Algorithmic Game Theory: Problem Set 2

Due online via GradeScope before the start of class on Tuesday, February 23

Aaron Roth

Remember you can work together on problem sets, but list everyone you worked with, and everyone turn in their own assignment. Ask questions on Piazza.

Problem 1) Maximum-Weight Best Response Dynamics (20 points)

Consider the load balancing game on identical machines that we studied in Lecture 4. We proved that best response dynamics converges for this game. In this problem, we prove that a modification of best response dynamics converges quickly.

Maximum-Weight Best Response Dynamics

Let $a = (a_1, \ldots, a_n)$ be an arbitrary action profile.
while a is not a pure strategy Nash equilibrium **do**
  Among all players i who are not playing best responses in a, let i be the index of the player with largest weight w_i, and let a'_i be a best response to a_{-i}.
  Update the action profile to (a'_i, a_{-i}).
end while
Output a.

---

1. Prove that $\min_j \ell_j(a)$, the minimum load among all the machines, is non-decreasing as Maximum-Weight Best Response Dynamics is run (5 pts)
2. Call a player i "active" if she is currently not playing a best response, and inactive otherwise. Show that a player never goes from being "inactive" to being "active" unless another player moves onto her machine. (5 pts)
3. Let i denote the active player of maximum weight at some intermediate step of Maximum-Weight Best Response Dynamics. Show that in the round after i makes a best response move, any other players i' who become active must have have $w'_i < w_i$. Conclude that no player ever makes a best response move more than once, and hence that Maximum-Weight Best Response Dynamics converges after at most n steps, where n is the number of players. (10 pts)

Problem 2) Bandwidth Sharing Game (20 points)

As you saw on the last homework, even games with infinite action sets can have pure strategy Nash equilibria; here you will show another example of such a game using the potential function method.

In this game, each player wants to send flow along a shared channel of maximum capacity 1. On the one hand, each player wants to send as much flow as possible along the channel. On the other hand, the channel becomes less useful the closer it gets to its maximum capacity. Each player can choose to send an amount of flow $x_i \in [0,1]$ along the channel. (That is, the action set for each player i is $A_i = [0,1]$, and hence is not finite). For a profile of actions $x \in A$, player i has utility $u_i(x_i, x_{-i}) = x_i(1 - \sum_{j=1}^{n} x_j)$.

1. (10 points) Show that this game is an exact potential game, and conclude that it has a pure strategy Nash equilibrium. (Hint: First write down how much player i's utility changes, fixing the actions of all of the other player when i unilaterally deviates. Then try and find a potential function ϕ that changes by exactly this amount.)

2. (5 points) Find a Nash equilibrium of this game. What is the social welfare at this equilibrium? (*i.e.* the sum of utilities of all the players.)

3. (5 points) What is the optimal social welfare? (*i.e.* what is the social welfare at the profile of actions that maximizes it, regardless of whether or not this profile is an equilibrium.)

Problem 3) Load Balancing on Unrelated Machines (15 points)

Consider the following load balancing game: There are m machines $F = \{1, \ldots, m\}$ and n players. Each player i may choose a single machine on which to run his job: for all i, $A_i = F$. But now each player's job may take a different amount of time to run on each machine (the machines are not identical – one may have a faster CPU, one may have a faster graphics card, etc). For each machine $j \in F$ and each player i, there is a corresponding weight $w_{i,j}$ describing how long it takes to run player i's job on machine j. The cost of machine j is the sum of the weights of the jobs assigned to it: $\ell_j(a) = \sum_{i:a_i=j} w_{i,j}$. The cost of player i is the cost of the machine he selected: $c_i(a) = \ell_{a_i}(a)$.

The load balancing game we considered in class was the special case when each player had the same weight on every machine: $w_{i,j} = w_{i,j'} \equiv w_i$ for all j, j'.

1. (5 points) Fix an action profile a and suppose player i makes a best-response move (*i.e.* one that strictly decreasing his cost) from playing on machine a_i to playing on machine a'_i. Write $a' = (a'_i, a_{-i})$. Show that for all $j \notin \{a_i, a'_i\}$, $\ell_j(a) = \ell_j(a')$ and that:

$$\max(\ell_{a_i}(a'), \ell_{a'_i}(a')) < \max(\ell_{a_i}(a), \ell_{a'_i}(a))$$

2. (10 points) Show that the load balancing games on unrelated machines has an ordinal potential function (equivalently, a total ordering on all action profiles $a \in A$ such that best response moves only move to states later in the ordering), and conclude that such games always have pure strategy Nash equilibria. *Hint: This is not an exact potential game, so don't try and come up with an exact potential function. The best way to think about this problem is to try and identify an ordering on the action profiles.*