

Computing Good Nash Equilibria in Graphical Games ^{*}

Edith Elkind
Hebrew University of
Jerusalem, Israel, and
University of Southampton,
Southampton, SO17 1BJ, U.K.

Leslie Ann Goldberg
University of Liverpool
Liverpool L69 3BX, U.K.

Paul Goldberg
University of Liverpool
Liverpool L69 3BX, U.K.

ABSTRACT

This paper addresses the problem of fair equilibrium selection in graphical games. Our approach is based on the data structure called the *best response policy*, which was proposed by Kearns et al. [13] as a way to represent all Nash equilibria of a graphical game. In [9], it was shown that the best response policy has polynomial size as long as the underlying graph is a path. In this paper, we show that if the underlying graph is a bounded-degree tree and the best response policy has polynomial size then there is an efficient algorithm which constructs a Nash equilibrium that guarantees certain payoffs to all participants. Another attractive solution concept is a Nash equilibrium that maximizes the social welfare. We show that, while exactly computing the latter is infeasible (we prove that solving this problem may involve algebraic numbers of an arbitrarily high degree), there exists an FPTAS for finding such an equilibrium as long as the best response policy has polynomial size. These two algorithms can be combined to produce Nash equilibria that satisfy various fairness criteria.

Categories and Subject Descriptors

F.2 [Theory of Computation]: Analysis of Algorithms and Problem Complexity; J.4 [Computer Applications]: Social and Behavioral Sciences—*economics*

General Terms

Algorithms, Economics, Theory

Keywords

Graphical games, Nash equilibrium, approximation scheme

1. INTRODUCTION

In a large community of agents, an agent's behavior is not likely to have a direct effect on most other agents: rather, it is just the

^{*}Supported by the EPSRC research grants "Algorithmics of Network-sharing Games" and "Discontinuous Behaviour in the Complexity of randomized Algorithms".

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EC'07, June 11–15, 2007, San Diego, California, USA.

Copyright 2007 ACM 978-1-59593-653-0/07/0006 ...\$5.00.

agents who are close enough to him that will be affected. However, as these agents respond by adapting their behavior, more agents will feel the consequences and eventually the choices made by a single agent will propagate throughout the entire community.

This is the intuition behind *graphical games*, which were introduced by Kearns, Littman and Singh in [13] as a compact representation scheme for games with many players. In an n -player graphical game, each player is associated with a vertex of an underlying graph G , and the payoffs of each player depend on his action as well as on the actions of his neighbors in the graph. If the maximum degree of G is Δ , and each player has two actions available to him, then the game can be represented using $n2^{\Delta+1}$ numbers. In contrast, we need $n2^n$ numbers to represent a general n -player 2-action game, which is only practical for small values of n . For graphical games with constant Δ , the size of the game is linear in n .

One of the most natural problems for a graphical game is that of finding a Nash equilibrium, the existence of which follows from Nash's celebrated theorem (as graphical games are just a special case of n -player games). The first attempt to tackle this problem was made in [13], where the authors consider graphical games with two actions per player in which the underlying graph is a bounded-degree tree. They propose a generic algorithm for finding Nash equilibria that can be specialized in two ways: an exponential-time algorithm for finding an (exact) Nash equilibrium, and a fully polynomial time approximation scheme (FPTAS) for finding an approximation to a Nash equilibrium. For any $\epsilon > 0$ this algorithm outputs an ϵ -Nash equilibrium, which is a strategy profile in which no player can improve his payoff by more than ϵ by unilaterally changing his strategy.

While ϵ -Nash equilibria are often easier to compute than exact Nash equilibria, this solution concept has several drawbacks. First, the players may be sensitive to a small loss in payoffs, so the strategy profile that is an ϵ -Nash equilibrium will not be stable. This will be the case even if there is only a small subset of players who are extremely price-sensitive, and for a large population of players it may be difficult to choose a value of ϵ that will satisfy everyone. Second, the strategy profiles that are close to being Nash equilibria may be much better with respect to the properties under consideration than exact Nash equilibria. Therefore, the (approximation to the) value of the best solution that corresponds to an ϵ -Nash equilibrium may not be indicative of what can be achieved under an exact Nash equilibrium. This is especially important if the purpose of the approximate solution is to provide a good benchmark for a system of selfish agents, as the benchmark implied by an ϵ -Nash equilibrium may be unrealistic. For these reasons, in this paper we focus on the problem of computing exact Nash equilibria.

Building on ideas of [14], Elkind et al. [9] showed how to find an (exact) Nash equilibrium in polynomial time when the underlying

graph has degree 2 (that is, when the graph is a collection of paths and cycles). By contrast, finding a Nash equilibrium in a general degree-bounded graph appears to be computationally intractable: it has been shown (see [5, 12, 7]) to be complete for the complexity class PPAD. [9] extends this hardness result to the case in which the underlying graph has bounded pathwidth.

A graphical game may not have a unique Nash equilibrium, indeed it may have exponentially many. Moreover, some Nash equilibria are more desirable than others. Rather than having an algorithm which merely finds *some* Nash equilibrium, we would like to have algorithms for finding Nash equilibria with various socially-desirable properties, such as maximizing overall payoff or distributing profit fairly.

A useful property of the data structure of [13] is that it simultaneously represents the set of *all* Nash equilibria of the underlying game. If this representation has polynomial size (as is the case for paths, as shown in [9]), one may hope to extract from it a Nash equilibrium with the desired properties. In fact, in [13] the authors mention that this is indeed possible if one is interested in finding an (approximate) ϵ -Nash equilibrium. The goal of this paper is to extend this to exact Nash equilibria.

1.1 Our Results

In this paper, we study n -player 2-action graphical games on bounded-degree trees for which the data structure of [13] has size $\text{poly}(n)$. We focus on the problem of finding exact Nash equilibria with certain socially-desirable properties. In particular, we show how to find a Nash equilibrium that (nearly) maximizes the social welfare, i.e., the sum of the players' payoffs, and we show how to find a Nash equilibrium that (nearly) satisfies prescribed payoff bounds for all players.

Graphical games on bounded-degree trees have a simple algebraic structure. One attractive feature, which follows from [13], is that every such game has a Nash equilibrium in which the strategy of every player is a rational number. Section 3 studies the algebraic structure of those Nash equilibria that maximize social welfare. We show (Theorems 1 and 2) that, surprisingly, the set of Nash equilibria that maximize social welfare is more complex. In fact, for any algebraic number $\alpha \in [0, 1]$ with degree at most n , we exhibit a graphical game on a path of length $O(n)$ such that, in the unique social welfare-maximizing Nash equilibrium of this game, one of the players plays the mixed strategy α .¹ This result shows that it may be difficult to represent an optimal Nash equilibrium. It seems to be a novel feature of the setting we consider here, that an optimal Nash equilibrium is hard to represent, in a situation where it is easy to find and represent a Nash equilibrium.

As the social welfare-maximizing Nash equilibrium may be hard to represent efficiently, we have to settle for an approximation. However, the crucial difference between our approach and that of previous papers [13, 16, 19] is that we require our algorithm to output an exact Nash equilibrium, though not necessarily the optimal one with respect to our criteria. In Section 4, we describe an algorithm that satisfies this requirement. Namely, we propose an algorithm that for any $\epsilon > 0$ finds a Nash equilibrium whose total payoff is within ϵ of optimal. It runs in polynomial time (Theorem 3,4) for any graphical game on a bounded-degree tree for which the data structure proposed by [13] (the so-called *best response policy*, defined below) is of size $\text{poly}(n)$ (note that, as shown in [9], this is always the case when the underlying graph is a path). More pre-

cisely, the running time of our algorithm is polynomial in n , P_{\max} , and $1/\epsilon$, where P_{\max} is the maximum absolute value of an entry of a payoff matrix, i.e., it is a pseudopolynomial algorithm, though it is fully polynomial with respect to ϵ . We show (Section 4.1) that under some restrictions on the payoff matrices, the algorithm can be transformed into a (truly) polynomial-time algorithm that outputs a Nash equilibrium whose total payoff is within a $1 - \epsilon$ factor from the optimal.

In Section 5, we consider the problem of finding a Nash equilibrium in which the expected payoff of each player V_i exceeds a prescribed threshold T_i . Using the idea from Section 4 we give (Theorem 5) a fully polynomial time approximation scheme for this problem. The running time of the algorithm is bounded by a polynomial in n , P_{\max} , and ϵ . If the instance has a Nash equilibrium satisfying the prescribed thresholds then the algorithm constructs a Nash equilibrium in which the expected payoff of each player V_i is at least $T_i - \epsilon$.

In Section 6, we introduce other natural criteria for selecting a “good” Nash equilibrium and we show that the algorithms described in the two previous sections can be used as building blocks in finding Nash equilibria that satisfy these criteria. In particular, in Section 6.1 we show how to find a Nash equilibrium that approximates the maximum social welfare, while guaranteeing that each individual payoff is close to a prescribed threshold. In Section 6.2 we show how to find a Nash equilibrium that (nearly) maximizes the minimum individual payoff. Finally, in Section 6.3 we show how to find a Nash equilibrium in which the individual payoffs of the players are close to each other.

1.2 Related Work

Our approximation scheme (Theorem 3 and Theorem 4) shows a contrast between the games that we study and two-player n -action games, for which the corresponding problems are usually intractable. For two-player n -action games, the problem of finding Nash equilibria with special properties is typically NP-hard. In particular, this is the case for Nash equilibria that maximize the social welfare [11, 6]. Moreover, it is likely to be intractable even to approximate such equilibria. In particular, Chen, Deng and Teng [4] show that there exists some ϵ , inverse polynomial in n , for which computing an ϵ -Nash equilibrium in 2-player games with n actions per player is PPAD-complete.

