Journal of Artificial Intelligence Research 69 (2020) 67-84 Submitted 08/2016; published 09/2020

Research Note

On Sparse Discretization for Graphical Games

Luis E. Ortiz LEORTIZ@UMICH.EDU
Department of Computer and Information Science
University of Michigan - Dearborn
4901 Evergreen Road, Dearborn, MI 48128-1491 USA

Abstract

Graphical games are one of the earliest examples of the impact that the general field of graphical models have had in other areas, and in this particular case, in classical mathematical models in game theory. Graphical multi-hypermatrix games, a concept formally introduced in this research note, generalize graphical games while allowing the possibility of further space savings in model representation to that of standard graphical games. The main focus of this research note is discretization schemes for computing approximate Nash equilibria, with emphasis on graphical games, but also briefly touching on normal-form and polymatrix games. The main technical contribution is a theorem that establishes sufficient conditions for a discretization of the players' space of mixed strategies to contain an approximate Nash equilibrium. The result is actually stronger because every exact Nash equilibrium has a nearby approximate Nash equilibrium on the grid induced by the discretization. The sufficient conditions are weaker than those of previous results. In particular, a uniform discretization of size linear in the inverse of the approximation error and in the natural game-representation parameters suffices. The theorem holds for a generalization of graphical games, introduced here. The result has already been useful in the design and analysis of tractable algorithms for graphical games with parametric payoff functions and certain game-graph structures. For standard graphical games, under natural conditions, the discretization is logarithmic in the game-representation size, a substantial improvement over the linear dependency previously required. Combining the improved discretization result with old results on constraint networks in AI simplifies the derivation and analysis of algorithms for computing approximate Nash equilibria in graphical games.

1. Introduction

Graphical games (GGs) are graphical models for classical mathematical models in game theory originally introduced within the artificial intelligence community by Kearns, Littman, and Singh (2001). As is well-known to the readership of this journal, graphical models are modern computationally amenable compact network-based representations introduced and considerably developed and studied in the artificial intelligence (AI), statistics, and machine learning (ML) community over the last 40 years (see, e.g., Waltz, 1975, Montanari, 1974, Pearl, 1988, Dechter, 2003, and Koller & Friedman, 2009, for general introductory information and historical context). Graphical games have already impacted AI and several other areas of computer science, with theory being a particular highlight (see, e.g., Daskalakis, Goldberg, & Papadimitriou, 2009, and the references therein). Roughly speaking, as initially defined by Kearns et al. (2001), this class of *games* are *mathematical abstractions of strategic interactions among individuals, restricted or induced by a graph.*

In this particular case, a *graph*, a *binary action set*, and a *set of local payoff matrices*, or, equivalently, *utility functions*, define the game. Each of the n players has an individual local payoff hypermatrix, involving the action of the player and the player's neighbors in the graph, chosen from each player's set of 2 actions available. *Graphical multi-hypermatrix games (GMhGs)*, a concept formally introduced in this research note, generalize graphical games while allowing the possibility of further space savings in model representation to that of standard graphical games.

This research note revisits the simple *uniform-discretization scheme* that Kearns et al. (2001) proposed in the context of n-player 2-action GGs. They showed the following in that work. Let k be the size of the largest set of neighbors of any player and ϵ the approximation error parameter. If the *size of the individual grid* (i.e., number of grid points) that the uniform-discretization scheme induces over the probability of playing an action is $\Omega(2^k/\epsilon)$, then for each exact Nash equilibrium of the GG, its closest joint mixed-strategy in the resulting regular *joint* grid is an ϵ-Nash equilibrium of the GG. [1] Kearns et al. (2001) used that discretization to design a special type of dynamic-programming algorithm tailored to computing approximate Nash equilibria in GGs with tree-structured graphs, which they called **TreeNash**. That algorithm runs in time linear in the number of players and $O(2^{k^2})$, assuming a fixed ϵ. The size of the input representation of the n-player 2-action GG is $O(n\,2^k)$. Ortiz and Kearns (2003) later extended **TreeNash** as a heuristic for GGs with loopy graphs, leading to an algorithm called **NashProp**, which stands for "Nash Propagation."

