Algorithmic Game Theory – Handout 6

Uriel Feige, Robert Krauthgamer, Moni Naor

December 10, 2008

We consider graphical games with n players where the graph G is a tree of maximum degree d, and each player has 2 possible actions. We assume all payoffs are in $[0, 1]$, represented by $O(n)$ bits. Unless stated otherwise, we assume d is constant (with respect to n), and measure complexity (e.g. running time) only in terms of n. Here are some open problem:

- Is there a polynomial-time algorithm for computing an (exact) Nash equilibrium in (graphical) tree games (with constant d)? (The known algorithms find ε-Nash equilibrium.)
- Can the algorithm for ε-Nash equilibrium in tree game be extended to general d? (The known algorithm's runtime grows like d^d, slightly super-polynomial in the description size 2^d.)
- Can the algorithm for ε-Nash equilibrium in tree game be generalized to, say, planar graphs?

Reading. More information on graphical games can be found in [NRTV, Chapter 7] and references therein.

Homework. Please keep the answers to the following questions short and easy to read.

1. Prove that for every $\varepsilon, d > 0$ there is $\tau = \tau(d, \varepsilon)$, such that in every (graphical) tree game as above (n players, maximum degree d, 2 strategies for every player), for every Nash equilibrium there exists an ε-Nash equilibrium, where all players' probabilities are integer multiples of τ, and in addition the ε-Nash approximates the given Nash in the sense that each player's expected payoff is changed by no more than ε.

   Remark: For full credit, show $\tau \geq \Omega(\varepsilon/d)$.
2. For a given game, let OPT denote the maximum, over all Nash equilibria, of the social welfare (i.e. total over all players of expected payoff). Show that for every constant $\varepsilon > 0$ there is a polynomial-time algorithm that given as input a (graphical) tree game with constant d, computes an ε-Nash equilibrium with social welfare at least $OPT - n\varepsilon$.
3. Explain how to generalize the algorithm shown in class for finding an ε-Nash equilibrium in a (graphical) tree game to the following graph families: (a) G is a two-dimensional (rectangular) grid of size $r \times (n/r)$ for constant r; and (b) G is a cycle on n vertices. The running time should be polynomial in n.
4. **Extra credit:** Show that the case $\varepsilon = 0$ in Question 2, namely, the problem of finding a Nash equilibrium whose average utility is maximal (among all Nash equilibria) is NP-hard.

References

[NRTV] Noam Nisan, Tim Roughgarden, Eva Tardos and Vijay V. Vazirani (Editors), Algorithmic Game Theory, Cambridge University Press, 2007.