Lipton and Markakis [15] study the algebraic properties of Nash equilibria, and point out that standard quantifier elimination algorithms can be used to solve them. Note that these algorithms are not polynomial-time in general. The games we study in this paper have polynomial-time computable Nash equilibria in which all mixed strategies are rational numbers, but an optimal Nash equilibrium may necessarily include mixed strategies with high algebraic degree.

A *correlated equilibrium* (CE) (introduced by Aumann [2]) is a distribution over vectors of players' actions with the property that if any player is told his own action (the value of his own component) from a vector generated by that distribution, then he cannot increase his expected payoff by changing his action. Any Nash equilibrium is a CE but the converse does not hold in general. In contrast with Nash equilibria, correlated equilibria can be found for low-degree graphical games (as well as other classes of concisely-represented multiplayer games) in polynomial time [17]. But, for graphical games it is NP-hard to find a correlated equilibrium that maximizes total payoff [18]. However, the NP-hardness results apply to more general games than the one we consider here, in particular the graphs are not trees. From [2] it is also known that there exist 2-player, 2-action games for which the expected total payoff

¹A related result in a different context was obtained by Datta [8], who shows that n -player 2-action games are *universal* in the sense that any real algebraic variety can be represented as the set of totally mixed Nash equilibria of such games.

of the best correlated equilibrium is higher than the best Nash equilibrium, and we discuss this issue further in Section 7.

2. PRELIMINARIES AND NOTATION

We consider graphical games in which the underlying graph G is an n -vertex tree, in which each vertex has at most Δ children. Each vertex has two actions, which are denoted by 0 and 1. A mixed strategy of a player V is represented as a single number $v \in [0, 1]$, which denotes the probability that V selects action 1.

For the purposes of the algorithm, the tree is rooted arbitrarily. For convenience, we assume without loss of generality that the root has a single child, and that its payoff is independent of the action chosen by the child. This can be achieved by first choosing an arbitrary root of the tree, and then adding a dummy “parent” of this root, giving the new parent a constant payoff function, e.g., 0.

Given an edge (V, W) of the tree G , and a mixed strategy w for W , let $G_{(V,W),W=w}$ be the instance obtained from G by (1) deleting all nodes Z which are separated from V by W (i.e., all nodes Z such that the path from Z to V passes through W), and (2) restricting the instance so that W is required to play mixed strategy w .

Definition 1. Suppose that (V, W) is an edge of the tree, that v is a mixed strategy for V and that w is a mixed strategy for W . We say that v is a *potential best response* to w (denoted by $v \in \text{pbr}_V(w)$) if there is an equilibrium in the instance $G_{(V,W),W=w}$ in which V has mixed strategy v . We define the *best response policy* for V , given W , as $\mathcal{B}(W, V) = \{(w, v) \mid v \in \text{pbr}_V(w), w \in [0, 1]\}$.

The upstream pass of the generic algorithm of [13] considers every node V (other than the root) and computes the best response policy for V given its parent. With the above assumptions about the root, the downstream pass is straightforward. The root selects a mixed strategy w for the root W and a mixed strategy $v \in \mathcal{B}(W, V)$ for each child V of W . It instructs each child V to play v . The remainder of the downward pass is recursive. When a node V is instructed by its parent to adopt mixed strategy v , it does the following for each child U — It finds a pair $(v, u) \in \mathcal{B}(V, U)$ (with the same v value that it was given by its parent) and instructs U to play u .

The best response policy for a vertex U given its parent V can be represented as a union of rectangles, where a *rectangle* is defined by a pair of closed intervals (I_V, I_U) and consists of all points in $I_V \times I_U$; it may be the case that one or both of the intervals I_V and I_U consists of a single point. In order to perform computations on $\mathcal{B}(V, U)$, and to bound the number of rectangles, [9] used the notion of an *event point*, which is defined as follows. For any set $A \subseteq [0, 1]^2$ that is represented as a union of a finite number of rectangles, we say that a point $u \in [0, 1]$ on the U -axis is a *U -event point* of A if $u = 0$ or $u = 1$ or the representation of A contains a rectangle of the form $I_V \times I_U$ and u is an endpoint of I_U ; V -event points are defined similarly.

For many games considered in this paper, the underlying graph is an n -vertex path, i.e., a graph $G = (V, E)$ with $V = \{V_1, \dots, V_n\}$ and $E = \{(V_1, V_2), \dots, (V_{n-1}, V_n)\}$. In [9], it was shown that for such games, the best response policy has only polynomially-many rectangles. The proof that the number of rectangles in $\mathcal{B}(V_{j+1}, V_j)$ is polynomial proceeds by first showing that the number of event points in $\mathcal{B}(V_{j+1}, V_j)$ cannot exceed the number of event points in $\mathcal{B}(V_j, V_{j-1})$ by more than 2, and using this fact to bound the number of rectangles in $\mathcal{B}(V_{j+1}, V_j)$.

Let $P^0(V)$ and $P^1(V)$ be the expected payoffs to V when it plays 0 and 1, respectively. Both $P^0(V)$ and $P^1(V)$ are multilinear

functions of the strategies of V 's neighbors. In what follows, we will frequently use the following simple observation.

CLAIM 1. For a vertex V with a single child U and parent W , given any $A, B, C, D \in \mathbb{Q}$, $A', B', C', D' \in \mathbb{Q}$, one can select the payoffs to V so that $P^0(V) = Auw + Bu + Cw + D$, $P^1(V) = A'uw + B'u + C'w + D'$. Moreover, if all $A, B, C, D, A', B', C', D'$ are integer, the payoffs to V are integer as well.

PROOF. We will give the proof for $P^0(V)$; the proof for $P^1(V)$ is similar. For $i, j = 0, 1$, let P_{ij} be the payoff to V when U plays i , V plays 0 and W plays j . We have $P^0(V) = P_{00}(1 - u)(1 - w) + P_{10}u(1 - w) + P_{01}(1 - u)w + P_{11}uw$. We have to select the values of P_{ij} so that $P_{00} - P_{10} - P_{01} + P_{11} = A$, $-P_{00} + P_{10} = B$, $-P_{00} + P_{01} = C$, $P_{00} = D$. It is easy to see that the unique solution is given by $P_{00} = D$, $P_{01} = C + D$, $P_{10} = B + D$, $P_{11} = A + B + C + D$. \square

The input to all algorithms considered in this paper includes the payoff matrices for each player. We assume that all elements of these matrices are integer. Let P_{\max} be the greatest absolute value of any element of any payoff matrix. Then the input consists of at most $n2^{\Delta+1}$ numbers, each of which can be represented using $\lceil \log P_{\max} \rceil$ bits.

3. NASH EQUILIBRIA THAT MAXIMIZE THE SOCIAL WELFARE: SOLUTIONS IN $\mathbb{R} \setminus \mathbb{Q}$

From the point of view of social welfare, the best Nash equilibrium is the one that maximizes the sum of the players' expected payoffs. Unfortunately, it turns out that computing such a strategy profile exactly is not possible: in this section, we show that even if all players' payoffs are integers, the strategy profile that maximizes the total payoff may have irrational coordinates; moreover, it may involve algebraic numbers of an arbitrary degree.

3.1 Warm-up: quadratic irrationalities

We start by providing an example of a graphical game on a path of length 3 with integer payoffs such that in the Nash equilibrium that maximizes the total payoff, one of the players has a strategy in $\mathbb{R} \setminus \mathbb{Q}$. In the next subsection, we will extend this example to algebraic numbers of arbitrary degree n ; to do so, we have to consider paths of length $O(n)$.

THEOREM 1. There exists an integer-payoff graphical game G on a 3-vertex path UVW such that, in any Nash equilibrium of G that maximizes social welfare, the strategy, u , of the player U and the total payoff, p , satisfy $u, p \in \mathbb{R} \setminus \mathbb{Q}$.

PROOF. The payoffs to the players in G are specified as follows. The payoff to U is identically 0, i.e., $P^0(U) = P^1(U) = 0$. Using Claim 1, we select the payoffs to V so that $P^0(V) = -uw + 3w$ and $P^1(V) = P^0(V) + w(u + 2) - (u + 1)$, where u and w are the (mixed) strategies of U and W , respectively. It follows that V is indifferent between playing 0 and 1 if and only if $w = f(u) = \frac{u+1}{u+2}$. Observe that for any $u \in [0, 1]$ we have $f(u) \in [0, 1]$. The payoff to W is 0 if it selects the same action as V and 1 otherwise.

CLAIM 2. All Nash equilibria of the game G are of the form $(u, 1/2, f(u))$. That is, in any Nash equilibrium, V plays $v = 1/2$ and W plays $w = f(u)$. Moreover, for any value of u , the vector of strategies $(u, 1/2, f(u))$ constitutes a Nash equilibrium.

PROOF. It is easy to check that for any $u \in [0, 1]$, the vector $(u, 1/2, f(u))$ is a Nash equilibrium. Indeed, U is content to play

any mixed strategy u no matter what V and W do. Furthermore, V is indifferent between 0 and 1 as long as $w = f(u)$, so it can play $1/2$. Finally, if V plays 0 and 1 with equal probability, W is indifferent between 0 and 1, so it can play $f(u)$.