An unpublished note drafted back in December 2002 (Ortiz, 2002) provided a significantly sharper bound of $O(k/\epsilon)$ on the sufficient size of the discretization to achieve the same approximation result. [2] The revised bound was *logarithmic* in the representation size of the game, as opposed to the previous *linear* bound that Kearns et al. (2001) derived. [3] The revised, significantly tighter upper-bound on the sufficient discretization size led to an improved running time of $O(k^k)$ for **TreeNash** in terms of just k. This meant that when using the sparser discretization derived in the old note (Ortiz, 2002), **TreeNash** becomes a quasi-polynomial time approximation scheme (quasi-PTAS) [4] to compute an ϵ-Nash equilibrium. Daskalakis and Papadimitriou (2008) independently rediscovered this result using a proof technique that differs from the simple algebraic approach used in the old note (Ortiz, 2002), and here. Similarly, some of the results regarding algorithmic implications presented here that followed from the old note (Ortiz, 2002), particularly for normal-form games, have also been independently rediscovered in the literature using different approaches throughout the years (see, e.g., some of the results of Lipton, Markakis, & Mehta, 2003, and Daskalakis

1. In the standard definition of an ϵ-Nash equilibrium in game theory, players tolerate losing expected *gains*, up to an ϵ amount, from not unilaterally deviating. There are other, mostly newer concepts of approximate Nash Equilibrium, such as "well-supported approximate equilibrium," which impose stronger requirements. This research note focuses on the *standard* definition.
2. The note was later posted online at `http://www.cis.upenn.edu/~mkearns/teaching/cgt/revised_approx_bnd.pdf` as part of a course on computational game theory taught by Prof. Michael Kearns during the Spring of 2003 at the University of Pennsylvania.
3. Note that Kearns et al. (2001) only considered the case of *binary* actions, hence the maximum number of actions m does not play a role in the results within the context of that paper.
4. Please see textbooks such as those of Vazirani (2001) and Williamson and Shmoys (2011) for formal definitions of different approximation schemes.

& Papadimitriou, 2008). Those particular results followed immediately from the improved discretization bound given in the old note (Ortiz, 2002), combined with previously known results from the *constraint-satisfaction-problem (CSP)* and graphical-models literature.

The present research note extends the old note (Ortiz, 2002) and shows how the improved discretization-size bounds for GGs, and some specializations, fall off immediately as corollaries of a theorem established here which holds for GMhGs. In particular, the current research note presents a result on the sufficient grid size for uniform discretization to contain an approximate Nash equilibrium in the following sense: for every exact Nash equilibrium of the game, its closest mixed strategy in ℓ_∞ distance in the induced grid is an approximate Nash equilibrium. As example corollaries of the main result, we obtain that for graphical games with largest neighborhood size k and in which the maximum number of actions (pure strategies) of each player is m, the sufficient grid size is $O(km/\epsilon)$, which implies a sufficient size of $O(nm/\epsilon)$ for standard normal-form games. We also obtain that for n-player m-action polymatrix games, the sufficient size is $O(m/\epsilon)$. Sparse discretization yields immediate computational results based on connection to algorithms for CSPs.

The remaining of the research note is organized as follows. Section 2 introduces preliminary notation and relevant concepts. Section 3 introduces the discretization scheme of primary interest here. Section 4 presents a corollary of the main sparse-discretization result within the context of standard graphical games in normal-form. Section 5 presents and discusses several results on polynomial and quasi-polynomial time algorithms for several interesting subclasses of graphical and normal-form games that fall off from that connection. Section 6 formally defines and connects GMhGs to other game classes, including previous game models in other areas both within and outside game theory, AI and computer science, among others. It then presents the sparse-discretization theorem in the context of GMhGs. That section also contains a brief discussion of classes of games proposed for practical security-and-defense-related applications in which the theorem has already proved useful to derive FPTAS algorithms.

2. Preliminaries

This section introduces the basic technical notation and concepts necessary to understand the upcoming technical sections of this research note.

2.1 Basic Notation

Denote by $x \equiv (x_1, x_2, \ldots, x_n)$ an n-dimensional vector and by

$$x_{-i} \equiv (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$

the same vector without component i. Similarly, for every set $S \subset [n] \equiv \{1, \ldots, n\}$, denote by $x_S \equiv (x_i : i \in S)$ the (sub-)vector formed from x using only components in S. Thus, if $S^c \equiv [n] - S$ denotes the complement of S, $x \equiv (x_S, x_{S^c}) \equiv (x_i, x_{-i})$ for every i. If $A_1, \ldots, A_n$ are sets, denote by $A \equiv \times_{i\in[n]} A_i$, $A_{-i} \equiv \times_{j\in[n]-\{i\}} A_j$ and $A_S \equiv \times_{j\in S} A_j$.

