CS 6840: Algorithmic Game Theory **Spring 2017**

Problem Set 1

Instructor: Éva Tardos **due Wednesday, February 15**

There are 5 questions on this problem set of varying difficulty. For full credit you should solve 4 of the 5 problems. Solving all 5 correctly results in extra credit. A full solution for each problem includes proving that your answer is correct. If you cannot solve a problem, write down how far you got, and why are you stuck.

You may work in pairs and hand in a shared homework with both of your names marked. You may discuss homework questions with other students, but closely collaborate only with your partner. Send me or Thodoris email if you have trouble finding a partner, and we'll see if we can help. You may use any fact we proved in class or is proven in the Roughgarden book without providing the proof or reference, and may read the relevant chapters of the other reference book. However, **you may not use other published papers, or the Web to find your answer**.

You need to type your solution, and submit on CMS in pdf format (only). If you have a partner, use CMS to mark the two of you as partners, and only upload one copy of the solution. You may add hand drawn figures to the file, but all text needs to be typed. If your solution is complex, say more than about half a page, please include a 3-line summary to help us understand the argument.

Please use Piazza for questions, so everyone can benefit from the answers. Great if you can answer other student's questions on Piazza. Office hours are posted on the course web page `http://www.cs.cornell.edu/courses/cs6840/2017sp/`

(1) [Roughgarden problem 11.2] In class we evaluated the cost of a solution in the routing game by summing over all players. An alternate way to evaluate a solution is by the maximum cost of any path. So for example, in the Pigou example with two links with cost functions 1 and x (or also with 1 and x^k), the Nash equilibrium solution of cost 1 is actually also optimal under this objective function: there is no solution where all players use paths of cost less than 1. This objective function can be a bit weird if the game is multiple source-sink pairs (say, if one pair is close and others are far apart). For this problem, consider this "minimize the cost of the maximum cost path used" objective function in the case of a single source and single sink, so all players want to go from the same source s to the same destination t.

- **(a)** Show that a Nash equilibrium flow is not always guaranteed to be optimal for the objective of minimizing the maximum path used. Show that the Price of Anarchy with respect to this objective function can be as large as 4/3 even with linear cost functions on the edges ($ax + b$ as costs).
- **(a)** Show that the price of anarchy with respect to this maximum cost path objective function for the case of single-source and single sink networks cannot be worse than the price of anarchy for the case of sum of costs objective considered in class (e.g., it is at most 4/3 for linear cost functions).

(2) Consider the following load balancing game: there are n jobs, each controlled by a separate and selfish user. There are m identical servers S that can serve jobs. Each job i has a weight w_i and selects a server to be assigned to. The load of the server is the sum of the weights of the jobs assigned to it. Each agent experiences cost equal to the load of the server it is assigned to. Their goal is therefore to choose a server with minimum load. With all jobs presenting different weights, it doesn't seem so natural to add up the costs of all players, instead we will consider the objective of minimizing the maximum load of a server (the makespan in scheduling terminology).

(a) Show that the price of anarchy for the makespan for pure strategy Nash equilibria (solutions where players do not randomize in selecting jobs) is at most 2.
(b) Give a class of examples to show that the price of anarchy can be arbitrarily close to 2.

(3) Consider the special case of this game where each job j is of weight $w_j = 1$, so the load on a server is the number of jobs it serves. However, assume that each job j has an associated set $S_j \subseteq S$ of servers where it can possibly be served. Further, each server i has a load dependent response time: $r_i(x)$ is the response time of server i if its load is x. We assume that $r_i(x)$ is a monotone increasing function for all i. You may also assume that $r_i(x)$ is convex if that helps. Please note in your answer what assumption you are using and where (r monotone or convex?).