Conversely, suppose that $v > 1/2$. Then W strictly prefers to play 0, i.e., $w = 0$. Then for V we have $P^1(V) = P^0(V) - (u + 1)$, i.e., $P^1(V) < P^0(V)$, which implies $v = 0$, a contradiction. Similarly, if $v < 1/2$, player W prefers to play 1, so we have $w = 1$. Hence, $P^1(V) = P^0(V) + (u + 2) - (u + 1)$, i.e., $P^1(V) > P^0(V)$, which implies $v = 1$, a contradiction. Finally, if $v = 1/2$, but $w \neq f(u)$, player V is not indifferent between 0 and 1, so he would deviate from playing $1/2$. This completes the proof of Claim 2. \square

By Claim 2, the total payoff in any Nash equilibrium of this game is a function of u . More specifically, the payoff to U is 0, the payoff to V is $-uf(u) + 3f(u)$, and the payoff to W is $1/2$. Therefore, the Nash equilibrium with the maximum total payoff corresponds to the value of u that maximizes

$$g(u) = -u \frac{(u+1)}{u+2} + 3 \frac{u+1}{u+2} = -\frac{(u-3)(u+1)}{u+2}.$$

To find extrema of $g(u)$, we compute $h(u) = -\frac{d}{du}g(u)$. We have

$$h(u) = \frac{(2u-2)(u+2) - (u-3)(u+1)}{(u+2)^2} = \frac{u^2 + 4u - 1}{(u+2)^2}.$$

Hence, $h(u) = 0$ if and only if $u \in \{-2 + \sqrt{5}, -2 - \sqrt{5}\}$. Note that $-2 + \sqrt{5} \in [0, 1]$.

The function $g(u)$ changes sign at -2 , -1 , and 3 . We have $g(u) < 0$ for $g > 3$, $g(u) > 0$ for $u < -2$, so the extremum of $g(u)$ that lies between 1 and 3, i.e., $u = -2 + \sqrt{5}$, is a local maximum. We conclude that the social welfare-maximizing Nash equilibrium for this game is given by the vector of strategies $(-2 + \sqrt{5}, 1/2, (5 - \sqrt{5})/5)$. The respective total payoff is

$$0 - \frac{(\sqrt{5}-5)(\sqrt{5}-1)}{\sqrt{5}} + \frac{1}{2} = 13/2 - 2\sqrt{5}.$$

This concludes the proof of Theorem 1.

3.2 Strategies of arbitrary degree

We have shown that in the social welfare-maximizing Nash equilibrium, some players' strategies can be quadratic irrationalities, and so can the total payoff. In this subsection, we will extend this result to show that we can construct an integer-payoff graphical game on a path whose social welfare-maximizing Nash equilibrium involves arbitrary algebraic numbers in $[0, 1]$.

THEOREM 2. *For any degree- n algebraic number $\alpha \in [0, 1]$, there exists an integer payoff graphical game on a path of length $O(n)$ such that, in all social welfare-maximizing Nash equilibria of this game, one of the players plays α .*

PROOF. Our proof consists of two steps. First, we construct a rational expression $R(x)$ and a segment $[x', x'']$ such that $x', x'' \in \mathbb{Q}$ and α is the only maximum of $R(x)$ on $[x', x'']$. Second, we construct a graphical game whose Nash equilibria can be parameterized by $u \in [x', x'']$, so that at the equilibrium that corresponds to u the total payoff is $R(u)$ and, moreover, some player's strategy is u . It follows that to achieve the payoff-maximizing Nash equilibrium, this player has to play α . The details follow.

LEMMA 1. *Given an algebraic number $\alpha \in [0, 1]$, $\deg(\alpha) = n$, there exist $K_2, \dots, K_{2n+2} \in \mathbb{Q}$ and $x', x'' \in (0, 1) \cap \mathbb{Q}$ such*

that α is the only maximum of

$$R(x) = \frac{K_2}{x+2} + \dots + \frac{K_{2n+2}}{x+2n+2}$$

on $[x', x'']$.

PROOF. Let $P(x)$ be the minimal polynomial of α , i.e., a polynomial of degree n with rational coefficients whose leading coefficient is 1 such that $P(\alpha) = 0$. Let $A = \{\alpha_1, \dots, \alpha_n\}$ be the set of all roots of $P(x)$. Consider the polynomial $Q_1(x) = -P^2(x)$. It has the same roots as $P(x)$, and moreover, for any $x \notin A$ we have $Q_1(x) < 0$. Hence, A is the set of all maxima of $Q_1(x)$. Now, set $R(x) = \frac{Q_1(x)}{(x+2)\dots(x+2n+1)(x+2n+2)}$. Observe that $R(x) \leq 0$ for all $x \in [0, 1]$ and $R(x) = 0$ if and only if $Q_1(x) = 0$. Hence, the set A is also the set of all maxima of $R(x)$ on $[0, 1]$.

Let $d = \min\{|\alpha_i - \alpha| \mid \alpha_i \in A, \alpha_i \neq \alpha\}$, and set $\alpha' = \max\{\alpha - d/2, 0\}$, $\alpha'' = \min\{\alpha + d/2, 1\}$. Clearly, α is the only zero (and hence, the only maximum) of $R(x)$ on $[\alpha', \alpha'']$. Let x' and x'' be some rational numbers in (α', α) and (α, α'') , respectively; note that by excluding the endpoints of the intervals we ensure that $x', x'' \neq 0, 1$. As $[x', x''] \subset [\alpha', \alpha'']$, we have that α is the only maximum of $R(x)$ on $[x', x'']$.

As $R(x)$ is a proper rational expression and all roots of its denominator are simple, by partial fraction decomposition theorem, $R(x)$ can be represented as

$$R(x) = \frac{K_2}{x+2} + \dots + \frac{K_{2n+2}}{x+2n+2},$$

where K_2, \dots, K_{2n+2} are rational numbers. \square

Consider a graphical game on the path

$$U_{-1}V_{-1}U_0V_0U_1V_1\dots U_{k-1}V_{k-1}U_k,$$

where $k = 2n + 2$. Intuitively, we want each triple (U_{i-1}, V_{i-1}, U_i) to behave similarly to the players U, V , and W from the game described in the previous subsection. More precisely, we define the payoffs to the players in the following way.

- The payoff to U_{-1} is 0 no matter what everyone else does.
- The expected payoff to V_{-1} is 0 if it plays 0 and $u_0 - (x'' - x')u_{-1} - x'$ if it plays 1, where u_0 and u_{-1} are the strategies of U_0 and U_{-1} , respectively.
- The expected payoff to V_0 is 0 if it plays 0 and $u_1(u_0 + 1) - u_0$ if it plays 1, where u_0 and u_1 are the strategies of U_0 and U_1 , respectively.
- For each $i = 1, \dots, k-1$, the expected payoff to V_i when it plays 0 is $P^0(V_i) = A_i u_i u_{i+1} - A_i u_{i+1}$, and the expected payoff to V_i when it plays 1 is $P^1(V_i) = P^0(V_i) + u_{i+1}(2 - u_i) - 1$, where $A_i = -K_{i+1}$ and u_{i+1} and u_i are the strategies of U_{i+1} and U_i , respectively.
- For each $i = 0, \dots, k$, the payoff to U_i does not depend on V_i and is 1 if U_i and V_{i-1} select different actions and 0 otherwise.

We will now characterize the Nash equilibria of this game using a sequence of claims.

CLAIM 3. *In all Nash equilibria of this game V_{-1} plays $1/2$, and the strategies of u_{-1} and u_0 satisfy $u_0 = (x'' - x')u_{-1} + x'$. Consequently, in all Nash equilibria we have $u_0 \in [x', x'']$.*

PROOF. The proof is similar to that of Claim 2. Let $f(u_{-1}) = (x'' - x')u_{-1} + x'$. Clearly, the player V_{-1} is indifferent between playing 0 and 1 if and only if $u_0 = f(u_{-1})$. Suppose that $v_{-1} < 1/2$. Then U_0 strictly prefers to play 1, i.e., $u_0 = 1$, so we have

$$P^1(V_{-1}) = P^0(V_{-1}) + 1 - (x'' - x')u_{-1} - x'.$$

As

$$1 - x'' \leq 1 - (x'' - x')u_{-1} - x' \leq 1 - x'$$

for $u_{-1} \in [0, 1]$ and $x'' < 1$, we have $P^1(V_{-1}) > P^0(V_{-1})$, so V_{-1} prefers to play 1, a contradiction. Similarly, if $v_{-1} > 1/2$, the player U_0 strictly prefers to play 0, i.e., $u_0 = 0$, so we have

$$P^1(V_{-1}) = P^0(V_{-1}) - (x'' - x')u_{-1} - x'.$$

As $x' < x''$, $x' > 0$, we have $P^1(V_{-1}) < P^0(V_{-1})$, so V_{-1} prefers to play 0, a contradiction. Finally, if V_{-1} plays $1/2$, but $u_0 \neq f(u_{-1})$, player V_{-1} is not indifferent between 0 and 1, so he would deviate from playing $1/2$.

Also, note that $f(0) = x'$, $f(1) = x''$, and, moreover, $f(u_{-1}) \in [x', x'']$ if and only if $u_{-1} \in [0, 1]$. Hence, in all Nash equilibria of this game we have $u_0 \in [x', x'']$. \square

CLAIM 4. In all Nash equilibria of this game for each $i = 0, \dots, k-1$, we have $v_i = 1/2$, and the strategies of the players U_i and U_{i+1} satisfy $u_{i+1} = f_i(u_i)$, where $f_0(u) = u/(u+1)$ and $f_i(u) = 1/(2-u)$ for $i > 0$.