If $G = (V, E)$ is an undirected graph, then for each $i \in V$ denote by $\mathcal{N}_i \equiv \{j \mid (j, i) \in E\}$ the *neighbors of node/vertex* i in G, and $N_i \equiv \mathcal{N}_i \bigcup \{i\}$ the *neighborhood of node/vertex* i in G. Note that we have $i \notin \mathcal{N}_i$ but $i \in N_i$ for all $i \in V$.

2.2 Graphical Games in Local Normal-Form Payoff Representations

This section formally defines *graphical games (GGs)*, which are graphical models for compact representations of classical models in game theory (Kearns et al., 2001). GGs extend and generalize *normal-form games.*

Definition 1. *A* graphical game (GG) *consists of an* undirected graph $G = (V, E)$, [5] *where each node $i \in V$ in the graph corresponds to a player i in the game, and for each player i, we have a set of* actions *or pure strategies A_i and a local payoff hypermatrix/function $M'_i : A_{N_i} \to \mathbb{R}$, where N_i is the* neighborhood *of player i in G. The (global) payoff hypermatrix/function M_i of player i is such that, for each joint-action $x \in A \equiv A_V$, we have $M_i(x) \equiv M'_i(x_{N_i})$.*

Note that in a GG, the payoff that each individual player i receives when all players, including i, take joint-action/pure-strategy x is a *function of the joint-actions x_{N_i} of player i's neighborhood N_i only, thus conditionally independent of x_{V-N_i} given x_{N_i}*. It is convention to let $V = \{1, \ldots, n\} \equiv [n]$, so that $n \equiv |V|$. The representation size of each local payoff hypermatrix M'_i is $\Theta(|A_{N_i}|) = O(m^k)$, where $m \equiv \max_{i \in V} |A_i|$ and $k \equiv \max_{i \in V} |N_i|$. The representation size of the GG is $\Theta(\sum_{i \in V} |A_{N_i}|) = O(nm^k)$. If for all i we have $N_i = V$, then the GG is a standard *normal-form game*, also called strategic- or matrix-form game, which has a representation size $\Theta(n|A|) = O(nm^n)$. Said differently, a GG with a complete/fully-connected graph is equivalent to a normal-form game. Hence, a GG achieves considerable representation savings over normal-form games whenever $k \ll n$.

2.3 Solution Concepts

A *joint mixed strategy* $p \equiv (p_1, \ldots, p_n)$ in a game is formed from each individual *mixed strategy* $p_i \equiv (p_i(x_i) : x_i \in A_i)$ for player i, which is a probability distribution over the player's actions A_i (i.e., $p_i(x_i) \geq 0$ for all $x_i \in A_i$ and $\sum_{x_i \in A_i} p_i(x_i) = 1$). Denote by $\mathcal{P}_i \equiv \{\, p_i \mid p_i(x_i) \geq 0, \text{ for all } x_i \in A_i \text{ and } \sum_{x_i \in A_i} p_i(x_i) = 1\}$ the set of all possible mixed strategies of player i (i.e., all possible probability distributions over A_i). Similarly to the vector notation introduced above, for all i and any clique/set $S \subset V$, denote by p_{-i} and p_S the mixed strategies corresponding to all the players *except* i and all the players in clique S, respectively, so that $p \equiv (p_i, p_{-i}) \equiv (p_S, p_{V-S})$. A joint mixed strategy p induces a joint (product) probability distribution over the joint action space A, such that, for all $x \in A$, $p(x) \equiv \prod_{i \in V} p_i(x_i)$ is the probability, with respect to joint mixed strategy p, that joint action x is played. Using the standard abuse of notation in the literature, the *expected payoff* of player i with respect to joint mixed strategy p is simply denoted by $M_i(p) \equiv \sum_{x \in A} p(x) M_i(x)$.

