An Optimization Approach for Approximate Nash Equilibria*

Haralampos Tsaknakis[1] and Paul G. Spirakis[1,2]

[1] Research Academic Computer Technology Institute (RACTI), Greece
{tsaknak,spirakis}@cti.gr
[2] Dept. of Computer Eng. and Informatics, Patras University, Patras Greece

Abstract. In this paper we propose a new methodology for determining approximate Nash equilibria of non-cooperative bimatrix games and, based on that, we provide an efficient algorithm that computes 0.3393-approximate equilibria, the best approximation till now. The methodology is based on the formulation of an appropriate function of pairs of mixed strategies reflecting the maximum deviation of the players' payoffs from the best payoff each player could achieve given the strategy chosen by the other. We then seek to minimize such a function using descent procedures. As it is unlikely to be able to find global minima in polynomial time, given the recently proven intractability of the problem, we concentrate on the computation of stationary points and prove that they can be approximated arbitrarily close in polynomial time and that they have the above mentioned approximation property. Our result provides the best ϵ till now for polynomially computable ϵ-approximate Nash equilibria of bimatrix games. Furthermore, our methodology for computing approximate Nash equilibria has not been used by others.

1 Introduction

Ever since it was proved that the problem of finding exact Nash equilibria is intractable in the sense that it is PPAD-complete even for 2-player games [2], attention has been focused on finding ϵ-approximate such equilibria for $\epsilon > 0$. In this respect, simple algorithms have recently been provided for finding approximate equilibria for constant $\epsilon = \frac{3}{4}$ and $\epsilon = \frac{1}{2}$ ([4], [5]) for general bimatrix games (and for positively normalized payoff matrices) based on examining small supports of 1 or 2 for either player. A well known result provides 0.38-approximate Nash equilibria of normalized bimatrix games in polynomial time ([3]). Concurrently with us , [1] gave an approach based on [7] that provides 0.36-approximate Nash equilibria of normalized bimatrix games. Furthermore, it has been shown ([6]) that the more general approximation problem of finding a fully polynomial time approximation scheme for any $\epsilon > 0$, has similar complexity with the problem of finding exact Nash equilibria.

* This work has been partially supported by the IST 6th Framework Programme of the European Union under contract 001907 DELIS.

X. Deng and F.C. Graham (Eds.): WINE 2007, LNCS 4858, pp. 42–56, 2007.

For a different, stronger, notion of approximation, i.e. the well supported approximate Nash equilibria, the best known result so far provides 0.658-approximate well supported equilibria for normalized bimatrix games in polynomial time ([7]).

Most of the reported investigations of finding approximate equilibria for constant ϵ are based on the examination of small supports of the strategy sets of the players and the algorithms presented are based on brute force search over all such supports.

In this work we adopt a different approach that does not rely on any prespecified small supports neither on an indiscriminate search over all small support strategies. We define an equivalent optimization problem in the strategy spaces of both players and attempt to obtain a stationary point of a specific function that measures the maximum deviation of the players' payoffs from the best payoff each player could achieve given the strategy chosen by the other. We do so through a descent procedure along feasible directions in the strategy spaces of both players simultaneously. Feasible descent directions are computed by solving linear programming problems. Also, by solving similar linear programs we can determine whether or not there is a descent direction at any given point in the strategy spaces. If a descent direction does not exist, then we have reached a stationary point. We prove that at any stationary point of that function we obtain strategy pairs such that at least one of them is an 0.3393-approximate Nash equilibrium. We also prove that an almost stationary point of the function can be reached in polynomial time with respect to the input data of the game, and that point suffices to get arbitrarily close to 0.3393. Our work can be accessed as a full technical report (revised) also in [10].

2 Definitions and Notation

Let R, C denote the m by n row and column players' payoff matrices respectively, for m, n any positive integers. We assume that both payoff matrices are positively normalized, i.e. all their entries belong to $[0, 1]$ (without loss of generality any game can be equivalently transformed to a positively normalized game by appropriate shifting and scaling each one of the payoff matrices).

Let us denote by e_k the k-dimensional column vector having all its entries equal to 1 (for positive integer k). Let

$$\Delta_k = \{u : u \in R^k, u \geq 0, e_k^\tau u = 1\}$$

be the k-dimentional standard simplex (superscript τ denotes transpose).

Also, for any vector $u \in R^k$, we define the following :

$$supp(u) = \{i \in (1, k) : u_i \neq 0\}$$

being the support index subset of $u \in R^k$ and also

$$suppmax(u) = \{i \in (1, k) : u_i \geq u_j \ \forall j \in (1, k)\}$$

being the index subset where all entries are equal to the maximum entry of $u \in R^k$.

We also denote by

$$max(u) = \{u_i : u_i \geq u_j, \text{for all } j\}$$

the value of the maximum entry of the vector and by

$$max_S(u) = \{u_i, i \in S : u_i \geq u_j, \text{for all } j \in S\}$$

the value of the maximum entry of the vector within an index subset $S \subset (1, k)$.

Finally, we denote by $\overline{S}$ the complement of an index set S, i.e. $\overline{S} = \{i \in (1, k), i \notin S\}$.