PROOF. The proof of this claim is also similar to that of Claim 2. We use induction on i to prove that the statement of the claim is true and, additionally, $u_i \neq 1$ for $i > 0$.

For the base case $i = 0$, note that $u_0 \neq 0$ by the previous claim (recall that x', x'' are selected so that $x', x'' \neq 0, 1$) and consider the triple (U_0, V_0, U_1) . Let v_0 be the strategy of V_0 . First, suppose that $v_0 > 1/2$. Then U_1 strictly prefers to play 0, i.e., $u_1 = 0$. Then for V_0 we have $P^1(V_0) = P^0(V_0) - u_0$. As $u_0 \neq 0$, we have $P^1(V_0) < P^0(V_0)$, which implies $v_1 = 0$, a contradiction. Similarly, if $v_0 < 1/2$, player U_1 prefers to play 1, so we have $u_1 = 1$. Hence, $P^1(V_0) = P^0(V_0) + 1$. It follows that $P^1(V_0) > P^0(V_0)$, which implies $v_0 = 1$, a contradiction. Finally, if $v_0 = 1/2$, but $u_1 \neq u_0/(u_0+1)$, player V_0 is not indifferent between 0 and 1, so he would deviate from playing $1/2$. Moreover, as $u_1 = u_0/(u_0+1)$ and $u_0 \in [0, 1]$, we have $u_1 \neq 1$.

The argument for the inductive step is similar. Namely, suppose that the statement is proved for all $i' < i$ and consider the triple (U_i, V_i, U_{i+1}) .

Let v_i be the strategy of V_i . First, suppose that $v_i > 1/2$. Then U_{i+1} strictly prefers to play 0, i.e., $u_{i+1} = 0$. Then for V_i we have $P^1(V_i) = P^0(V_i) - 1$, i.e., $P^1(V_i) < P^0(V_i)$, which implies $v_i = 0$, a contradiction. Similarly, if $v_i < 1/2$, player U_{i+1} prefers to play 1, so we have $u_{i+1} = 1$. Hence, $P^1(V_i) = P^0(V_i) + 1 - u_i$. By inductive hypothesis, we have $u_i < 1$. Consequently, $P^1(V_i) > P^0(V_i)$, which implies $v_i = 1$, a contradiction. Finally, if $v_i = 1/2$, but $u_{i+1} \neq 1/(2-u_i)$, player V_i is not indifferent between 0 and 1, so he would deviate from playing $1/2$. Moreover, as $u_{i+1} = 1/(2-u_i)$ and $u_i < 1$, we have $u_{i+1} < 1$. \square

CLAIM 5. Any strategy profile of the form

$$(u_{-1}, 1/2, u_0, 1/2, u_1, 1/2, \dots, u_{k-1}, 1/2, u_k),$$

where $u_{-1} \in [0, 1]$, $u_0 = (x'' - x')u_{-1} + x'$, $u_1 = u_0/(u_0+1)$, and $u_{i+1} = 1/(2-u_i)$ for $i \geq 1$ constitutes a Nash equilibrium.

PROOF. First, the player U_{-1} 's payoffs do not depend on other players' actions, so he is free to play any strategy in $[0, 1]$. As long

as $u_0 = (x'' - x')u_{-1} + x'$, player V_{-1} is indifferent between 0 and 1, so he is content to play $1/2$; a similar argument applies to players V_0, \dots, V_{k-1} . Finally, for each $i = 0, \dots, k$, the payoffs of player U_i only depend on the strategy of player V_{i-1} . In particular, as long as $v_{i-1} = 1/2$, player U_i is indifferent between playing 0 and 1, so he can play any mixed strategy $u_i \in [0, 1]$. To complete the proof, note that $(x'' - x')u_{-1} + x' \in [0, 1]$ for all $u_{-1} \in [0, 1]$, $u_0/(u_0+1) \in [0, 1]$ for all $u_0 \in [0, 1]$, and $1/(2-u_i) \in [0, 1]$ for all $u_i \in [0, 1]$, so we have $u_i \in [0, 1]$ for all $i = 0, \dots, k$. \square

Now, let us compute the total payoff under a strategy profile of the form given in Claim 5. The payoff to U_{-1} is 0, and the expected payoff to each of the U_i , $i = 0, \dots, k$, is $1/2$. The expected payoffs to V_{-1} and V_0 are 0. Finally, for any $i = 1, \dots, k-1$, the expected payoff to V_i is $T_i = A_i u_i u_{i+1} - A_i u_{i+1}$. It follows that to find a Nash equilibrium with the highest total payoff, we have to maximize $\sum_{i=1}^{k-1} T_i$ subject to conditions $u_{-1} \in [0, 1]$, $u_0 = (x'' - x')u_{-1} + x'$, $u_1 = u_0/(u_0+1)$, and $u_{i+1} = 1/(2-u_i)$ for $i = 1, \dots, k-1$.

We would like to express $\sum_{i=1}^{k-1} T_i$ as a function of u_0 . To simplify notation, set $u = u_0$.

LEMMA 2. For $i = 1, \dots, k$, we have $u_i = \frac{u+i-1}{u+i}$.

PROOF. The proof is by induction on i . For $i = 1$, we have $u_1 = u/(u+1)$. Now, for $i \geq 2$ suppose that $u_{i-1} = (u+i-2)/(u+i-1)$. We have $u_i = 1/(2-u_{i-1}) = (u+i-1)/(2u+2i-2-u-i+2) = (u+i-1)/(u+i)$. \square

It follows that for $i = 1, \dots, k-1$ we have

$$T_i = A_i \frac{u+i-1}{u+i} \frac{u+i}{u+i+1} - A_i \frac{u+i}{u+i+1} = -A_i \frac{1}{u+i+1} = \frac{K_{i+1}}{u+i+1}.$$

Observe that as u_{-1} varies from 0 to 1, u varies from x' to x'' . Therefore, to maximize the total payoff, we have to choose $u \in [x', x'']$ so as to maximize

$$\frac{K_2}{u+2} + \dots + \frac{K_k}{u+k} = R(u).$$

By construction, the only maximum of $R(u)$ on $[x', x'']$ is α . It follows that in the payoff-maximizing Nash equilibrium of our game U_0 plays α .

Finally, note that the payoffs in our game are rational rather than integer. However, it is easy to see that we can multiply all payoffs to a player by their greatest common denominator without affecting his strategy. In the resulting game, all payoffs are integer. This concludes the proof of Theorem 2.

4. APPROXIMATING THE SOCIALLY OPTIMAL NASH EQUILIBRIUM

We have seen that the Nash equilibrium that maximizes the social welfare may involve strategies that are not in \mathbb{Q} . Hence, in this section we focus on finding a Nash equilibrium that is *almost* optimal from the social welfare perspective. We propose an algorithm that for any $\epsilon > 0$ finds a Nash equilibrium whose total payoff is within ϵ from optimal. The running time of this algorithm is polynomial in $1/\epsilon$, n and $|P_{\max}|$ (recall that P_{\max} is the maximum absolute value of an entry of a payoff matrix).

While the negative result of the previous section is for graphical games on paths, our algorithm applies to a wider range of scenarios. Namely, it runs in polynomial time on bounded-degree trees

as long as the best response policy of each vertex, given its parent, can be represented as a union of a polynomial number of rectangles. Note that path graphs always satisfy this condition: in [9] we showed how to compute such a representation, given a graph with maximum degree 2. Consequently, for path graphs the running time of our algorithm is guaranteed to be polynomial. (Note that [9] exhibits a family of graphical games on bounded-degree trees for which the best response policies of some of the vertices, given their parents, have exponential size, when represented as unions of rectangles.)

Due to space restrictions, in this version of the paper we present the algorithm for the case where the graph underlying the graphical game is a path. We then state our result for the general case; the proof can be found in the full version of this paper [10].

Suppose that \mathbf{s} is a strategy profile for a graphical game G . That is, \mathbf{s} assigns a mixed strategy to each vertex of G . Let $EP_V(\mathbf{s})$ be the expected payoff of player V under \mathbf{s} and let $EP(\mathbf{s}) = \sum_V EP_V(\mathbf{s})$. Let

$$M(G) = \max\{EP(\mathbf{s}) \mid \mathbf{s} \text{ is a Nash equilibrium for } G\}.$$

THEOREM 3. *Suppose that G is a graphical game on an n -vertex path. Then for any $\epsilon > 0$ there is an algorithm that constructs a Nash equilibrium \mathbf{s}' for G that satisfies $EP(\mathbf{s}') \geq M(G) - \epsilon$. The running time of the algorithm is $O(n^4 P_{\max}^3 / \epsilon^3)$.*

PROOF. Let $\{V_1, \dots, V_n\}$ be the set of all players. We start by constructing the best response policies for all $V_i, i = 1, \dots, n-1$. As shown in [9], this can be done in time $O(n^3)$.

Let $N > 5n$ be a parameter to be selected later, set $\delta = 1/N$, and define $X = \{j\delta \mid j = 0, \dots, N\}$. We say that v_j is an event point for a player V_i if it is a V_i -event point for $\mathcal{B}(V_i, V_{i-1})$ or $\mathcal{B}(V_{i+1}, V_i)$. For each player V_i , consider a finite set of strategies X_i given by

$$X_i = X \cup \{v_j \mid v_j \text{ is an event point for } V_i\}.$$

It has been shown in [9] that for any $i = 2, \dots, n$, the best response policy $\mathcal{B}(V_i, V_{i-1})$ has at most $2n + 4$ V_i -event points. As we require $N > 5n$, we have $|X_i| \leq 2N$; assume without loss of generality that $|X_i| = 2N$. Order the elements of X_i in increasing order as $x_1^i = 0 < x_2^i < \dots < x_{2N}^i$. We will refer to the strategies in X_i as *discrete* strategies of player V_i ; a strategy profile in which each player has a discrete strategy will be referred to as a *discrete* strategy profile.