Definition 2. *For any $\epsilon \geq 0$, a joint mixed-strategy p^* is called an* ϵ-Nash equilibrium *if for every player i, and for all $x_i \in A_i$, $M_i(p^*_i, p^*_{-i}) \geq M_i(x_i, p^*_i) - \epsilon$. That is, no player can increase its expected payoff more than ϵ by* unilaterally *deviating from its mixed strategy part p^*_i in the equilibrium, assuming the others play according to their respective parts p^*_{-i}. A* Nash equilibrium*, or more formally, a mixed-strategy Nash Equilibrium, is then a 0-Nash equilibrium, also referred to as an* exact Nash equilibrium *to differentiate it from its approximation counterpart.*

5. It is easy to extend the same result to GGs with *directed* graphs.

Note that, for all $i \in V$, and $p_{-i} \in \mathcal{P}_{-i}$, we have

$$\max_{p_i \in \mathcal{P}_i} M_i(p_i, p_{-i}) = \max_{x_i \in A_i} M_i(x_i, p_{-i})$$

if and only if $\max_{p_i \in \mathcal{P}_i} M_i(p_i, p_{-i}) \geq M_i(x'_i, p_{-i})$, for all $x'_i \in A_i$. [6] Note also that the *exact*-equilibrium conditions are invariant to affine transformations of the payoff functions. On the other hand, the *approximate*-equilibrium conditions are shift-invariant but not scale-invariant, because scaling the payoffs affects the approximation error term. [7] At the same time, note that the approximation error parameter ϵ is a single *global* value, constraining the largest gains from unilateral deviations of *any* player. Thus, implicit in the definition of approximate Nash equilibrium is the fact that all the payoffs have the same scale. Otherwise, the parameter ϵ is not very meaningful. We can bring all the players' payoff functions to the same scale in order to endow ϵ with a globally consistent meaning and interpretation over all the players. In the case of GGs with local payoff matrices represented in tabular/matrix/normal-form, the most typical and convenient scale normalization of the payoff values is such that, for each player $i \in V$, we have $\min_x M_i(x) = \min_{x_{N_i}} M'_i(x_{N_i}) = 0$ and $\max_x M_i(x) = \max_{x_{N_i}} M'_i(x_{N_i}) = 1$. That way, the parameter ϵ is meaningful and globally consistent with respect to the minimum and maximum payoff values of every player being 0 and 1, respectively. Because standard GGs use "tabular" representations, we do not lose generality, with respect to computation, by assuming that the maximum and the minimum local payoff values of each player are 0 and 1, respectively: we can compute them both *efficiently* (i.e., in time linear in the representation size of the game). In the case of GG generalizations however, such normalization is NP-hard. Section 6 revisits this point.

3. Discretization Schemes

The discretization scheme considered here is similar to that of Kearns et al. (2001), except that we allow for the possibility of different discretization sizes for the mixed strategies of players.

Definition 3. *In an* (individually-uniform) discretization scheme*, for each player i and each of the player's action x_i, the uncountable set $I = [0, 1]$ of possible value assignments to the probability $p_i(x_i)$ is approximated by a finite grid defined by the set $\widetilde{I}_i = \{0, \tau_i, 2\tau_i, \ldots, (s_i -$*

6. The fact that $\max_{x_i \in A_i} M_i(x_i, p_{-i}) \geq M_i(x'_i, p_{-i})$, for all $x'_i \in A_i$ proves one direction. For the reverse direction, first recall that the maximum of any linear function over a simplex is achieved at a corner of the simplex. In this case, because we are dealing with the probabily simplex $\mathcal{P}_i$ over A_i, each corner corresponds to an action $x_i \in A_i$. Let $x^*_i \in \arg\max_{x_i \in A_i} M_i(x_i, p_{-i})$. Then $M_i(x^*_i, p_{-i}) = \max_{p_i \in \mathcal{P}_i} M_i(p_i, p_{-i}) \geq \max_{x'_i \in A_i} M_i(x'_i, p_{-i}) = M_i(x^*_i, p_{-i})$, where the first equality follows from the stated poperty of the maximum of a linear function over a simplex, the inequality follows from the hypothesis, and last equality follows from the definition of x^*_i. That proves the reverse direction.

7. For each player i, let $C_i \in \mathbb{R}^+$ and $g_i : A_{-i} \to \mathbb{R}$. Let $C \equiv \max_i C_i$. Consider a new transformed game defined by transformed payoff matrices $\{\bar{M}_i\}$ of the original game: formally, for each player i, set $\bar{M}_i(x_i, x_{-i}) \equiv C_i \times M_i(x_i, x_{-i}) + g_i(x_{-i})$, for all $x_i \in A_i$ and $x_{-i} \in A_{-i}$. Using linearity of expectation, the fact that joint mixed strategies are product distributions, and simple algebra, it follows that, for all $p^* \in \mathcal{P}$, p^* is an ϵ-Nash equilibrium of the original game if and only if p^* is an $(C\epsilon)$-Nash equilibrium of the new transformed game. The last statement yields the property stated in the body of the paper about the *exact*-equilibrium conditions being invariant to affine transformations of the payoff functions because a 0-Nash equilibrium is an exact Nash equilibrium, by definition.