The problem of finding an ϵ-approximate Nash equilibrium in the game (R, C), for some $\epsilon \geq 0$, is to compute a pair of strategies $\overline{x}$ in Δ_m and $\overline{y}$ in Δ_n such that the following relationships hold :

$$x^\tau R\overline{y} \leq \overline{x}^\tau R\overline{y} + \epsilon \text{ for all } x \in \Delta_m$$

and

$$\overline{x}^\tau Cy \leq \overline{x}^\tau C\overline{y} + \epsilon \text{ for all } y \in \Delta_n$$

3 Optimization Formulation

Key to our approach is the definition of the following continuous function mapping $\Delta_m \times \Delta_n$ into $[0, 1]$:

$$f(x, y) = max\{max(Ry) - x^\tau Ry, max(C^\tau x) - x^\tau Cy\} \tag{1}$$

It is evident that $f(x, y) \geq 0$ for all $(x, y) \in \Delta_m \times \Delta_n$ and that exact Nash equilibria of (R, C) correspond to pairs of strategies such that $f(x, y) = 0$. Furthermore, ϵ- approximate equilibria correspond to strategy pairs that satisfy $f(x, y) \leq \epsilon$. This function represents the maximum deviation of the players' payoffs from the best payoff each player could achieve given the strategy chosen by the other.

An optimization formulation based on mixed integer programming methods was suggested in [9]. However, no approximation results were obtained there.

The function $f(x, y)$ is not jointly convex with respect to both x and y. However, it is convex in x alone, if y is kept fixed and vice versa.

Let us define the two ingredients of the function $f(x, y)$ as follows :
$f_R(x, y) = max(Ry) - x^\tau Ry$
and
$f_C(x, y) = max(C^\tau x) - x^\tau Cy$

From any point in $(x, y) \in \Delta_m \times \Delta_n$ we consider variations of $f(x, y)$ along feasible directions in both players' strategy spaces of the following form :

$$(1 - \epsilon) \begin{bmatrix} x \\ y \end{bmatrix} + \epsilon \begin{bmatrix} x' \\ y' \end{bmatrix}$$

where, $0 \leq \epsilon \leq 1, (x', y') \in \Delta_m \times \Delta_n$ (the vectors in brackets are $m+n$ - dimensional column vectors).

The variation of the function along such a feasible direction is defined by the following relationship:

$$Df(x,y,x',y',\epsilon) = f(x + \epsilon(x' - x), y + \epsilon(y' - y)) - f(x,y)$$

We have derived an explicit formula for $Df(x,y,x',y',\epsilon)$ (see Appendix), which is a piecewise quadratic function of ϵ and the number of switches of the linear terms of the function is at most $m+n$. Therefore, for fixed (x',y') this function can be minimized with respect to ϵ in polynomial time. Furthermore, there always exists a positive number, say $\epsilon^\star$, such that for any $\epsilon \leq \epsilon^\star$ the coefficient of the linear term of this function of ϵ coincides with the gradient, as defined below. The number $\epsilon^\star$ generally depends on both (x,y) and (x',y').(See Appendix A.3).

We define the gradient of f at the point (x,y) along an arbitrary feasible direction specified by another point (x',y') as follows:

$$Df(x,y,x',y') = \lim_{\epsilon \to 0} \frac{1}{\epsilon} Df(x,y,x',y',\epsilon)$$

The gradient $Df(x,y,x',y')$ of f at any point $(x,y) \in \Delta_m \times \Delta_n$ along a feasible direction (determined by another point $(x',y') \in \Delta_m \times \Delta_n$) provides the rate of decrease (or increase) of the function along that direction. For fixed (x,y), $Df(x,y,x',y')$ is a convex polyhedral function in (x',y'). In fact we have derived the explicit form of $Df(x,y,x',y')$ as the maximum of two linear forms in the (x',y') space (see the derivations below and in the Appendix A.1). At any point (x,y) we wish to minimize the gradient function with respect to (x',y') to find the steepest possible descent direction, or to determine that no such descent is possible.

Let us define the following index sets:

$$S_R(y) = suppmax(Ry) \; and \; S_C(x) = suppmax(C^\tau x)$$

By definition, $S_R(y) \subset (1,m)$ and $S_C(x) \subset (1,n)$.

From the Appendix A.1 we get :

(a) If $f_R(x,y) = f_C(x,y)$ then

$$Df(x,y,x',y') = max(T_1(x,y,x',y'), T_2(x,y,x',y')) - f(x,y)$$

where

$$m_1(y') = max(Ry') \; over \; the \; subset \; S_R(y)$$

and

$$m_2(x') = max(C^\tau x') \; over \; the \; subset \; S_C(x)$$

and

$$T_1(x,y,x',y') = m_1(y') - x^\tau Ry' - (x')^\tau Ry + x^\tau Ry$$

and

$$T_2(x,y,x',y') = m_2(x') - x^\tau Cy' - (x\prime)^\tau Cy + x^\tau Cy$$

(b) If $f_R(x,y) > f_C(x,y)$ then
$Df(x,y,x',y') = T_1(x,y,x',y') - f(x,y)$
and
(c) If $f_R(x,y) < f_C(x,y)$ then
$Df(x,y,x',y') = T_2(x,y,x',y') - f(x,y)$. In the cases (b) and (c) the functions T_1 and T_2 are as defined in case (a).