We will now show that even we restrict each player V_i to strategies from X_i , the players can still achieve a Nash equilibrium, and moreover, the best such Nash equilibrium (with respect to the social welfare) has total payoff at least $M(G) - \epsilon$ as long as N is large enough.

Let \mathbf{s} be a strategy profile that maximizes social welfare. That is, let $\mathbf{s} = (s_1, \dots, s_n)$ where s_i is the mixed strategy of player V_i and $EP(\mathbf{s}) = M(G)$. For $i = 1, \dots, n$, let $t_i = \max\{x_j^i \mid x_j^i \leq s_i\}$. First, we will show that the strategy profile $\mathbf{t} = (t_1, \dots, t_n)$ is a Nash equilibrium for G .

Fix any $i, 1 < i \leq n$, and let $R = [v_1, v_2] \times [u_1, u_2]$ be the rectangle in $\mathcal{B}(V_i, V_{i-1})$ that contains (s_i, s_{i-1}) . As v_1 is a V_i -event point of $\mathcal{B}(V_i, V_{i-1})$, we have $v_1 \leq t_i$, so the point (t_i, s_{i-1}) is inside R . Similarly, the point u_1 is a V_{i-1} -event point of $\mathcal{B}(V_i, V_{i-1})$, so we have $u_1 \leq t_{i-1}$, and therefore the point (t_i, t_{i-1}) is inside R . This means that for any $i, 1 < i \leq n$, we have $t_{i-1} \in \text{pbr}_{V_{i-1}}(t_i)$, which implies that $\mathbf{t} = (t_1, \dots, t_n)$ is a Nash equilibrium for G .

Now, let us estimate the expected loss in social welfare caused by playing \mathbf{t} instead of \mathbf{s} .

LEMMA 3. *For any pair of strategy profiles \mathbf{t}, \mathbf{s} such that $|t_i - s_i| \leq \delta$ we have $|EP_{V_i}(\mathbf{s}) - EP_{V_i}(\mathbf{t})| \leq 24P_{\max}\delta$ for any $i = 1, \dots, n$.*

PROOF. Let P_{klm}^i be the payoff of the player V_i , when he plays k , V_{i-1} plays l , and V_{i+1} plays m . Fix $i = 1, \dots, n$ and for $k, l, m \in \{0, 1\}$, set

$$\begin{aligned} t^{klm} &= t_{i-1}^k (1 - t_{i-1})^{1-k} t_i^l (1 - t_i)^{1-l} t_{i+1}^m (1 - t_{i+1})^{1-m} \\ s^{klm} &= s_{i-1}^k (1 - s_{i-1})^{1-k} s_i^l (1 - s_i)^{1-l} s_{i+1}^m (1 - s_{i+1})^{1-m}. \end{aligned}$$

We have

$$\begin{aligned} |EP_{V_i}(\mathbf{s}) - EP_{V_i}(\mathbf{t})| &\leq \sum_{k,l,m=0,1} |P_{klm}^i (t^{klm} - s^{klm})| \leq \\ &8P_{\max} \max_{klm} |t^{klm} - s^{klm}| \end{aligned}$$

We will now show that for any $k, l, m \in \{0, 1\}$ we have $|t^{klm} - s^{klm}| \leq 3\delta$; clearly, this implies the lemma.

Indeed, fix $k, l, m \in \{0, 1\}$. Set

$$\begin{aligned} x &= t_{i-1}^k (1 - t_{i-1})^{1-k}, x' = s_{i-1}^k (1 - s_{i-1})^{1-k}, \\ y &= t_i^l (1 - t_i)^{1-l}, y' = s_i^l (1 - s_i)^{1-l}, \\ z &= t_{i+1}^m (1 - t_{i+1})^{1-m}, z' = s_{i+1}^m (1 - s_{i+1})^{1-m}. \end{aligned}$$

Observe that if $k = 0$ then $x - x' = (1 - t_{i-1}) - (1 - s_{i-1})$, and if $k = 1$ then $x - x' = t_{i-1} - s_{i-1}$, so $|x - x'| \leq \delta$. A similar argument shows $|y - y'| \leq \delta, |z - z'| \leq \delta$. Also, we have $x, x', y, y', z, z' \in [0, 1]$. Hence, $|t^{klm} - s^{klm}| = |xyz - x'y'z'| = |xyz - x'y'z + x'y'z - x'y'z' + x'y'z' - x'y'z'| \leq |x - x'|yz + |y - y'|x'z + |z - z'|x'y' \leq 3\delta$. \square

Lemma 3 implies $\sum_{i=1}^n |EP_{V_i}(\mathbf{s}) - EP_{V_i}(\mathbf{t})| \leq 24nP_{\max}\delta$, so by choosing $\delta < \epsilon / (24nP_{\max})$, or, equivalently, setting $N > 24nP_{\max} / \epsilon$, we can ensure that the total expected payoff for the strategy profile \mathbf{t} is within ϵ from optimal.

We will now show that we can find the best discrete Nash equilibrium (with respect to the social welfare) using dynamic programming. As \mathbf{t} is a discrete strategy profile, this means that the strategy profile found by our algorithm will be at least as good as \mathbf{t} .

Define $m_i^{l,k}$ to be the maximum total payoff that V_1, \dots, V_{i-1} can achieve if each $V_j, j \leq i$, chooses a strategy from X_j , for each $j < i$ the strategy of V_j is a potential best response to the strategy of V_{j+1} , and, moreover, V_{i-1} plays x_{i-1}^l , V_i plays x_i^k . If there is no way to choose the strategies for V_1, \dots, V_{i-1} to satisfy these conditions, we set $m_i^{l,k} = -\infty$. The values $m_i^{l,k}, i = 1, \dots, n; k, l = 1, \dots, N$, can be computed inductively, as follows.

We have $m_1^{l,k} = 0$ for $k, l = 1, \dots, N$. Now, suppose that we have already computed $m_j^{l,k}$ for all $j < i; k, l = 1, \dots, N$. To compute $m_i^{k,l}$, we first check if $(x_i^k, x_{i-1}^l) \in \mathcal{B}(V_i, V_{i-1})$. If this is not the case, we have $m_i^{l,k} = -\infty$. Otherwise, consider the set $Y = X_{i-2} \cap \text{pbr}_{V_{i-2}}(x_{i-1}^l)$, i.e., the set of all discrete strategies of V_{i-2} that are potential best responses to x_{i-1}^l . The proof of Theorem 1 in [9] implies that the set $\text{pbr}_{V_{i-2}}(x_{i-1}^l)$ is non-empty: the player V_{i-2} has a potential best response to any strategy of V_{i-1} , in particular, x_{i-1}^l . By construction of the set X_{i-2} , this implies that Y is not empty. For each $x_{i-2}^j \in Y$, let p_{jlk} be the payoff that V_{i-1} receives when V_{i-2} plays x_{i-2}^j , V_{i-1} plays x_{i-1}^l , and V_i plays x_i^k . Clearly, p_{jlk} can be computed in constant time. Then we have $m_i^{l,k} = \max\{m_{i-1}^{j,l} + p_{jlk} \mid x_{i-2}^j \in Y\}$.

Finally, suppose that we have computed $m_n^{l,k}$ for $l, k = 1, \dots, N$. We still need to take into account the payoff of player V_n . Hence,

we consider all pairs (x_n^k, x_{n-1}^l) that satisfy $x_{n-1}^l \in \text{pbr}_{V_{n-1}}(x_n^k)$, and pick the one that maximizes the sum of $m_n^{k,l}$ and the payoff of V_n when he plays x_n^k and V_{n-1} plays x_{n-1}^l . This results in the maximum total payoff the players can achieve in a Nash equilibrium using discrete strategies; the actual strategy profile that produces this payoff can be reconstructed using standard dynamic programming techniques.

It is easy to see that each $m_n^{l,k}$ can be computed in time $O(N)$, i.e., all of them can be computed in time $O(nN^3)$. Recall that we have to select $N \geq (24nP_{\max})/\epsilon$ to ensure that the strategy profile we output has total payoff that is within ϵ from optimal. We conclude that we can compute an ϵ -approximation to the best Nash equilibrium in time $O(n^4 P_{\max}^3/\epsilon^3)$. This completes the proof of Theorem 3.

To state our result for the general case (i.e., when the underlying graph is a bounded-degree tree rather than a path), we need additional notation. If G has n players, let $q(n)$ be an upper bound on the number of event points in the representation of any best response policy. That is, we assume that for any vertex U with parent V , $B(V, U)$ has at most $q(n)$ event points. We will be interested in the situation in which $q(n)$ is polynomial in n .

THEOREM 4. *Let G be an n -player graphical game on a tree in which each node has at most Δ children. Suppose we are given a set of best-response policies for G in which each best-response policy $B(V, U)$ is represented by a set of rectangles with at most $q(n)$ event points. For any $\epsilon > 0$, there is an algorithm that constructs a Nash equilibrium \mathbf{s}' for G that satisfies $EP(\mathbf{s}') \geq M(G) - \epsilon$. The running time of the algorithm is polynomial in n , P_{\max} and ϵ^{-1} provided that the tree has bounded degree (that is, $\Delta = O(1)$) and $q(n)$ is a polynomial in n . In particular, if*

$$N = \max((\Delta + 1)q(n) + 1, n2^{\Delta+2}(\Delta + 2)P_{\max}\epsilon^{-1})$$

and $\Delta > 1$ then the running time is $O(n\Delta(2N)^\Delta)$.