$1)\tau_i, 1\}$ *of values separated by the same distance* $\tau_i = 1/s_i$ *for some integer* s_i. *Thus the* discretization size *is* $|\widetilde{I}_i| = s_i + 1$. *Then, we would only consider mixed strategies* q_i *such that* $q_i(x_i) \in \widetilde{I}_i$ *for all* x_i, *and* $\sum_{x_i \in A_i} q_i(x_i) = 1$. *The* induced discretized space *of joint mixed strategies is* $\widetilde{I} \equiv \times_{i \in V} \widetilde{I}_i^{|A_i|}$, *subject to the individual normalization constraints stated in the last sentence.*

Kearns et al. (2001) and Ortiz and Kearns (2003) provide illustrative figures graphically showing examples resulting from the discretization scheme.

4. Sparse Discretization

The obvious question is, how small can we make s_i and still guarantee that there exists an ϵ-Nash equilibrium in the induced discretized space of joint mixed strategies? The following corollary provides a stronger answer for GGs: it provides values for the s_i's that guarantee that for *every* Nash equilibrium, its closest point in the induced grid is an ϵ-Nash equilibrium. An interesting aspect of the result is that s_i depends only on information local to player i's neighborhood: the number of actions $|A_i|$ available to player i and the largest number of neighbors $|\mathcal{N}_j|$ *of the neighbors* $j \in \mathcal{N}_i$ of player i (i.e., the quantity $\max_{j \in \mathcal{N}_i} |\mathcal{N}_j|$).

Note that the corresponding discretization bound provided in Kearns et al. (2001) in the context of GGs is *exponential* in the largest neighborhood size k. In contrast, the bound here is *linear* in k, a substantial reduction.

The corollary is a GG instantiation of a more general result on sparse discretization, given in Theorem 3, which holds for a broader class of GG generalizations. The statement and discussion of the theorem is in Section 6. The statement of the corollary uses notation introduced above.

Corollary 1. (Sparse Discretization for Graphical Games) *For any graphical game and any* $\epsilon > 0$, *a (individually-uniform) discretization with*

$$s_i = \left\lceil \frac{2\,|A_i|\,\max_{j \in \mathcal{N}_i} |\mathcal{N}_j|}{\epsilon} \right\rceil = O\left(\frac{m\,k}{\epsilon}\right)$$

for each player i *is sufficient to guarantee that for* every *exact Nash equilibrium of the game, its closest (in* ℓ_∞ *distance) joint mixed strategy in the induced discretized space is an* ϵ*-Nash equilibrium of the game.*

Remarks. It is important to recall that Kearns et al. (2001) also used ℓ_∞ distance in their original work. That notion of distance has remained standard for uniform sparse discretization, at least within the AI community. It is also well-known that for any arbitrary but fixed approximation parameter $\epsilon > 0$, even if a grid point corresponds to an ϵ-Nash equilibrium, there might not be any *exact* Nash equilibrium nearby.

5. Algorithmic Implications

This section presents the algorithmic implications of the sparse-discretization approach, borrowing heavily from standard results in AI, and highlighting their simplifying usefulness and powers.

5.1 Reductions to Consistency in CSPs

This section involves concepts from AI and graph theory. In the interest of space, the reader is referred to appropriate standard references (see, e.g., the textbooks of Russell & Norvig, 2003 for AI, Bollobas, 1979 for graph theory, and Dechter, 2003 for graph theory as used in AI).

Corollary 1 has several immediate computational consequences for the problem of approximate Nash equilibria in GGs with local payoff matrices represented in tabular form, and in turn, also for multi-player games represented in standard normal (tabular) form.

To simplify the presentation, let us assume that the payoff values are in $[0, 1]$, all the players in the GG have the same number of actions m and the largest neighborhood size in the graph of the game is k, so that the representation size of the GG is $O(nm^k)$. Under those conditions, the uniform discretization presented in Theorem 3 has size $s = s_i = O(mk/\epsilon)$ for all i.