The problem of finding $Df(x,y)$ as the minimum over all $(x',y') \in \Delta_m \times \Delta_n$ of the function $Df(x,y,x',y')$, is a linear programming problem.

This problem can be equivalently expressed as the following mini-max problem by introducing appropriate dual variables (we derive it for (x,y) such that $f_R(x,y) = f_C(x,y)$ since this is the most interestng case and the cases where the two terms are different can be reduced to this by solving an LP, as we shall see below) as follows :

Minimize (over x', y') the maximum (over w, z, ρ) of the function

$$[\rho w^\tau, (1-\rho)z^\tau]G(x,y)\begin{bmatrix} y' \\ x' \end{bmatrix}$$

where :

(a) the maximum is taken with respect to dual variables w, z, ρ such that :
$w \in \Delta_m, supp(w) \subset S_R(y)$ and $z \in \Delta_n, supp(z) \subset S_C(x)$ and $\rho \in [0,1]$.
(b) The minimum is taken with respect to $(x',y') \in \Delta_m \times \Delta_n$, and
(c) the matrix $G(x,y)$ is the following $(m+n)$ *by* $(m+n)$ matrix :

$$G(x,y) = \begin{bmatrix} R - e_m x^\tau R & -e_m y^\tau R^\tau + e_m {e_m}^\tau x^\tau Ry \\ -e_n x^\tau C + e_n {e_n}^\tau x^\tau Cy & C^\tau - e_n y^\tau C^\tau \end{bmatrix}$$

The probability vectors w and z play the role of price vectors (or penalty vectors) for penalizing deviations from the support sets $S_R(y)$ and $S_C(x)$, and the parameter ρ plays the role of a trade-off parameter between the two parts of the function $f(x,y)$. In fact, the w, z and ρ are not independent variables but they are taken all together to represent a single $(m+n)$-dimensional probability vector on the left hand side (the maximizing term) of the linear mini-max problem.

Solving the above mini-max problem we obtain w, z, ρ, x' and y' that are all functions of the point (x,y) and take values in their respective domains of definition. Let us denote by $V(x,y)$ the value of the solution of the mini-max problem at the point (x,y). The solution of this problem yields a feasible descent direction (as a matter of fact the steepest feasible descent direction) for the function $f(x,y)$ if $Df(x,y) = V(x,y) - f(x,y) < 0$. Following such a descent direction we can perform an appropriate line search with respect to the parameter ϵ and find a new point that gives a lower value of the function $f(x,y)$. Applying repeatedly such a descent procedure we will eventually reach a point where no further reduction is possible. Such a point is a stationary point that satisfies $Df(x,y) \geq 0$.

In the next section we examine the approximation properties of stationary points. In fact, we prove that given any stationary point we can determine pairs of strategies such that at least one of them is a 0.3393-approximate Nash equilibrium.

4 Approximation Properties of Stationary Points

Let us assume that we have a stationary point $(x^\star, y^\star)$ of the function $f(x, y)$. Then, based on the above analysis and notation, the following relationship should be true :

$$Df(x^\star, y^\star) = V(x^\star, y^\star) - f(x^\star, y^\star) \geq 0$$

Let $(w^\star, z^\star) \in \Delta_m \times \Delta_n, \rho^\star \in [0, 1]$ be a solution of the linear mini-max problem (with matrix $G(x^\star, y^\star)$) with respect to the dual variables corresponding to the pair $(x^\star, y^\star)$. Such a solution should satisfy the relations $supp(w^\star) \subset S_R(y^\star)$ and $supp(z^\star) \subset S_C(x^\star)$.

Let us define the following quantities:

$$\lambda = \min_{y' : supp(y') \subset S_C(x^\star)} \{(w^\star - x^\star)^T R y'\}$$

and

$$\mu = \min_{x' : supp(x') \subset S_R(y^\star)} \{x'^T C(z^\star - y^\star)\}.$$

From the fact that R, C are positively normalized it follows that both λ and μ are less than or equal to 1.

At any point $(x^\star, y^\star)$ these quantities basically define the rates of decrease (or increase) of the function f along directions of the form $(1-\epsilon)(x^\star, y^\star) + \epsilon(x^\star, y')$ and $(1-\epsilon)(x^\star, y^\star) + \epsilon(x', y^\star)$, i.e. the rates of decrease that are obtained when we keep one player's strategy fixed and move probability mass of the other player into his own maximum support, towards decreasing his own deviation from the maximum payoff he can achieve.

From the stationarity property of the point $(x^\star, y^\star)$ it follows that both λ and μ are nonnegative. Indeed, in the opposite case there would be a descent direction, which contradicts the stationarity condition.