For the proof of this theorem, see [10].

4.1 A polynomial-time algorithm for multiplicative approximation

The running time of our algorithm is pseudopolynomial rather than polynomial, because it includes a factor which is polynomial in P_{\max} , the maximum (in absolute value) entry in any payoff matrix. If we are interested in multiplicative approximation rather than additive one, this can be improved to polynomial.

First, note that we cannot expect a multiplicative approximation for all inputs. That is, we cannot hope to have an algorithm that computes a Nash equilibrium with total payoff at least $(1 - \epsilon)M(G)$. If we had such an algorithm, then for graphical games G with $M(G) = 0$, the algorithm would be required to output the optimal solution. To show that this is infeasible, observe that we can use the techniques of Section 3.2 to construct two integer-coefficient graphical games on paths of length $O(n)$ such that for some $X \in \mathbb{R}$ the maximal total payoff in the first game is X , the maximal total payoff in the second game is $-X$, and for both games, the strategy profiles that achieve the maximal total payoffs involve algebraic numbers of degree n . By combining the two games so that the first vertex of the second game becomes connected to the last vertex of the first game, but the payoffs of all players do not change, we obtain a graphical game in which the best Nash equilibrium has total payoff 0, yet the strategies that lead to this payoff have high algebraic complexity.

However, we can achieve a multiplicative approximation when all entries of the payoff matrices are positive and the ratio between

any two entries is polynomially bounded. Recall that we assume that all payoffs are integer, and let $P_{\min} > 0$ be the smallest entry of any payoff matrix. In this case, for any strategy profile the payoff to player i is at least P_{\min} , so the total payoff in the social-welfare maximizing Nash equilibrium \mathbf{s} satisfies $M(G) \geq nP_{\min}$. Moreover, Lemma 3 implies that by choosing $\delta < \epsilon/(24P_{\max}/P_{\min})$, we can ensure that the Nash equilibrium \mathbf{t} produced by our algorithm satisfies

$$\sum_{i=1}^n EP_{V_i}(\mathbf{s}) - \sum_{i=1}^n EP_{V_i}(\mathbf{t}) \leq 24P_{\max}\delta n \leq \epsilon n P_{\min} \leq \epsilon M(G),$$

i.e., for this value of δ we have $\sum_{i=1}^n EP_{V_i}(\mathbf{t}) \geq (1 - \epsilon)M(G)$. Recall that the running time of our algorithm is $O(nN^3)$, where N has to be selected to satisfy $N > 5n$, $N = 1/\delta$. It follows that if $P_{\min} > 0$, $P_{\max}/P_{\min} = \text{poly}(n)$, we can choose N so that our algorithm provides a multiplicative approximation guarantee and runs in time polynomial in n and $1/\epsilon$.

5. BOUNDED PAYOFF NASH EQUILIBRIA

Another natural way to define what is a “good” Nash equilibrium is to require that each player’s expected payoff exceeds a certain threshold. These thresholds do not have to be the same for all players. In this case, in addition to the payoff matrices of the n players, we are given n numbers T_1, \dots, T_n , and our goal is to find a Nash equilibrium in which the payoff of player i is at least T_i , or report that no such Nash equilibrium exists. It turns out that we can design an FPTAS for this problem using the same techniques as in the previous section.

THEOREM 5. *Given a graphical game G on an n -vertex path and n rational numbers T_1, \dots, T_n , suppose that there exists a strategy profile \mathbf{s} such that \mathbf{s} is a Nash equilibrium for G and $EP_{V_i}(\mathbf{s}) \geq T_i$ for $i = 1, \dots, n$. Then for any $\epsilon > 0$ we can find in time $O(\max\{nP_{\max}^3/\epsilon^3, n^4/\epsilon^3\})$ a strategy profile \mathbf{s}' such that \mathbf{s}' is a Nash equilibrium for G and $EP_{V_i}(\mathbf{s}') \geq T_i - \epsilon$ for $i = 1, \dots, n$.*

PROOF. The proof is similar to that of Theorem 3. First, we construct the best response policies for all players, choose $N > 5n$, and construct the sets X_i , $i = 1, \dots, n$, as described in the proof of Theorem 3.

Consider a strategy profile \mathbf{s} such that \mathbf{s} is a Nash equilibrium for G and $EP_{V_i}(\mathbf{s}) \geq T_i$ for $i = 1, \dots, n$. We construct a strategy profile $\mathbf{t}_i = \max\{x_i^j \mid x_i^j \leq s_i\}$ and use the same argument as in the proof of Theorem 3 to show that \mathbf{t} is a Nash equilibrium for G . By Lemma 3, we have $|EP_{V_i}(\mathbf{s}) - EP_{V_i}(\mathbf{t})| \leq 24P_{\max}\delta$, so choosing $\delta < \epsilon/(24P_{\max})$, or, equivalently, $N > \max\{5n, 24P_{\max}/\epsilon\}$, we can ensure $EP_{V_i}(\mathbf{t}) \geq T_i - \epsilon$ for $i = 1, \dots, n$.

Now, we will use dynamic programming to find a discrete Nash equilibrium that satisfies $EP_{V_i}(\mathbf{t}) \geq T_i - \epsilon$ for $i = 1, \dots, n$. As \mathbf{t} is a discrete strategy profile, our algorithm will succeed whenever there is a strategy profile \mathbf{s} with $EP_{V_i}(\mathbf{s}) \geq T_i - \epsilon$ for $i = 1, \dots, n$.

Let $z_i^{l,k} = 1$ if there is a discrete strategy profile such that for any $j < i$ the strategy of the player V_j is a potential best response to the strategy of V_{j+1} , the expected payoff of V_j is at least $T_j - \epsilon$, and, moreover, V_{i-1} plays x_{i-1}^l , V_i plays x_i^k . Otherwise, let $z_i^{l,k} = 0$. We can compute $z_i^{l,k}$, $i = 1, \dots, n$; $k, l = 1, \dots, N$ inductively, as follows.

We have $z_1^{l,k} = 1$ for $k, l = 1, \dots, N$. Now, suppose that we have already computed $z_j^{l,k}$ for all $j < i$; $k, l = 1, \dots, N$. To compute $z_i^{k,l}$, we first check if $(x_i^k, x_{i-1}^l) \in \mathcal{B}(V_i, V_{i-1})$. If this

is not the case, clearly, $z_i^{k,l} = 0$. Otherwise, consider the set $Y = X_{i-2} \cap \text{pbr}_{V_{i-2}}(x_{i-1}^l)$, i.e., the set of all discrete strategies of V_{i-2} that are potential best responses to x_{i-1}^l . It has been shown in the proof of Theorem 3 that $Y \neq \emptyset$. For each $x_{i-2}^j \in Y$, let p_{jlk} be the payoff that V_{i-1} receives when V_{i-2} plays x_{i-2}^j , V_{i-1} plays x_{i-1}^l , and V_i plays x_i^k . Clearly, p_{jlk} can be computed in constant time. If there exists an $x_{i-2}^j \in Y$ such that $z_{i-1}^{j,l} = 1$ and $p_{jlk} \geq T_{i-2} - \epsilon$, set $z_i^{l,k} = 1$. Otherwise, set $z_i^{l,k} = 0$.

Having computed $z_n^{l,k}$, $l, k = 1, \dots, N$, we check if $z_n^{l,k} = 1$ for some pair (l, k) . If such a pair of indices exists, we instruct V_n to play x_n^k and use dynamic programming techniques (or, equivalently, the downstream pass of the algorithm of [13]) to find a Nash equilibrium s' that satisfies $EP_{V_i}(s') \geq T_i - \epsilon$ for $i = 1, \dots, n$ (recall that V_n is a dummy player, i.e., we assume $T_n = 0$, $EP_n(s') = 0$ for any choice of s'). If $z_n^{l,k} = 0$ for all $l, k = 1, \dots, N$, there is no discrete Nash equilibrium s' that satisfies $EP_{V_i}(s') \geq T_i - \epsilon$ for $i = 1, \dots, n$ and hence no Nash equilibrium s (not necessarily discrete) such that $EP_{V_i}(s) \geq T_i$ for $i = 1, \dots, n$.

The running time analysis is similar to that for Theorem 3; we conclude that the running time of our algorithm is $O(nN^3) = O(\max\{nP_{\max}^3/\epsilon^3, n^4/\epsilon^3\})$. \square

REMARK 1. *Theorem 5 can be extended to trees of bounded degree in the same way as Theorem 4.*

5.1 Exact Computation

Another approach to finding Nash equilibria with bounded payoffs is based on inductively computing the subsets of the best response policies of all players so as to exclude the points that do not provide sufficient payoffs to some of the players. Formally, we say that a strategy v of the player V is a *potential best response to a strategy w of its parent W with respect to a threshold vector $\mathbf{T} = (T_1, \dots, T_n)$* , (denoted by $v \in \text{pbr}_V(w, \mathbf{T})$) if there is an equilibrium in the instance $G_{(V,W), w=w}$ in which V plays mixed strategy v and the payoff to any player V_i downstream of V (including V) is at least T_i . The *best response policy for V with respect to a threshold vector \mathbf{T}* is defined as $\mathcal{B}(W, V, \mathbf{T}) = \{(w, v) \mid v \in \text{pbr}_V(w, \mathbf{T}), w \in [0, 1]\}$.