5.1.1 Sparse-discretization GG-induced CSP

Once we introduce a discretization over the space of mixed strategies, then it is natural to formulate the problem of computing ϵ-Nash equilibria on the induced discretized space as a CSP, or more specifically in the case of GGs, as a special type of constraint network (Dechter, 2003). (In the interest of keeping this research note short, please see Russell & Norvig, 2003, or other introductory textbook on AI, for general information on CSPs. The presentation here contains only the CSP concepts necessary exclusively within the context of the topic of this research note.) Several researchers have taken this or related approaches either explicitly or implicitly (Kearns et al., 2001; Vickrey & Koller, 2002; Ortiz & Kearns, 2003; Soni, Singh, & Wellman, 2007). The CSP for the game has one variable, domain, and constraint for each player. Each variable corresponds to mixed strategy q_i for each player i. Each variable's domain corresponds to the discretized set $\widetilde{I}_i^m$ of mixed strategies for each player i, properly corrected to account for normalization. [8] The *approximate best-response equilibrium conditions* are the following: for each player i, each constraint function (table) $c_i : \widetilde{I}_i^{mk} \to \{0, 1\}$ is defined so that, for all $q_{N(i)} \in \widetilde{I}_i^{mk}$, $c_i(q_{N(i)}) = 1$ if and only if $M_i(q_i, q_{-i}) \geq \max_{x_i' \in A_i} M_i(x_i', q_{-i}) - \epsilon$. Each constraint has arity at most k and encodes the approximate best-response equilibrium conditions, each represented in tabular form using $s^{mk} = O((mk/\epsilon)^{mk})$ bits. The transformation takes time $O(poly(n, (mk/\epsilon)^{mk}))$ and the size of the resulting CSP is $T = O(n(mk/\epsilon)^{mk})$.

For graphical games, it is natural to consider the number of players n as being the "free" parameter of the representation (i.e., the main parameter of interest in terms of problem input size). Indeed, Kearns et al. (2001) considered the case of $m = 2$ only. Hence, if $mk \log(mk) = O(\log(n))$ and $\epsilon = n^{\Omega(-1/(mk))} = (mk)^{-\Omega(1)} = (\log n)^{-\Omega(1)}$, then both the time to perform the CSP transformation and its resulting representation size are polynomial in the representation size of the game (i.e., polynomial in the number of players). Similarly, if m and k are bounded (by a constant, independent of n), then the representation size of the game is linear in n and the running time of the transformation is polynomial in n and $1/\epsilon$. Note that this is a natural restriction on the game parameters because otherwise the

8. Recall that we only consider $q_i \in \widetilde{I}_i^m$ such that $\sum_{x_i \in A_i} q_i(x_i) = 1$.

representation size would be exponential in the number of players, thus defeating the main purpose of the GG representation in the first place: *succinctness.*

At this point, we can apply any of a large number of existing off-the-shelf techniques for solving the induced game-CSP. We can also apply techniques such as **NashProp** (Ortiz & Kearns, 2003) that take advantage of the particular properties of the game best-response constraints. Instead, in Sections 5.1.2 and 5.1.4, we will see how standard results for solving constraint networks (Dechter, 2003) lead immediately to simple derivations of algorithmic results for GGs.

5.1.2 An approach based on the GG-induced CSP hypergraph

An approach to solving CSPs in AI works on the *hypergraph induced by the game CSP* (Gottlob, Leone, & Scarcello, 2001). If T is the representation size of the game-CSP, w is the hypertree width of the hypergraph, and the corresponding hypertree decomposition for the CSP has been computed, then solving the CSP takes time $O(T^{w+1}\log(T))$ (Gottlob, Leone, & Scarcello, 2000, 2002; Gottlob et al., 2001) (see also page 158 of "Bibliographical and Historical Notes" Section in Chapter 5 of Russell & Norvig, 2003). As an anonymous reviewer pointed out, for the special case of GGs, solving the game-CSP given its hypertree decomposition takes time $O(n\ \rho^{w+1}\log\rho)$, where $\rho = O\left(s^{mk}\right)$ is the maximum size of the tabular representation encoding any constraint of the game-CSP. In the case of GGs, we can compute the hypertree decomposition that the algorithm would use in time $O(n^{2w+2})$.

The following theorem summarizes the discussion. As a preamble to a discussion of primal graphs and treewidth later in the section, note that it is known that a CSP might have hypergraphs with bounded hypertree width, but whose *primal* graph has *unbounded* treewidth. However, the treewidth *always bounds* the hypertree width. Thus, the restriction that the hypertree width be bounded by a constant may not be as limiting to the application of the results as it first appears.