Let us define a pair of strategies $(\hat{x}, \hat{y}) \in \Delta_m \times \Delta_n$ as follows:

$$(\hat{x}, \hat{y}) = \begin{cases} (x^\star, y^\star) & \text{,if } f(x^\star, y^\star) \leq f(\tilde{x}, \tilde{y}) \\ (\tilde{x}, \tilde{y}) & \text{,otherwise} \end{cases}$$

where

$$(\tilde{x}, \tilde{y}) = \begin{cases} \left(\frac{1}{1+\lambda-\mu} w^\star + \frac{\lambda-\mu}{1+\lambda-\mu} x^\star, z^\star\right) & \text{,if } \lambda \geq \mu \\ \left(w^\star, \frac{1}{1+\mu-\lambda} z^\star + \frac{\mu-\lambda}{1+\mu-\lambda} y^\star\right) & \text{,if } \lambda < \mu. \end{cases}$$

We now express the main result of this paper in the following theorem :

Theorem 1. *The pair of strategies* $(\hat{x}, \hat{y})$ *defined above, is a* 0.3393*-approximate Nash equilibrium.*

Proof. From the definition of $(\hat{x}, \hat{y})$ we have :

$$f(\hat{x}, \hat{y}) \leq min\{f(x^\star, y^\star), f(\tilde{x}, \tilde{y})\} \tag{2}$$

Using the stationarity condition for $(x^\star, y^\star)$ we obtain :

$$f(x^\star, y^\star) \leq V(x^\star, y^\star)$$

But $V(x^\star, y^\star)$ is less than or equal to

$$\rho^\star E_1 + (1 - \rho^\star) E_2$$

where

$$E_1 = (w^{\star\tau} R y' - x^{\star\tau} R y' - x'^{\tau} R y^\star + x^{\star\tau} R y^\star)$$

and

$$E_2 = (z^{\star\tau} C^\tau x' - x^{\star\tau} C y' - x'^{\tau} C y^\star + x^{\star\tau} C y^\star)$$

and this holds $\forall (x', y') \in \Delta_m \times \Delta_n$
Setting $x' = x^\star$ and $y' : supp(y') \subset S_C(x^\star)$ in the above inequality we get :

$$f(x^\star, y^\star) \leq \rho^\star \lambda. \tag{3}$$

Next, setting $y' = y^\star$ and $x' : supp(x') \subset S_R(y^\star)$ in the same inequality, we get :

$$f(x^\star, y^\star) \leq (1 - \rho^\star) \mu. \tag{4}$$

Now using the definition of the strategy pair $(\tilde{x}, \tilde{y})$ above and exploiting the inequalities

$$\begin{gathered}(w^\star - x^\star)^T R z^\star \geq \lambda, \text{since } supp(z^\star) \subset S_C(x^\star) \\ w^{*T} C(z^\star - y^\star) \geq \mu, \text{since } supp(w^\star) \subset S_R(y^\star)\end{gathered}$$

we obtain: (assume $\lambda \geq \mu$)

$$\begin{aligned} f_R(\tilde{x}, \tilde{y}) &= \max\{R\tilde{y}\} - \tilde{x}^T R \tilde{y} = \max\{Rz^\star\} - \left(\frac{1}{1 + \lambda - \mu} w^\star + \frac{\lambda - \mu}{1 + \lambda - \mu} x^\star\right)^T R z^\star \\ &= \max\{Rz^\star\} - \frac{1}{1 + \lambda - \mu} w^{*T} R z^\star - \frac{\lambda - \mu}{1 + \lambda - \mu} x^{*T} R z^\star \\ &\leq \max\{Rz^\star\} - x^{*T} R z^\star - \frac{\lambda}{1 + \lambda - \mu} \leq \frac{1 - \mu}{1 + \lambda - \mu}. \end{aligned}$$

Similarly, setting $D = C^T$,

$$\begin{aligned}
f_C(\tilde{x},\tilde{y}) &= \max\{D\tilde{x}\} - \tilde{x}^T C\tilde{y} \\
&= \max\left\{\frac{1}{1+\lambda-\mu}Dw^\star + \frac{\lambda-\mu}{1+\lambda-\mu}Dx^\star\right\} - \frac{1}{1+\lambda-\mu}w^{*T}Cz^\star - \frac{\lambda-\mu}{1+\lambda-\mu}x^{*T}Cz^\star \\
&\leq \frac{1}{1+\lambda-\mu}\max\{Dw^\star\} + \frac{\lambda-\mu}{1+\lambda-\mu}\max\{Dx^\star\} - \frac{1}{1+\lambda-\mu}w^{*T}Cz^\star - \\
&\quad - \frac{\lambda-\mu}{1+\lambda-\mu}\max\{Dx^\star\} \\
&= \frac{1}{1+\lambda-\mu}(\max\{Dw^\star\} - w^{*T}Cy^\star) - \frac{1}{1+\lambda-\mu}(w^{*T}Cz^\star - w^{*T}Cy^\star) \\
&\leq \frac{1-\mu}{1+\lambda-\mu}.
\end{aligned}$$

From the above relationships we obtain:

$$f(\tilde{x},\tilde{y}) \leq \frac{1-\mu}{1+\lambda-\mu} \qquad \text{for } \lambda \geq \mu \tag{5}$$

(A similar inequality can be obtained if $\lambda < \mu$ and we interchange λ and μ)

In all cases, combining inequalities (3), (4), (5) and using the definition of $(\hat{x},\hat{y})$ above, we get the following:

$$f(\hat{x},\hat{y}) \leq \min\left\{\rho^\star\lambda, (1-\rho^\star)\mu, \frac{1-\min\{\lambda,\mu\}}{1+\max\{\lambda,\mu\}-\min\{\lambda,\mu\}}\right\}. \tag{6}$$

We can prove that the quantity in (6) cannot exceed the number 0.3393 for any $\rho^\star, \lambda, \mu \in [0,1]$. For the proof see Appendix A.2.