It is easy to see that if any of the sets $\mathcal{B}(V_j, V_{j-1}, \mathbf{T})$, $j = 1, \dots, n$, is empty, it means that it is impossible to provide all players with expected payoffs prescribed by \mathbf{T} . Otherwise, one can apply the downstream pass of the original algorithm of [13] to find a Nash equilibrium. As we assume that V_n is a dummy vertex whose payoff is identically 0, the Nash equilibrium with these payoffs exists as long as $T_n \leq 0$ and $\mathcal{B}(V_n, V_{n-1}, \mathbf{T})$ is not empty.

Using the techniques developed in [9], it is not hard to show that for any $j = 1, \dots, n$, the set $\mathcal{B}(V_j, V_{j-1}, \mathbf{T})$ consists of a finite number of rectangles, and one can compute $\mathcal{B}(V_{j+1}, V_j, \mathbf{T})$ given $\mathcal{B}(V_j, V_{j-1}, \mathbf{T})$. The advantage of this approach is that it allows us to represent *all* Nash equilibria that provide required payoffs to the players. However, it is not likely to be practical, since it turns out that the rectangles that appear in the representation of $\mathcal{B}(V_j, V_{j-1}, \mathbf{T})$ may have irrational coordinates.

CLAIM 6. *There exists a graphical game G on a 3-vertex path UVW and a vector $\mathbf{T} = (T_1, T_2, T_3)$ such that $\mathcal{B}(V, W, \mathbf{T})$ cannot be represented as a union of a finite number of rectangles with rational coordinates.*

PROOF. We define the payoffs to the players in G as follows. The payoff to U is identically 0, i.e., $P^0(U) = P^1(U) = 0$. Using Claim 1, we select the payoffs to V so that $P^0(V) = uw$,

$P^1(V) = P^0(V) + w - .8u - .1$, where u and w are the (mixed) strategies of U and W , respectively. It follows that V is indifferent between playing 0 and 1 if and only if $w = f(u) = .8u + .1$; observe that for any $u \in [0, 1]$ we have $f(u) \in [0, 1]$. It is not hard to see that we have

$$\mathcal{B}(W, V) = [0, .1] \times \{0\} \cup [.1, .9] \times [0, 1] \cup [.9, 1] \times \{1\}.$$

The payoffs to W are not important for our construction; for example, set $P_0(W) = P_1(W) = 0$.

Now, set $\mathbf{T} = (0, 1/8, 0)$, i.e., we are interested in Nash equilibria in which V 's expected payoff is at least $1/8$. Suppose $w \in [0, 1]$. The player V can play a mixed strategy v when W is playing w as long as U plays $u = f^{-1}(w) = 5w/4 - 1/8$ (to ensure that V is indifferent between 0 and 1) and $P^0(V) = P^1(V) = uw = w(5w/4 - 1/8) \geq 1/8$. The latter condition is satisfied if $w \leq (1 - \sqrt{41})/20 < 0$ or $w \geq (1 + \sqrt{41})/20$. Note that we have $.1 < (1 + \sqrt{41})/20 < .9$. For any other value of w , any strategy of U either makes V prefer one of the pure strategies or does not provide it with a sufficient expected payoff. There are also some values of w for which V can play a pure strategy (0 or 1) as a potential best response to W and guarantee itself an expected payoff of at least $1/8$; it can be shown that these values of w form a finite number of segments in $[0, 1]$. We conclude that any representation of $\mathcal{B}(W, V, \mathbf{T})$ as a union of a finite number of rectangles must contain a rectangle of the form $[(1 + \sqrt{41})/20, w''] \times [v', v'']$ for some $w'', v', v'' \in [0, 1]$. \square

On the other hand, it can be shown that for any integer payoff matrices and threshold vectors and any $j = 1, \dots, n-1$, the sets $\mathcal{B}(V_{j+1}, V_j, \mathbf{T})$ contain no rectangles of the form $[u', u''] \times \{v\}$ or $\{v\} \times [w', w'']$, where $v \in \mathbb{R} \setminus \mathbb{Q}$. This means that if $\mathcal{B}(V_n, V_{n-1}, \mathbf{T})$ is non-empty, i.e., there is a Nash equilibrium with payoffs prescribed by \mathbf{T} , then the downstream pass of the algorithm of [13] can always pick a strategy profile that forms a Nash equilibrium, provides a payoff of at least T_i to the player V_i , and has no irrational coordinates. Hence, unlike in the case of the Nash equilibrium that maximizes the social welfare, working with irrational numbers is not necessary, and the fact that the algorithm discussed in this section has to do so can be seen as an argument against using this approach.

6. OTHER CRITERIA FOR SELECTING A NASH EQUILIBRIUM

In this section, we consider several other criteria that can be useful in selecting a Nash equilibrium.

6.1 Combining welfare maximization with bounds on payoffs

In many real life scenarios, we want to maximize the social welfare subject to certain restrictions on the payoffs to individual players. For example, we may want to ensure that no player gets a negative expected payoff, or that the expected payoff to player i is at least $P_{\max}^i - \xi$, where P_{\max}^i is the maximum entry of i 's payoff matrix and ξ is a fixed parameter. Formally, given a graphical game G and a vector T_1, \dots, T_n , let S be the set of all Nash equilibria s of G that satisfy $T_i \leq EP_{V_i}(s)$ for $i = 1, \dots, n$, and let $\hat{s} = \text{argmax}_{s \in S} EP(s)$.

If the set S is non-empty, we can find a Nash equilibrium s' that is ϵ -close to satisfying the payoff bounds and is within ϵ from \hat{s} with respect to the total payoff by combining the algorithms of Section 4 and Section 5.

Namely, for a given $\epsilon > 0$, choose δ as in the proof of Theorem 3, and let X_i be the set of all discrete strategies of player V_i (for a

formal definition, see the proof of Theorem 3). Combining the proofs of Theorem 3 and Theorem 5, we can see that the strategy profile $\hat{\mathbf{t}}$ given by $\hat{t}_i = \max\{x_i^j \mid x_i^j \leq \hat{s}_i\}$ satisfies $EP_{V_i}(\hat{\mathbf{t}}) \geq T_i - \epsilon$, $|EP(\hat{\mathbf{s}}) - EP(\hat{\mathbf{t}})| \leq \epsilon$.

Define $\hat{m}_i^{l,k}$ to be the maximum total payoff that V_1, \dots, V_{i-1} can achieve if each V_j , $j \leq i$, chooses a strategy from X_j , for each $j < i$ the strategy of V_j is a potential best response to the strategy of V_{j+1} and the payoff to player V_j is at least $T_j - \epsilon$, and, moreover, V_{i-1} plays x_{i-1}^l , V_i plays x_i^k . If there is no way to choose the strategies for V_1, \dots, V_{i-1} to satisfy these conditions, we set $m_i^{l,k} = -\infty$. The $\hat{m}_i^{l,k}$ can be computed by dynamic programming similarly to the $m_i^{l,k}$ and $z_i^{l,k}$ in the proofs of Theorems 3 and 5. Finally, as in the proof of Theorem 3, we use $m_n^{l,k}$ to select the best discrete Nash equilibrium subject to the payoff constraints.

Even more generally, we may want to maximize the total payoff to a subset of players (who are assumed to be able to redistribute the profits fairly among themselves) while guaranteeing certain expected payoffs to (a subset of) the other players. This problem can be handled similarly.

6.2 A minimax approach

A more egalitarian measure of the quality of a Nash equilibrium is the minimal expected payoff to a player. The optimal solution with respect to this measure is a Nash equilibrium in which the minimal expected payoff to a player is maximal. To find an approximation to such a Nash equilibrium, we can combine the algorithm of Section 5 with binary search on the space of potential lower bounds. Note that the expected payoff to any player V_i given a strategy \mathbf{s} always satisfies $-P_{\max} \leq EP_{V_i}(\mathbf{s}) \leq P_{\max}$.

For a fixed $\epsilon > 0$, we start by setting $T' = -P_{\max}$, $T'' = P_{\max}$, $T^* = (T' + T'')/2$. We then run the algorithm of Section 5 with $T_1 = \dots = T_n = T^*$. If the algorithm succeeds in finding a Nash equilibrium \mathbf{s}' that satisfies $EP_{V_i}(\mathbf{s}') \geq T^* - \epsilon$ for all $i = 1, \dots, n$, we set $T' = T^*$, $T^* = (T' + T'')/2$; otherwise, we set $T'' = T^*$, $T^* = (T' + T'')/2$ and loop. We repeat this process until $|T' - T''| \leq \epsilon$. It is not hard to check that for any $p \in \mathbb{R}$, if there is a Nash equilibrium \mathbf{s} such that $\min_{i=1, \dots, n} EP_{V_i}(\mathbf{s}) \geq p$, then our algorithm outputs a Nash equilibrium \mathbf{s}' that satisfies $\min_{i=1, \dots, n} EP_{V_i}(\mathbf{s}') \geq p - 2\epsilon$. The running time of our algorithm is $O(\max\{nP_{\max}^3 \log \epsilon^{-1}/\epsilon^3, n^4 \log \epsilon^{-1}/\epsilon^3\})$.

6.3 Equalizing the payoffs

When the players' payoff matrices are not very different, it is reasonable to demand that the expected payoffs to the players do not differ by much either. We will now show that Nash equilibria in this category can be approximated in polynomial time as well.