Theorem 1. *There exists an algorithm that, given as input a number $\epsilon > 0$ and a GG with n players, maximum neighborhood size k, and maximum number of actions m, and whose corresponding CSP has a hypergraph with hypertree width w, computes an ϵ-Nash equilibrium of the GG in time $O\left(n^{2w+2} + nmk(mk/\epsilon)^{mk(w+1)}\log(mk/\epsilon)\right) = n^{O(w)} + n[(mk/\epsilon)^{mk}]^{O(w)} = [n\,(mk/\epsilon)^{mk}]^{O(w)}$.*

The following corollary characterizes the computational complexity of the approximation schemes resulting from instances of the last theorem.

Corollary 2. *There exists an algorithm that, given as input a GG with corresponding hypergraph of hypertree width w bounded by a constant independent of the number of players n, with a logarithmic function of n restricting the maximum number of actions m and the maximum neighborhood size k as $mk\log(mk) = O(\log(n))$, and given $\epsilon = n^{\Omega(-1/(mk))} = (mk)^{-\Omega(1)} = (\log n)^{-\Omega(1)}$, outputs an ϵ-Nash equilibrium of the game in time polynomial in the representation size of the input. If, in particular, both m and k are bounded by constants independent of n, then the algorithm runs in polynomial time in n and $1/\epsilon$, for any $\epsilon > 0$; hence, the algorithm is a* fully polynomial time approximation scheme (FPTAS). *If, instead, the expression constraining m and k as a logarithmic function of*

n, $mk\log(mk) = O(\log(n))$, holds, and $w = polylog(n)$, then the algorithm is a quasi-polynomial time approximation scheme (quasi-PTAS).

5.1.3 A side note on normal-form games

For normal-form games, we have $k = n$ and $w = 1$. [9] This leads to the following corollary.

Corollary 3. *There exists a quasi-PTAS for computing an ϵ-Nash equilibrium of n-player m-action normal-form games with $m = O(poly(n))$ that runs in time $N^{O(polylog(N)\log(1/\epsilon))} = \left(\frac{1}{\epsilon}\right)^{O(polylog(N))}$, where $N = n^{\Theta(n)}$ is the representation size of the game. If, in particular, m is bounded by a constant independent of n, then the running time is $N^{O(\log \frac{\log(N)}{\epsilon})}$, where $N = 2^{\Theta(n)}$ is the corresponding representation size of the game.*

We can also obtain the same result by using the sparse-support approach of Lipton et al. (2003), even if $m = 2^{O(n)}$, but the dependence is *exponential* in $1/\epsilon^2$ (i.e., $N^{O(polylog(N)(1/\epsilon)^2)}$; or $N^{O(\frac{\log(N)}{\epsilon^2})}$, if m is bounded by a constant). The result of the last corollary for the case of m fixed or bounded by a constant is stated by Daskalakis and Papadimitriou (2008). The same result for m fixed or bounded by a constant also follows from Theorem 4 of Kearns (2007). While no formal proof appears for Theorem 4 of Kearns (2007), the theorem follows immediately from the proof in the original unpublished note (Ortiz, 2002).

It is important to emphasize that in the case of normal-form games, we could have obtained the result directly by using an exhaustive search over the induced grid over mixed strategies. That is essentially what the algorithm referred to in the corollary reduces to in this case. Hence, we could output not just one ϵ-Nash equilibrium, but *all* ϵ-Nash equilibria in the induced grid in the worst-case running time given in the corollary. The algorithms based on the sparse-support approach can only guarantee to output *one* ϵ-Nash equilibrium among all mixed strategies of a given maximum support.

5.1.4 An approach based on the GG-induced CSP primal graph

As it has been common knowledge in the graphical-models community for quite a while now (see, e.g., the textbooks of Dechter, 2003; Russell & Norvig, 2003, and the references therein, for a recent account of previous work in this area), another approach to solving a (discrete) CSP is to build a clique (or join) tree from the *primal graph* of the CSP. In the case of GGs, the primal graph is the graph of the game-CSP created by forming cliques of every neighborhood. Then, we can apply a dynamic programming (or message-passing) algorithm on the clique tree. The graphical model for the CSP, consisting of the primal graph and a (tabular) representation of the constraint information, is called a *constraint network* (Dechter, 2003; Russell & Norvig, 2003). Once the clique tree is built for the constraint network, the running time is linear in the number of nodes of the join tree and exponential in the size of the largest clique associated to a node in the clique tree. In particular, if the primal graph has treewidth w', the largest clique associated to the optimal clique tree is $w' + 1$ (Dechter, 2003). Given the optimal clique tree, the running time to