This concludes the proof of our main Theorem.

5 Descent Procedure

A stationary point of any general Linear Complementarity problem can be approximated arbitrarily close **in polynomial time** via the method of Y. Ye [11]. We give here an alternative approach, directly applicable to our problem.

We present here an algorithm for finding a pair of stategies that achieve the 0.3393 approximation bound. The algorithm is based on a descent procedure of the function $f(x,y)$, $(x,y) \in \Delta_m \times \Delta_n$, and consists of the following steps: (set $b = 0.3393$)

1. Start with an arbitrary $(x,y) = (x_0,y_0)$ in $\Delta_m \times \Delta_n$ (e.g. the uniform distribution). Produce another pair (x,y) with lower value of $f(x,y)$ and for which $f_R(x,y) = f_C(x,y)$ as follows :
   (a) If $f_R(x_0,y_0) > f_C(x_0,y_0)$, keep y_0 fixed and solve the LP :
   minimize (over $x \in \Delta_m$) the

$$max(Ry_0) - x^\tau Ry_0$$

under the constraints :

$$max(C^{\tau}x) - x^{\tau}Cy_0 \leq max(Ry_0) - x^{\tau}Ry_0$$

(b) If $f_R(x_0, y_0) < f_C(x_0, y_0)$, keep x_0 fixed and solve the LP : minimize (over $y \in \Delta_n$) the

$$max(C^{\tau}x_0) - {x_0}^{\tau}Cy$$

under the constraints :

$$max(Ry) - {x_0}^{\tau}Ry \leq max(C^{\tau}x_0) - {x_0}^{\tau}Ry$$

2. Solve the linear minimax problem with the matrix $G(x, y)$ as defined in section 3. Compute the value of $V(x, y)$, the pair of strategies (x', y'), the index sets $S_R(y) \subset (1, m)$, $S_C(x) \subset (1, n)$, the vectors w, z, the parameter ρ, and the values of λ, μ as defined in sections 3 and 4 for the current point (x, y). Also determine the pair of strategies $(\tilde{x}, \tilde{y})$ as defined in section 4.
3. If at least one of the following conditions is true, stop and exit – a pair of strategies achieving the approximation bound b has been found.
   (i) $V(x, y) - f(x, y) \geq 0$ (stationary condition: either $f(x, y)$ or $f(\tilde{x}, \tilde{y})$ is $\leq b$)
   (ii) $f(x, y) \leq b$
   (iii) $f(\tilde{x}, \tilde{y}) \leq b$
   (iv) $f(x', y') \leq b$
   (v) $f(x', y) \leq b$
   (vi) $f(x, y') \leq b$
4. If none of the conditions of step 3 is satisfied, compute the minimum with respect to ϵ of the function $f(x + \epsilon(x' - x), y + \epsilon(y' - y))$ along the direction specified by the pair (x', y') found in step 2, and set $(x, y) = (x + \epsilon(x' - x), y + \epsilon(y' - y))$ (such a minimization with respect to ϵ can be performed in polynomial time, as mentioned earlier, since the number of switches of the linear terms of the piecewise quadratic function cannot exceed $m + n$).
   Furthermore, if for the new pair (x, y) we have $f_R(x, y) \neq f_C(x, y)$, solve the LP specified in Step 1 and compute the new (x, y) with lower value of the function $f(x, y)$ and for which $f_R(x, y) = f_C(x, y)$.
   Go to Step 2.
   End of descent.

In regard to the number of steps that are required for convergence and exit, we provide a convergence analysis in Appendix A.3 that shows that the algorithm converges in a polynomial number of iterations.

6 The Complexity of Our Algorithm

Our algorithm is basically the procedure descent of the function $f(x, y)$. The number q of the descent steps for convergence, given any $\delta > 0$, is $O(\frac{1}{\delta^2})$ and that suffices to get an $0.3393 + \delta$-approximate equilibrium.

So, the total time complexity of our method is $O(\frac{1}{\delta^2})T_LP(n)$ time (when $n \geq m$) where $T_LP(n)$ is the time to solve a linear program of size n. Thus, our method is an FPTAS with respect to approximating a stationary point and hence an approximate equilibrium of the stated quality.

An arbitrary point $(x, y) \in \Delta_m \times \Delta_n$ can be used to initialize the algorithm.

7 Discussion and Future Work

It is known from Bellare and Rogaway ([8]) that (even in a weaker sense) there is no polynomial time μ - approximation of the optimal value of the problem $min\{x^{\tau}Qx, s.t.Bx = b, 0 \leq x \leq e\}$ for some $\mu \in (0, \frac{1}{3})$, unless $P = NP$. Of course, here μ is a multiplicative relative accuracy and the reduction that they use involves matrices that are different from the ones in our case. However, this gives evidence that going below $\frac{1}{3}$ in the approximation of equilibria will probably require a radically different approach (if any), perhaps probabilistic. We are currently working on this.

