2w_{2,r}) + (w_{3,l}) + (w_{3,r}) + (w_{4,l} + w_{4,l}) + (0) + (w_{5,l}) + (w_{5,r}) + (w_{6,l}) + (w_{6,r})$$

As $w_{e,l} = w_{e,r} = w_e$, we get that:

$$\Phi(s) = 2w_1 + 3w_2 + 2w_3 + 3w_4 + 2w_5 + 2w_6 =$$
$$= 2(w_1 + w_2 + w_3 + w_4 + w_5 + w_6) + w_2 + w_4 =$$
$$= 2W_{tot} + (W_{tot} - W) =$$
$$= 3W_{tot} - W$$

Therefore, a cut of weight W is mapped to a strategy profile of Rosenthal potential: $\Phi(s) = 3W_{tot} - W$, and vice versa. Consequently, the local maximum of Max-Cut corresponds to a local minimum of the Rosenthal potential, which are exactly the PNE.

As we have shown that: Max-Cut $\leq_{PLS}$ (PNE in congestion games of affine delay functions) and as we know that Max-Cut is PLS-complete, it follows that computing a PNE in congestion games of affine delay functions is PLS-complete.