

We will maintain a FAQ for the problem set on the course Web page. You may use any fact we proved in class without proving the proof or reference. However, you may not use published papers.

You are expected to attempt all problems. If you cannot solve a problem, write down how far you got, and why are you stuck.

Cooperation in developing answers is encouraged. However, each student must write down all answers separately.

(1) Consider the **atomic** load balancing game from the early lectures: there are n jobs each controlled by a separate and selfish user. There are m servers S that can serve jobs, and each job j has an associated set $S_j \subseteq S$ of servers where it can possibly be served. For this problem we assume that the load of each jobs is 1, and 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 is that helps. We showed in class that this is an (atomic) potential game, that is the Nash equilibria of this game are the local optima for the objective function Φ used on Monday, September 5th. **Hint:** You may use the fact that the minimum cost matching problem (defined below) can be solved in polynomial time. This may be useful as a subroutine.

- (a) Give a polynomial time algorithm to find an equilibrium.
- (b) We considered two possible definitions of social optimum for this game. First consider the assignment of jobs to servers that minimizes the maximum response time, and give a polynomial time algorithm to find the best assignment for this objective function.
- (c) Next considered the assignment of jobs to servers that minimizes the sum of all response times (or average response time), and give a polynomial time algorithm to find the best assignment for this objective function.

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.

(2) Consider the game from the previous problem in the special case that the response time is directly proportional to the load, that is $r_i(x) = x$ for all i , so the goal of the users is to be on servers with small load. In this problem we consider the ratio of the worst possible Nash equilibrium and the optimum under the objective function minimizing the maximum load. (Often referred to as the *min-max* objective.)

Recall from class (or the notes from August 29th) that Nash equilibria in this case can be quite bad. Show that if there are m machines, than the maximum load in a Nash equilibrium is at most an $O(\log m)$ factor above the minimum possible value of the maximum load.

(3) Consider the non-atomic multicommodity flow problem we have been discussing in class (defined on Wednesday, September 7). We defined Nash equilibria as a flow f such that for all pairs of paths P and Q connecting the same pair of terminals, if $f_P > 0$ then $\ell_P(f) \leq \ell_Q(f)$.

A maybe more intuitive definition would be as follows. A flow f is a Nash equilibrium, if the following holds. For any pairs of paths P and Q connecting the same pair of terminals, such that $f_P > 0$ and any $0 < \delta \leq f_P$ if we define an alternate flow \hat{f} by setting

$$\hat{f}_R = \begin{cases} f_P - \delta & \text{if } R = P \\ f_Q + \delta & \text{if } R = Q \\ f_R & \text{otherwise} \end{cases}$$

then $\ell_P(f) \leq \ell_Q(\hat{f})$.

This definition considers the flow f and a very small δ amount of flow on a path P , and wonders if this small amount of flow is happier on path P or should it switch to another path Q . Here we model the flow being non-atomic by allowing arbitrary small amounts of flow to switch, but we do not allow “zero” amount to switch, as it is less clear what that means.

Show that the two definitions are the same.

(4) In class on September 9 we proved the bicriteria bound comparing a Nash flow to an optimal flow. When we want to consider capacitated edges, an edge with capacity u_e can be modeled by a delay function $\ell_e(x) = a_e/(u_e - x)$, where a_e is another constants associated with the edge e (in addition to the capacity u_e . (Note that this delay does model capacity u_e as the delay grows to infinity as the flow approaches the capacity.)

Show that the total delay of a Nash flow in such a network is bounded above by the minimum possible total delay of a flow satisfying the same demands in a network with only 1/2 the capacity, i.e., where each capacity is halved (u_e is replaced by $u_e/2$ for each e without changing a_e).

(5) In the bicriteria bound we proved on September 9 we assumed that all flow is very sensitive to delay, and the flow is in a real Nash equilibrium. It is maybe more reasonable to model a stable state as one of the approximate equilibria, i.e., assume only that for all pairs of paths P and Q connecting the same pair of terminals, if $f_P > 0$ then $\ell_P(f) \leq (1 + \varepsilon)\ell_Q(f)$ for some sensitivity parameter $\varepsilon > 0$. Show that there is a version of the bicriteria bound that holds also for all approximate Nash flows.

(6) Consider the one commodity special case of the nonatomic selfish routing game discussed in class where all traffic goes from a common source s to a common destination t . Assume for this problem that the delay on each edge is a nonnegative, linear and monotone increasing function of the load.

For this problem we define a flow f^* to be optimal if the longest paths that carries flow is as short as possible, and we define a flow to be *fair* if all flow is carried on equal length paths. (This definition assumes that users realize the existence of a better path only by seeing other users who use that path, and the length of path not carrying flow is not relevant for the definition.) We know from class (essentially by definition) that the Nash flow is fair. From the Braess paradox example, we also see that there can be a fair flow that is better than the flow at Nash equilibrium.

- (a) Prove that the Nash flow is at most a factor of 4/3 worse than the optimal for the objective of minimizing the longest path carrying flow.
- (b) For this part consider a flow f^* that minimizes average delay, that is, minimizes $\sum_P f_P d_P(f)$. We know that this optimal flow may not be fair. We measure the unfairness of this flow by

the ratio of the lengths of the longest and shortest (s, t) paths that carries flow. Prove that the unfairness of the flow f^* is at most 2.