References

1. Boss, H., Byrka, J., Markakis, E.: New Algorithms for Approximate Nash Equilibria in Bimatrix Games. In: this volume of WINE (2007)
2. Chen, X., Deng, X.: Settling the complexity of 2-player Nash equilibrium. In: Proc. of the 47th IEEE Symp. on Foundations of Comp. Sci (FOCS 2006), pp. 261–272. IEEE Computer Society Press, Los Alamitos (2006)
3. Daskalakis, C., Mehta, A., Papadimitriou, C.: Progress in approximate Nash Equilibria. In: Proc. of the 8th ACM Conf. on Electronic Commerce (EC 2007), ACM Press, New York (2007)
4. Daskalakis, C., Mehta, A., Papadimitriou, C.: A note on approximate Nash equilibria. In: Spirakis, P.G., Mavronicolas, M., Kontogiannis, S.C. (eds.) WINE 2006. LNCS, vol. 4286, pp. 297–306. Springer, Heidelberg (2006)
5. Kontogiannis, S., Panagopoulou, P., Spirakis, P.G.: Polynomial algorithms for approximating Nash equilibria in bimatrix games. In: Spirakis, P.G., Mavronicolas, M., Kontogiannis, S.C. (eds.) WINE 2006. LNCS, vol. 4286, pp. 282–296. Springer, Heidelberg (2006)
6. Chen, X., Deng, X., Teng, S.: Computing Nash Equilibria: Approximation and smoothed complexity. In: Proc. of the 47th IEEE Symp. on Foundations of Comp. Sci (FOCS 2006), pp. 603–612. IEEE Press, New York (2006)
7. Kontogiannis, S., Spirakis, P.G.: Efficient algorithms for constant well supported approximate equilibria of bimatrix games. In: ICALP 2007 (2007)
8. Bellare, M., Rogaway, P.: The complexity of approximating a nonlinear program. Mathematical Programming 69, 429–441 (1995)
9. Sandholm, T., Gilpin, A., Conitzer, V.: Mixed-integer Programming Methods for Finding Nash Equilibria. In: Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 2005), pp. 495–501 (2005)
10. Tsaknakis, H., Spirakis, P.G.: An Optimization Approach for Approximate Nash Equilibria. In: the Electronic Colloqium on Computational Complexity (ECCC), as a Technical Report with number TR07067

11. Ye, Y.: A fully polynomial time approximation algorithm for computing a stationary point of the general linear complementarity problem. Mathematics of Operations Research 18(2), 334–345 (1993)

A Appendix

A.1 Appendix A.1

Using the definitions for any $(x, y) \in \Delta_m \times \Delta_n$ i.e :

$$\begin{aligned} f_R(x, y) &= max(Ry) - x^\tau Ry \\ f_C(x, y) &= max(C^\tau x) - x^\tau Cy \\ f(x, y) &= max\{f_R(x, y), f_C(x, y)\} \end{aligned}$$

we have, for any $(x', y') \in \Delta_m \times \Delta_n$ and any $\epsilon \in [0, 1]$ that :

$$Df(x, y, x', y', \epsilon) = f(x + \epsilon(x' - x), y + \epsilon(y' - y)) - f(x, y)$$

This can be written as (analytically)

$$max\{f_R(x+\epsilon(x'-x), y+\epsilon(y'-y)), f_C(x+\epsilon(x'-x), y+\epsilon(y'-y))\}-max\{f_R(x, y), f_C(x, y)\}$$

and this is actually $max(K_1, K_2)$ where

$$K_1 = \epsilon Df_R + \Lambda f_R - \epsilon^2 Hf_R - (1 - \epsilon)max\{0, f_C(x, y) - f_R(x, y)\}$$

and also

$$K_2 = \epsilon Df_C + \Lambda f_C - \epsilon^2 Hf_C - (1 - \epsilon)max\{0, f_R(x, y) - f_C(x, y)\}$$

where now the functions $Df_R, \Lambda f_R, Hf_R, Df_C, \Lambda f_C, Hf_C$ are defined below.

$$\begin{aligned} Df_R(x, y, x', y') = \{max(Ry')overS_R(y)\} - x^\tau Ry' - x'^\tau Ry + x^\tau Ry - f(x, y) \\ and \\ Hf_R(x, y, x', y') = (x' - x)^\tau R(y' - y) \\ and \\ Df_C(x, y, x', y') = \{max(C^\tau x')overS_C(x)\} - x^\tau Cy' - x'^\tau Cy + x^\tau Cy - f(x, y) \\ and \\ Hf_C(x, y, x', y') = (x' - x)^\tau C(y' - y) \end{aligned}$$

In order to define $\Lambda f_R, \Lambda f_C$ we remind the reader that $S_R(y) = suppmax(Ry)$ and that $S_C(x) = suppmax(C^\tau x)$ and we will also use their complements :

$\bar{S_R}(y)$ being the complement of $S_R(y)$ in the index set $\{1, m\}$ and

$\bar{S_C}(x)$ being the complement of $S_C(x)$ in the index set $\{1,n\}$
Let now

M_y be the maximum of Ry over $S_R(y)$

$M_{y'}$ be the maximum of Ry' over $S_R(y)$

and

M_x be the maximum of $C^\tau x$ over $S_C(x)$

$M_{x'}$ be the maximum of $C^\tau x'$ over $S_C(x)$

Finally $\Lambda f_R(x, y, x', y', \epsilon)$ is the maximum of
(0, max over $\bar{S_R}(y)$ of $(I(y, y') + J(y))$) where

