ETH Zürich
Andrei Ivanov,
Lorenzo Laneve,
Varun Maram,
Paolo Penna

October 29, 2021

Deadline: Beginning of next lecture

Algorithmic Game Theory

Autumn 2021

Exercise Set 6

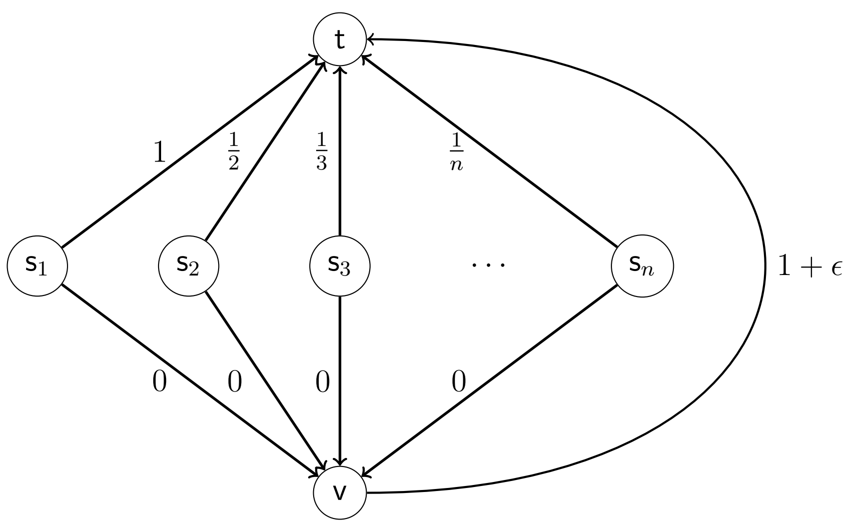
Your solutions to this exercise sheet will be **graded**. Together with the other three graded exercise sheets, it will account for 30% of your final grade for the course.

You are expected to solve the exercises carefully and then write a nice complete exposition of your solution (preferably using **LaTeX** or similar computer editors – the appearance of your solution will also be part of the grade). You are welcome to discuss the tasks with your fellow students, but we expect each of you to hand in **your own** individual write-up. Your write-up should list all collaborators.

Please submit your solutions via moodle before the beginning of next lecture (**November 5, 10:00 am**). If you cannot use moodle, please send solutions by **email** to agt-course@lists.inf.ethz.ch.

Exercise 1: (3 Points)

Consider the following **cost-sharing** game with n players. Each player i has source node s_i and destination node t.



A player has two possible strategies: Either take the direct edge or take the detour via v. Recall that the cost of each edge is *shared equally* among the players that use that edge.

Your task:

- In general we have the following inequalities between Price of Stability for a certain class of equilibria:

$$PoS_{\mathsf{CCE}} \leq PoS_{\mathsf{MNE}} \leq PoS_{\mathsf{PNE}} \ .$$

Show that for the specific game in the figure above, all inequalities are actually equalities, that is,

$$PoS_{\mathsf{CCE}} = PoS_{\mathsf{MNE}} = PoS_{\mathsf{PNE}} \ .$$

Exercise 2: (2+2+2 Points)

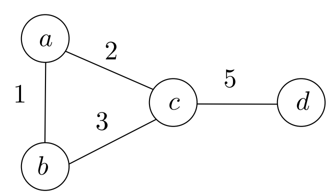
Consider the following scenario: we are given a graph and a set of "feasible trees"

$$\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_M\}$$

spanning the nodes of the graph. We are **only allowed to choose a tree in $\mathcal{T}$** and we would like the one of minimal total weight (sum of edge costs).

Your task:

1. Describe a truthful mechanism (algorithm and payments) for this problem.
2. Prove that your mechanism is truthful using one of the techniques/results seen in the lectures.
3. Suppose $\mathcal{T}$ consists of all possible spanning trees of your graph. Explain how your mechanism can guarantee voluntary participation for each player whose removal does **not disconnect** the graph. Moreover, run your mechanism on this example:



and compute the payments for each player (the numbers are the reported costs).

Note: In this exercise, each edge corresponds to a different player, the cost t_e of edge e is private and player e can report a different cost c_e to the mechanism. If the chosen tree contains edge e then the cost for player e is t_e, otherwise it is equal 0. The utility of each player is the received payment minus the cost.

Exercise 3: (3+2 Points)

Consider the mechanism design problem of scheduling jobs on **selfish related machines** in the lecture notes (Lecture 6):

> *We have k jobs of size $J_1, J_2, \ldots, J_k$ and a set of n machines (players). Each machine i has a type t_i and an allocation a of jobs to machines specifies the amount of work $w_i(a)$ which is allocated to machine i (the sum of all jobs weights that a puts on machine i). The parameter t_i is the cost (time) required by machine i to process one unit of work, and therefore the cost of player i for allocation a is $w_i(a) \cdot t_i$. Each player i can report a possibly different cost c_i to the mechanism.*

Your task:

1. Prove that there exists a **truthful** mechanism for the problem of scheduling selfish related machines which minimizes the **makespan** or **maximum** cost.
   (Prove Theorem 5 in Lecture 6).

2. For two machines, consider the following simple **greedy** algorithm.

   - Process jobs one by one in decreasing order of size, and assign the k^{th} job to the machine which is the "most favorable for this job" up to this point. That is, given the allocation a^{k-1} of the first $k-1$ jobs, do the following:

     IF $c_1 \cdot (w_1(a^{k-1}) + J_k) \leq c_2 \cdot (w_2(a^{k-1}) + J_k)$ THEN
     allocate J_k on machine 1
     ELSE allocate J_k on machine 2

   Show that there is no truthful mechanism (**greedy**, P), that is, no matter how we define the payments P, the resulting mechanism cannot be truthful for this problem.

Note: You can use any theorem in lecture notes black box (without reproving it).