CS364A: Take-Home Final

Due to Peerapong in Gates 460 by 3:30 PM on Thursday, March 17, 2011

Instructions: Same as the problem sets.

Problem 21

Let A denote the payoff matrix of a two-player zero-sum game, with A_{ij} denoting the integral payoff from the column player to the row player in the outcome (i, j).

(a) (5 points) A *minimax pair* of mixed strategies $\hat{x}, \hat{y}$ satisfies $\hat{x} \in \arg\max_x(\min_y x^T Ay)$ and $\hat{y} \in \arg\min_y(\max_x x^T Ay)$. Prove that the mixed-strategy Nash equilibria (of a two-player, zero-sum game) are precisely the minimax pairs.

(b) (5 points) Prove that such a minimax pair (and hence a mixed-strategy Nash equilibrium) can be computed in polynomial time.

[Hint: you should assume and use the fact that linear programming can be solved in polynomial time.]

(c) (5 points) Recall that we proved the minimax theorem using no-regret algorithms. State and prove a quantitative version, in the form of an algorithm that computes an "approximate Nash equilibrium" (in some sense that you should define precisely) and that runs in time polynomial in the input size and in the reciprocal of your approximation parameter.

Problem 22

Consider a two-player, non-zero sum game, specified by payoff matrices A and B, in which both players have n strategies. Assume that all payoffs are rational numbers between 0 and 1.

(a) (4 points) The *support* of a mixed strategy is the set of pure strategies that are played with strictly positive probability. Suppose you were promised that there was a mixed-strategy Nash equilibrium of (A, B) with row and column supports S and T, respectively. Show how you can then recover such an equilibrium in polynomial time.

[Hint: again, feel free to use linear programming.]

(b) (3 points) Give a $2^{O(n)}$-time algorithm to compute an exact mixed-strategy Nash equilibrium of a two-player game.

(c) (9 points) Consider a mixed-strategy Nash equilibrium (x^*, y^*) of (A, B). Suppose we draw $K = (s \log n)/\epsilon^2$ independent samples $(r_1, c_1), \ldots, (r_K, c_K)$ from the product distribution $x^* \times y^*$, where $\epsilon > 0$ is a parameter and $s > 0$ is a sufficiently large constant. Form the corresponding empirical distributions $\hat{x}$ and $\hat{y}$, where components are the frequencies of play in the sequence. Prove that, with high probability, $(\hat{x}, \hat{y})$ is an ϵ-approximate Nash equilibrium in the sense that every strategy in a player's support has expected payoff within (additive) ϵ of that of a best response.

[Hint: if you don't know about them, read up about Chernoff Bounds (e.g., in Motwani/Raghavan's *Randomized Algorithms*).]

(d) (5 points) Use (an extension of) (a) and (c) to give an $O(n^{(\log n)/\epsilon^2})$-time algorithm for computing an ϵ-approximate mixed-strategy Nash equilibrium of a two-player game.

(e) (4 points) Extend your algorithm and analysis to compute, with the same running time bound, an ϵ-approximate mixed-strategy Nash equilibrium with expected total payoff close to that of the best (maximum-payoff) exact Nash equilibrium.

Problem 23

We continue the study of algorithms for approximate Nash equilibria in bimatrix games, in the sense of the previous problem. However, we'll use a weaker definition of approximate equilibria than in Problem 22(c): now, we'll call $(\hat{x}, \hat{y})$ an *ϵ-approximate Nash equilibrium* if $\hat{x}^T A\hat{y} \geq x^T A\hat{y} - \epsilon$ for all row mixed strategies x and $\hat{x}^T B\hat{y} \geq \hat{x}^T By - \epsilon$ for all column mixed strategies y.

(a) (5 points) Prove that there is a $\frac{1}{2}$-approximate Nash equilibrium in which one player plays a pure strategy and the other randomizes uniformly between two strategies.

(b) (10 points) Consider the following idea for improving on (a). Suppose we form the matrix $A - B$, view it as a zero-sum game, and solve for an exact mixed-strategy Nash equilibrium (x^*, y^*). (By Problem 22(c), this can be done in polynomial time.) Perhaps (x^*, y^*) is an α-approximate Nash equilibrium for (A, B) for a satisfactory value of α; if not, we generalize (a) and have one player play a pure strategy while the other randomizes (not necessarily uniformly) between a pure strategy and its optimal strategy for $A - B$.

Can you use these ideas (with appropriate parameter choices) to improve upon the result in (a)? What is the minimum value of ϵ for which your algorithm computes an ϵ-approximate Nash equilibrium in polynomial time?

Problem 24

Consider again the problem of computing some exact mixed-strategy Nash equilibrium of a (non-zero-sum) bimatrix game.

(a) (5 points) A bimatrix game (A, B) is *symmetric* if both players have the same strategy set and $B = A^T$. (Rock-Paper-Scissors and the Prisoner's Dilemma are both examples.) Nash's theorem (or the Lemke-Howson algorithm) can be adapted to show that every symmetric game has a *symmetric* Nash equilibrium, in which both players employ the same mixed strategy. Prove that the problem of computing a mixed Nash equilibrium of a general bimatrix game polynomial-time reduces to that of computing a symmetric Nash equilibrium of a symmetric bimatrix game.

(b) (5 points) Prove that the problem of computing a mixed Nash equilibrium of a general bimatrix game polynomial-time reduces to that of computing *any* Nash equilibrium of a symmetric bimatrix game.

(c) (5 points) Prove that the problem of computing a mixed Nash equilibrium of a general bimatrix game polynomial-time reduces to that of computing an *asymmetric* Nash equilibrium of a symmetric bimatrix game, if one exists.

(d) (10 extra credit points) Prove that it is NP-complete to decide whether or not a symmetric bimatrix game has an asymmetric Nash equilibrium.

Problem 25

(a) (10 points) *Algorithmic Game Theory*, Exercise 10.6.

(b) (10 points) *Algorithmic Game Theory*, Exercise 10.7.