This can be thought of as an atomic routing game: add a new terminal t and connect all servers i to the terminal with an edge $e = (i, t)$ of cost $r_i(x)$, and connect all jobs j to the servers $i \in S_j$ with an edge of cost 0. We will see we in lecture on Friday, Feb 3, that all atomic routing games (in fact all congestion games) are potential games (see also Roughgarden, Lecture 13). This implies that Nash equilibria are local optima of the corresponding potential function Φ.

Hint: For this problem, you may use the fact that the minimum cost matching problem can be solved in polynomial time. This may be useful as a subroutine. **The minimum cost matching problem** is given by a bipartite graph G, costs on the edges and an integer k, and the problem is to find a matching in G of size k of minimum possible cost. You can see any book on algorithms or combinatorial optimization for algorithms for this problem.

(a) Give a polynomial time algorithm to find an equilibrium.
(b) Consider the assignment of jobs to servers that minimizes social welfare: the sum of all response times **over jobs** (or average response time), and give a polynomial time algorithm to find the best assignment for this objective function.
(c) Consider the special case when all jobs can be processed on any server. Are all pure Nash equilibria socially optimal in this case?
(d) A mixed Nash equilibrium is a probability distribution for all players on strategies, so that no player can improve his or her expected cost by changing strategy. In this part consider the special case when all jobs can be processed on any server and all servers have identical delay functions $r_i(x) = x$. An example of a mixed Nash equilibrium is when all players choose between the servers uniformly at random. In such a mixed Nash all players' expected cost is $1 + \frac{n-1}{m}$, and this example with $n = m$ shows that the price of anarchy for mixed Nash equilibria is at least $1 + \frac{(n-1)}{m}$. (See example 17.4 in the Algorithmic Game Theory book.) Show that this example is worst possible, and the price of anarchy is at most $1 + \frac{(n-1)}{m}$, that is, the sum of the expected cost of the players in a mixed Nash equilibrium is at most $1 + \frac{(n-1)}{m}$ times the minimum possible such cost.

(4) Consider a variant of the previous question where we assume each job j has a different weight w_j, and the load on a server is the sum of the weights over the jobs assigned to this particular server. This is no longer a congestion game in the sense we defined in class, as congestion is no longer a function of just the number of jobs.

(a) Show that there is no potential Φ function with the property that for any pure strategy, and any single player changing strategy, the change in the potential function is same as the change in the players's cost who is changing strategy.
(b) Show that this game has a pure Nash equilibrium. Is the equilibrium necessarily unique?
(c) Is a sequence of best responses guaranteed to find a Nash equilibrium, or can a best response sequence cycle?

(5) Consider the following grouping game. Players are nodes of a graph, and edges represent social relations. We assume that edge (i, j) connecting nodes i and j has an associated value $u_{ij} \geq 0$. We will think of the value u_{ij} as the strength of their friendship, the benefit players i and j both incur if nodes i and j are together. In the game each player can choose between affiliating with one of two social groups A or another one B. The result is a partition of the nodes into two sets. The utility of a player $i \in A$ is $U_i(A, B) = \sum_{j \in A} u_{ij}$, and if player i is in B its utility is $U_i(A, B) = \sum_{k \in B} u_{ik}$, i.e., the sum of the utilities for partners in the same group. Assume that the values are symmetric, namely that $u_{ij} = u_{ji}$ for all i and j.

- **(a)** Show that this is a potential game where the social welfare $\frac{1}{2} \sum_i U_i(A, B)$ is a potential function.
- **(b)** Show that any (pure strategy) Nash equilibrium has social welfare at least $1/2$ of the maximum possible. (This is a utility game, not a cost game, so a suboptimal solution has less utility.)
- **(c)** It turns out that all potential games are congestion games with the right definition of congestion. Define a congestion game that is equivalent to this partition game.
- **(d)** How does the game change if we do not assume symmetry? that is if u_{ij} may not equal u_{ji}? Is the resulting game still a potential game? Does it have a bound of 2 on the price of anarchy? Is it still true that any sequence of best responses converges to a Nash equilibrium?