$I(y, y') = \epsilon((Ry' - e_m M_{y'}) + (M_y e_m - Ry))$ and

$J(y) = -(M_y e_m - Ry)$

Also finally $\Lambda f_C(x, y, x', y', \epsilon)$ is also the maximum of
((0, max over $\bar{S_C}(x)$ of $(I(x, x') + J(x))$) where

$I(x, x') = \epsilon((C^\tau x' - e_n M_{x'}) + (M_x e_n - C^\tau x))$ and

$J(x) = -(M_x e_n - C^\tau x)$

From the above equations, the gradient at the point $(x, y) \in \Delta_m \times \Delta_n$ along a feasible direction specified by a $(x', y') \in \Delta_m \times \Delta_n$ can be determined by letting ϵ go to 0 and get finally :

$$Df(x, y, x', y') = \left\{ \begin{array}{ll} max(Df_R, Df_C) & \text{if } f_R(x, y) = f_C(x, y) \\ Df_R & \text{if } f_R(x, y) > f_C(x, y) \\ Df_C & \text{if } f_R(x, y) < f_C(x, y) \end{array} \right\}$$

A.2 Appendix A.2

We first notice that $\min\{\rho^\star\lambda, (1-\rho^\star)\mu\} \leq \frac{\lambda\mu}{\lambda+\mu}$. Indeed, if we assume that $\rho^\star\lambda > \frac{\lambda\mu}{\lambda+\mu}$ and $(1-\rho^\star)\mu > \frac{\lambda\mu}{\lambda+\mu}$ for some $\rho^\star, \lambda, \mu \in [0,1]$, we would have $\rho^\star > \frac{\mu}{\lambda+\mu}$ and $(1-\rho^\star) > \frac{\lambda}{\lambda+\mu}$, a contradiction. So

$$f(\hat{x}, \hat{y}) \leq \min\left\{\frac{\lambda\mu}{\lambda+\mu}, \frac{1-\min\{\lambda,\mu\}}{1+\max\{\lambda,\mu\}-\min\{\lambda,\mu\}}\right\}.$$

Set $\mu = \min\{\lambda, \mu\}$. For $\mu \leq \frac{1}{2}$ and since $\mu \leq \lambda$, we have $\frac{\lambda\mu}{\lambda+\mu} \leq \frac{\lambda \min\{1/2,\lambda\}}{\lambda+\min\{1/2,\lambda\}} \leq \frac{1}{3} < 0.3393$. Also, for $\mu \geq \frac{2}{3}$ we have $1-\mu \leq \frac{1}{3}$ and $\frac{1-\mu}{1+\lambda-\mu} \leq 1-\mu \leq \frac{1}{3} < 0.3393$, since $\lambda \geq \mu \geq \frac{2}{3}$.

Consider now cases for which $\frac{1}{2} < \mu < \frac{2}{3}$. If $\frac{1}{2} < \mu \leq \lambda \leq \frac{2}{3}$, then $\frac{\lambda\mu}{\lambda+\mu} \leq \frac{\lambda}{2} \leq \frac{1}{3} < 0.3393$.

For μ, λ such that $\frac{1}{2} < \mu < \frac{2}{3} < \lambda$, let us define $\xi = \frac{1-\mu}{\mu}$. Obviously, $\frac{1}{2} < \xi < 1$. Set $b = 0.3393$.

Let us assume that there are μ and λ satisfying the above relationships and also satisfy:

$$\frac{\lambda\mu}{\lambda+\mu} > b \qquad \text{and} \qquad \frac{1-\mu}{1+\lambda-\mu} > b.$$

Expressing these inequalities in terms of ξ and λ we get:

$$\frac{\xi(1-b)}{b(1+\xi)} > \lambda > \frac{b}{1-b(1+\xi)}.$$

Since $b < \frac{1}{2}$, the above inequality is equivalent to:

$$\xi(1-b)(1-b(1+\xi)) - b^2(1+\xi) > 0 \quad \Leftrightarrow \quad -\xi^2 b(1-b) + \xi(1-2b) - b^2 > 0.$$

It can be verified by direct calculation that the discriminant of the above quadratic is 0 for $b = 0.3393$ and the inequality becomes $-b(1-b)\left(\xi - \frac{1-2b}{2b(1-b)}\right)^2 > 0$, a contradiction.

Actually ,the constant b is the smallest real solution of the equation

$$4b(1-b)(1+b^2) = 1.$$

The bound is attained at $\mu = 0.582523$ and $\lambda = 0.81281$.

A.3 Appendix A.3

Let (x, y) be the current pair of strategies obtained during the descent procedure, for which none of the conditions of step 3 of the algorithm is satisfied. Then, we should have:

$$V(x, y) < b < f(x, y)$$

Indeed, since $V(x, y)$ is always $\leq \min\{\rho\lambda, (1-\rho)\mu\}$, if $V(x, y)$ was $\geq b$ we would also have $f(\tilde{x}, \tilde{y}) \leq b$, since $f(\tilde{x}, \tilde{y})$ is always $\leq \frac{1-\min(\lambda,\mu)}{1+\max(\lambda,\mu)-\min(\lambda,\mu)}$ and b is the maximum value for $\min\{\rho\lambda, (1-\rho)\mu, \frac{1-\min(\lambda,\mu)}{1+\max(\lambda,\mu)-\min(\lambda,\mu)}\}$ as proven before.

