Algorithmic Game Theory	**Problem Set 2**
CS 6840 Spring 2012	**Due Wednesday, February 29th**

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 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. You may use any fact we proved in class without proving the proof or reference, and may read the relevant chapters of the book. However, you may **not use other published papers, or the Web to find your answer**.

Solutions can be submitted on CMS in pdf format (only). If you have a partner, write both names on the solution, but only upload or submit it once. In any case, please type your solution or write neatly to make it easier to read. If your solution is complex, say more than about half a page, please include a 3-line summary to help us understand the argument.

We will maintain a FAQ for the problem set on the course Web page.

(1) Recall that a fair cost-sharing game is a congestion game, where each congestible element e has a total cost function $c_e(x)$ is the total cost of e is when x users are using e. We assumed $c_e(x)$ is monotone non-decreasing, and concave. Here we consider when this game is guaranteed to have strong Nash equilibria.

(a) Show that even when strategies congest only a single element each, the Nash minimizing the potential function may not be a strong Nash.

(b) Suppose we make the strong Nash definition weaker than what we used in class, require only that for any possible group deviation there is a player who is no better off after deviation (that is, assume that on equal cost players are not willing to participate in a deviating group). Does the statement in (a) remain true with this version of the definition also?

(b) Recall the prisoner dilemma game from the first lecture (or from Chapter 1). Show that the prisoner dilemma game as no strong Nash.

(d) Show that strategy strong Nash is not guaranteed to exists, not even for 2 player cost-sharing games.

(2) Consider a two player. Assume that there are 2 players, and each player chooses between n pure strategies. Assume that the game is given by the matrices A and B, listing the payoffs for the two players respectively for each $n \times n$ possible plays. This is the matrix that is traditionally called payoff matrix. Assume that the matrix A and B has random entries, say all entries in the range $[0, 1]$ filed out uniformly independently at random. Show that the probability that this random game has a pure (deterministic) Nash equilibrium is at least roughly $1 - 1/e$ if n is large. You may use the fact that for large n we have that $(1 - 1/n)^n \approx 1/e$.

Warning. You may want to compute the probability that a pair of strategies (i, j) forms a Nash. Unfortunately, these events are **not** independent!

(3) An action s_i of a player i is ϵ-dominated by action s_i' if for all strategy profiles s_{-i} of the other players $u_i(s_i, s_{-i}) \leq u_i(s'_i, s_{-i}) - \epsilon$, that is s_i is at least an ϵ worse for i than s'_i no matter what the other payers do. Let s_i be an ϵ-dominated action of a player i.

(a) Show that if a player i uses the weighted majority algorithm discussed in class on Monday, February 20, to choose his/her strategies, that the probability $\pi(s_i)$ that he/she is playing strategy s_i goes to zero over time.
(b) Give an example of a game with a coarse correlated equilibrium, and an ϵ-dominated action of a player i, where player i is playing action s_i with positive probability.
(c) Can this also happen in correlated equiliria? (i.e., can there be a correlated equilibrium when player i plays his/her ϵ-dominated action with positive probability?

(4) Recall the set-up for online regret-minimization from Lecture (Monday, February 20): there is a fixed set A of actions; each day $t = 1, \ldots, T$ you pick an action at $a^t \in A$ (possibly from a probability distribution) based only on information from previous days; and then a cost vector $c^t : A \to [0, 1]$ is unveiled. The goal is to design a (randomized) algorithm that, for every sequence of cost vectors, has small expected average regret. [Recall that the (average, per time-step) regret is the difference between your average cost $\frac{1}{T}\sum_{t=1}^{T} c^t(a^t)$ and the average cost of the best fixed action $\frac{1}{T}\min_{a\in A}\sum_{t=1}^{T} c^t(a)$.]

(a) The most natural algorithm is to pick the strategy each day that seemed best so far, that is at time t pick the strategy a^t that minimizes $\sum_{s=1}^{t-1} c^s(a)$. Show that the average regret of this algorithm can be $\Omega(1)$ as T goes to infinity. How large can you make the ratio between the average cost of the best fixed action, and the average cost of this algorithm?
(b) Lets consider the following randomized pre-processing step: independently for each action a, initialize the starting cumulative cost to a random variable: X_a to the number of coin flips needed until you get heads, assuming that the probability of heads is ϵ, using independent experiments for each action a. Then, every day t, you pick the action that minimizes the perturbed cumulative cost prior to that day: $-X_a + \sum_{s=0}^{t-1} c^s(a)$. We assume that the random variables X_a are independent from the costs c^t. Note that this selection can be implemented if the algorithm has access to the whole vector c^t after each day t.
Prove that, for each day t, with probability at least $1 - \epsilon$, the smallest perturbed cumulative cost, that is $\min_a -X_a + \sum_{s=0}^{t-1} c^s(a)$, is at least 1 less than the second-smallest item in this minimum.
(c) As a thought experiment, consider the (unimplementable) algorithm that, every day, picks the action that minimizes the perturbed cumulative cost $-X_a + \sum_{s=0}^{t} c^s(a)$, taking into account the current days cost vector. Prove that the average regret of this algorithm is at most $\max_a X_a/T$.
(d) Prove that $E[\max_a X_a] = O(\epsilon^{-1}\log n)$, where n is the number of actions.
(e) Use the previous parts to prove that, for a suitable choice of ϵ, the algorithm in (b) has expected average regret $O(\sqrt{\frac{\log n}{T}})$, just like the multiplicative weights algorithm covered in class. (Make any assumptions you want about how ties between actions are broken.)

(5) Hotelling games is a general class of games when k providers for a set of customers. Here we use the following simple case: G is a graph on n vertices. There are k providers, and each provider selects one of the nodes of the game, you can think of the location as a souvenir stand. Once the sellers selected their locations. Each node v in the graph has $n_v > 0$ customers (tourists), and each costumer selects the closest seller. In case of ties divide the n_v costumers uniformly among the closest sellers (OK if the fractions are not integers). The goal of the sellers is to attract as many buyers as possible. Let N_i be the total number of customers who selected seller i. In this game the traditional social welfare is not a good measure, as I assumed all customers choose a seller, and hence $\sum_i N_i = \sum_v n_v = N$. Instead we will look at a fairness measure, $\min_i N_i$.

(a) Show that the price of anarchy for pure Nash equilibria in this game is bounded by 2. By which we mean that if *Opt* denotes the value of the most fair allocation $\max \min_i N_i$, where the maximum is taking over the possible locations of the k sellers, then for any pure strategy Nash equilibria $\min_i N_i > Opt/2$.

(b) In this game the utility of a player i is between $[0, N]$. The weighted majority algorithm assumed utilities are in the range $[0, 1]$. Show how to adopt the weighted majority algorithm for this game.

(c) Show that if we play this game repeatedly, and a player i player used a no-regret algorithm, than this payer is guaranteed to get $N_i \geq Opt/2$ customers, independent of the strategies used by other players. More precisely, assume that the player has small total regret over T steps at most ϵTN, then he/she is guaranteed an average value at least $Opt/2 - f(\epsilon)$, where $f(\epsilon)$ goes to zero as ϵ goes to zero.