Indeed, observe that the algorithm of Section 5 can be easily modified to deal with upper bounds on individual payoffs rather than lower bounds. Moreover, we can efficiently compute an approximation to a Nash equilibrium that satisfies both the upper bound and the lower bound for each player. More precisely, suppose that we are given a graphical game G , $2n$ rational numbers $T_1, \dots, T_n, T'_1, \dots, T'_n$ and $\epsilon > 0$. Then if there exists a strategy profile \mathbf{s} such that \mathbf{s} is a Nash equilibrium for G and $T_i \leq EP_{V_i}(\mathbf{s}) \leq T'_i$ for $i = 1, \dots, n$, we can find a strategy profile \mathbf{s}' such that \mathbf{s}' is a Nash equilibrium for G and $T_i - \epsilon \leq EP_{V_i}(\mathbf{s}') \leq T'_i + \epsilon$ for $i = 1, \dots, n$. The modified algorithm also runs in time $O(\max\{nP_{\max}^3/\epsilon^3, [4n^4/\epsilon^3]\})$.

This observation allows us to approximate Nash equilibria in which all players' expected payoffs differ by at most ξ for any fixed $\xi > 0$. Given an $\epsilon > 0$, we set $T_1 = \dots = T_n = -P_{\max}$, $T'_1 = \dots = T'_n = -P_{\max} + \xi + \epsilon$, and run the modified version of the algorithm of Section 5. If it fails to find a solution, we in-

crement all T_i, T'_i by ϵ and loop. We continue until the algorithm finds a solution, or $T_i \geq P_{\max}$.

Suppose that there exists a Nash equilibrium \mathbf{s} that satisfies $|EP_{V_i}(\mathbf{s}) - EP_{V_j}(\mathbf{s})| \leq \xi$ for all $i, j = 1, \dots, n$. Set $r = \min_{i=1, \dots, n} EP_{V_i}(\mathbf{s})$; we have $r \leq EP_{V_i}(\mathbf{s}) \leq r + \xi$ for all $i = 1, \dots, n$. There exists a $k \geq 0$ such that $-P_{\max} + (k-1)\epsilon \leq r \leq -P_{\max} + k\epsilon$. During the k th step of the algorithm, we set $T_1 = \dots = T_n = -P_{\max} + (k-1)\epsilon$, i.e., we have $r - \epsilon \leq T_i \leq r$, $r + \xi \leq T'_i \leq r + \xi + \epsilon$. That is, the Nash equilibrium \mathbf{s} satisfies $T_i \leq r \leq EP_{V_i}(\mathbf{s}) \leq r + \xi \leq T'_i$, which means that when T_i is set to $-P_{\max} + (k-1)\epsilon$, our algorithm is guaranteed to output a Nash equilibrium \mathbf{t} that satisfies $r - 2\epsilon \leq T_i - \epsilon \leq EP_{V_i}(\mathbf{t}) \leq T'_i + \epsilon \leq r + \xi + 2\epsilon$. We conclude that whenever such a Nash equilibrium \mathbf{s} exists, our algorithm outputs a Nash equilibrium \mathbf{t} that satisfies $|EP_{V_i}(\mathbf{t}) - EP_{V_j}(\mathbf{t})| \leq \xi + 4\epsilon$ for all $i, j = 1, \dots, n$. The running time of this algorithm is $O(\max\{nP_{\max}^3/\epsilon^4, n^4/\epsilon^4\})$.

Note also that we can find the smallest ξ for which such a Nash equilibrium exists by combining this algorithm with binary search over the space $\xi = [0, 2P_{\max}]$. This identifies an approximation to the "fairest" Nash equilibrium, i.e., one in which the players' expected payoffs differ by the smallest possible amount.

Finally, note that all results in this section can be extended to bounded-degree trees.

7. CONCLUSIONS

We have studied the problem of equilibrium selection in graphical games on bounded-degree trees. We considered several criteria for selecting a Nash equilibrium, such as maximizing the social welfare, ensuring a lower bound on the expected payoff of each player, etc. First, we focused on the algebraic complexity of a social welfare-maximizing Nash equilibrium, and proved strong negative results for that problem. Namely, we showed that even for graphical games on paths, any algebraic number $\alpha \in [0, 1]$ may be the only strategy available to some player in all social welfare-maximizing Nash equilibria. This is in sharp contrast with the fact that graphical games on trees always possess a Nash equilibrium in which all players' strategies are rational numbers.

We then provided approximation algorithms for selecting Nash equilibria with special properties. While the problem of finding approximate Nash equilibria for various classes of games has received a lot of attention in recent years, most of the existing work aims to find ϵ -Nash equilibria that satisfy (or are ϵ -close to satisfying) certain properties. Our approach is different in that we insist on outputting an exact Nash equilibrium, which is ϵ -close to satisfying a given requirement. As argued in the introduction, there are several reasons to prefer a solution that constitutes an exact Nash equilibrium.

Our algorithms are fully polynomial time approximation schemes, i.e., their running time is polynomial in the inverse of the approximation parameter ϵ , though they may be pseudopolynomial with respect to the input size. Under mild restrictions on the inputs, they can be modified to be truly polynomial. This is the strongest positive result one can derive for a problem whose exact solutions may be hard to represent, as is the case for many of the problems considered here. While we prove our results for games on a path, they can be generalized to any tree for which the best response policies have compact representations as unions of rectangles. In the full version of the paper we describe our algorithms for the general case.

Further work in this vein could include extensions to the kinds of guarantees sought for Nash equilibria, such as guaranteeing total payoffs for subsets of players, selecting equilibria in which some players are receiving significantly higher payoffs than their peers, etc. At the moment however, it is perhaps more important to inves-

tigate whether Nash equilibria of graphical games can be computed in a decentralized manner, in contrast to the algorithms we have introduced here.

It is natural to ask if our results or those of [9] can be generalized to games with three or more actions. However, it seems that this will make the analysis significantly more difficult. In particular, note that one can view the bounded payoff games as a very limited special case of games with three actions per player. Namely, given a two-action game with payoff bounds, consider a game in which each player V_i has a third action that guarantees him a payoff of T_i no matter what everyone else does. Then checking if there is a Nash equilibrium in which none of the players assigns a non-zero probability to his third action is equivalent to checking if there exists a Nash equilibrium that satisfies the payoff bounds in the original game, and Section 5.1 shows that finding an exact solution to this problem requires new ideas.

Alternatively it may be interesting to look for similar results in the context of correlated equilibria (CE), especially since the best CE may have higher value (total expected payoff) than the best NE. The ratio between these values is called the *mediation value* in [1]. It is known from [1] that the mediation value of 2-player, 2-action games with non-negative payoffs is at most $\frac{4}{3}$, and they exhibit a 3-player game for which it is infinite. Furthermore, a 2-player, 3-action example from [1] also has infinite mediation value.

8. REFERENCES

- [1] I. Ashlagi, D. Monderer and M. Tennenholtz, On the Value of Correlation, Proceedings of Dagstuhl seminar 05011 (2005)
- [2] R. Aumann, Subjectivity and Correlation in Randomized Strategies, Journal of Mathematical Economics **1** pp. 67-96 (1974)
- [3] B. Blum, C. R. Shelton, and D. Koller, A Continuation Method for Nash Equilibria in Structured Games, Proceedings of IJCAI'03
- [4] X. Chen, X. Deng and S. Teng, Computing Nash Equilibria: Approximation and Smoothed Complexity, Proceedings of FOCS'06
- [5] X. Chen, X. Deng, Settling the Complexity of 2-Player Nash-Equilibrium, Proceedings of FOCS'06
- [6] V. Conitzer and T. Sandholm, Complexity Results about Nash Equilibria, Proceedings of IJCAI'03
- [7] C. Daskalakis, P. W. Goldberg and C. H. Papadimitriou, The Complexity of Computing a Nash Equilibrium, Proceedings of STOC'06
- [8] R. S. Datta, Universality of Nash Equilibria, Mathematics of Operations Research **28:3**, 2003
- [9] E. Elkind, L. A. Goldberg, and P. W. Goldberg, Nash Equilibria in Graphical games on Trees Revisited, Proceedings of ACM EC'06
- [10] E. Elkind, L. A. Goldberg, and P. W. Goldberg, Computing Good Nash Equilibria in Graphical Games, <http://arxiv.org/abs/cs.GT/0703133>
- [11] I. Gilboa and E. Zemel, Nash and Correlated Equilibria: Some Complexity Considerations, Games and Economic Behavior, **1** pp. 80-93 (1989)
- [12] P. W. Goldberg and C. H. Papadimitriou, Reducibility Among Equilibrium Problems, Proceedings of STOC'06
- [13] M. Kearns, M. Littman, and S. Singh, Graphical Models for Game Theory, Proceedings of UAI'01
- [14] M. Littman, M. Kearns, and S. Singh, An Efficient Exact Algorithm for Singly Connected Graphical Games, Proceedings of NIPS'01
- [15] R. Lipton and E. Markakis, Nash Equilibria via Polynomial Equations, Proceedings of LATIN'04
- [16] L. Ortiz and M. Kearns, Nash Propagation for Loopy Graphical Games, Proceedings of NIPS'03
- [17] C.H. Papadimitriou, Computing Correlated Equilibria in Multi-Player Games, Proceedings of STOC'05
- [18] C.H. Papadimitriou and T. Roughgarden, Computing Equilibria in Multi-Player Games, Proceedings of SODA'05
- [19] D. Vickrey and D. Koller, Multi-agent Algorithms for Solving Graphical Games, Proceedings of AAAI'02