We also have:

$$f(x+\epsilon(x'-x), y+\epsilon(y'-y)) - f(x, y) = \epsilon(V(x, y) - f(x, y)) + \max\{\Lambda f_R - \epsilon^2 H f_R, \Lambda f_C - \epsilon^2 H f_C\}$$

where Hf_R, Hf_C, Λf_R, Λf_C are as defined in appendix A.1.

The quantitites Λf_R, Λf_C are both piecewise linear convex functions of ϵ and are equal to 0 for $\epsilon \leq \epsilon^\star$, where $\epsilon^\star$ is given by $\epsilon^\star = min\{\epsilon_1^\star, \epsilon_2^\star, 1\}$ and $\epsilon_1^\star$ is the minimum over $i \in \overline{S}_R(y)$ of:

$$\frac{\max(Ry) - (Ry)_i}{\max(Ry) - (Ry)_i + (Ry')_i - \max_{S_R(y)}(Ry')} \text{ , for some } i \in \overline{S}_R(y)$$

and $\epsilon_2^\star$ is the minimum over $j \in \overline{S}_C(x)$ of :

$$\frac{\max(C^Tx) - (C^Tx)_j}{\max(C^Tx) - (C^Tx)_j + (C^Tx')_j - \max_{S_C(x)}(C^Tx')} \text{ , for some } j \in \overline{S}_C(x)$$

It is pointed out that the terms $\max(Ry) - (Ry)_i$ for $i \in \overline{S}_R(y)$ are always positive and at least one of them is $\geq f(x,y)$, since $f(x,y) = \sum_{i \in \overline{S}_R(y)} x_i(\max(Ry) - (Ry)_i)$. The same is true for the terms $\max(C^Tx) - (C^Tx)_j$ for $j \in \overline{S}_C(x)$. Furthermore, the above expressions for $\epsilon^\star$ are active only for those indices $i \in (1, m)$, $j \in (1, n)$, $i \in \overline{S}_R(y)$, $j \in \overline{S}_C(x)$ for which $(Ry')_i - \max_{S_R(y)}(Ry') \geq 0$ and $(C^Tx')_j - \max_{S_C(x)}(C^Tx') \geq 0$. If no such indices exist for the (x', y') pair of strategies, then the corresponding value of ϵ should be equal to 1.

The quantities Hf_R, Hf_C appearing in the quadratic terms of ϵ, are both bounded (in absolute value) by 2. So, the minimum possible descent that can be achieved is given by the following relationship :

$$f(x + \epsilon(x' - x), y + \epsilon(y' - y)) - f(x, y) = \epsilon(V(x, y) - f(x, y))$$
$$-\epsilon^2 \min(Hf_R, Hf_C) \leq \epsilon\,(V(x, y) - f(x, y)) + 2\epsilon^2,\ 0 \leq \epsilon \leq \epsilon^\star$$

Defining the new value of f as f_{new} and dropping the arguments (for simplicity) we get

$$f_{new} - b \leq (1 - \epsilon)(f - b) + \epsilon(V - b) + 2\epsilon^2$$

Minimizing with respect to ϵ, for $\epsilon \leq \epsilon^\star$, we get:

$$f_{new} - b \leq (f - b)\left(1 - \frac{b - V}{4}\right) - \frac{(f - b)^2 + (b - V)^2}{8},\ \text{if } \epsilon^\star \geq \frac{f - V}{4}$$

$$f_{new} - b \leq (f - b)(1 - \epsilon^\star) - (b - V)\epsilon^\star + 2{\epsilon^\star}^2,\ \text{if } \epsilon^\star < \frac{f - V}{4}$$

In the first case above, we obtain a significent reduction of $f_{new} - b$ if $\epsilon^\star$ is larger than $\frac{f-V}{4}$. In the second case, the reduction depends on how small $\epsilon^\star$ is.

If the value of $\epsilon^\star$ is small , then there is an index $i^\star \in \overline{S}_R(y)$ or an index $j^\star \in \overline{S}_C(x)$ such that the entry $(Ry)_{i^\star}$ or $(C^Tx)_{j^\star}$, is close to the maximum support of the vector Ry , or C^Tx. Such entries can be incorporated into the sets $S_R(y), S_C(x)$ by appropriately augmenting the supports of the vectors w, z in the formulation of the linear minimax problem described in Section 3.

Furthermore , it is not possible to encounter more than $m + n - 2$ such steps in a row without meeting one of the termination conditions of the algorithm , particularly the condition $f(x, y) \leq b$, since , if all the differences of the form $max(Ry) - (Ry)_i, i \in \overline{S}_R(y)$ are small , then $f(x, y)$ is also small.

From the above, we deduce that a termination condition of the algorithm can be approached as closely as desired , in polynomial time.

In fact, a detailed analysis of the number q of the steps needed , for any $\delta > 0$, in order to approximate a stationary point sufficiently close and find an $0.3393 + \delta$-approximate equilibrium, can show that q is $O(\frac{1}{\delta^2})$. A linear programming problem has to be solved in each such step.