9. For the case of normal-form games, each node in any hypertree decomposition of the game-CSP corresponds to the best-response constraint of exactly one player. Thus, by definition, we have that the hypertree width is one in this case.

solve the CSP is $O((|V|+|\mathcal{E}|)|D|^{w'+1})$, where V is the set of nodes of the primal graph (i.e., the number of variables of the CSP), D is the largest (finite) domain of any variable of the (discrete) CSP, and $\mathcal{E}$ its (finite) set of constraint functions (Dechter, 2003). In addition, we can build the clique tree in time $2^{O(w')}|V|\log(|V|)$ (Reed, 1992; Becker & Geiger, 2001). Hence the well-known result that if the primal graph of a CSP has a treewidth w' that is logarithmic in the number of nodes $|V|$, then we can solve the CSP in polynomial time if the CSP is represented in tabular form. The next theorem (Theorem 2) and corollary (Corollary 4) follow from the application of those previously known results just described for solving constraint networks and other related graphical models. Note that the treewidth w'' of the original graph of the game is always no smaller than the hypertree width w of its hypergraph (Gottlob, Greco, & Scarcello, 2005). In addition, the GG's primal graph treewidth w' satisfies $w' \leq (w''+1)k$ (Daskalakis & Papadimitriou, 2006). Thus, the interesting bounds in the hypertree case are as given in the corollary. Also, this means that the results can be easily extended to GGs with (original) graphs that have $O(\log(n))$ treewidth as long as k is bounded. Finally, if a graph with n nodes has treewidth w', then the graph has at most $(w'+1)n$ edges (see, e.g., the work of Becker & Geiger, 2001). Because the number of edges of a GG primal graph is $O(k^2 n)$, $w' = O(\log(n))$ implies $k = O(\sqrt{\log(n)})$.

Theorem 2. *There exists an algorithm that, given as input a number $\epsilon > 0$ and an n-player m-action GG with maximum neighborhood size k and primal-graph treewidth w', computes an ϵ-Nash equilibrium of the game in time $2^{O(w')}n\log(n) + n[(mk/\epsilon)^{mk}]^{O(w')}$.*

Corollary 4. *There exists a PTAS for computing an approximate Nash equilibria in n-player GGs with bounded maximum number of actions, bounded neighborhood size, and primal-graph treewidth $w' = O(\log(n))$.*

The new discretization bounds also provide significant improvements on the representation results and running times for **NashProp** and its variants (Kearns et al., 2001; Ortiz & Kearns, 2003). The new bounds may also provide improvements to previous discretization-based schemes for computing ϵ-Nash equilibria in other similar models, such as those in the work of Singh, Soni, and Wellman (2004) and Soni et al. (2007).

6. Graphical Multi-hypermatrix Games: Generalizing Graphical Games

This section introduces *graphical multi-hypermatrix games (GMhGs)*, a class of games that extends and generalizes GGs while capturing many classical game-theoretic model representations, as discussed below. This class of games is not some theoretical concoction: they are not only convenient in their generality, covering a large number of existing models, but also practical. Indeed, Yu and Berthod (1995) used the same type of games to establish an equivalence between *local* maximum-*a-posteriori* (MAP) inference in Markov random fields and the Nash equilibria of the induced game. Ortiz (2015) provides a brief summary and references to other examples of subclasses of GMhGs applied in artificial intelligence and other fields. We briefly discuss another application in AI after stating Theorem 3, the core theorem on sparse discretization in GMhGs.

Definition 4. *A* graphical multi-hypermatrix game (GMhG) *is defined by a set V of n players, and for each player $i \in V$, a set of* actions*, or pure strategies, A_i; a set $\mathcal{C}_i \subset 2^V$*

of cliques*, or hyperedges, such that if* $C \in \mathcal{C}_i$ *then* $i \in C$*; and a set* $\{M'_{i,C} : A_C \to \mathbb{R} \mid C \in \mathcal{C}_i\}$ *of* local-clique payoff matrices*. For each player* $i \in V$*, the sets* $N(i) \equiv \cup_{C \in \mathcal{C}_i} C$ *and* $\mathcal{N}_i \equiv \{j \in V \mid i \in N(j), j \neq i\}$ *are the clique of players affecting* i*'s payoff including* i *(i.e.,* i*'s neighborhood) and those affected by* i *not including* i*, respectively. The* local *and* global payoff matrices $M'_i : A_{N(i)} \to \mathbb{R}$ *and* $M_i : A \to \mathbb{R}$ *of* i *are (implicitly) defined as* $M'_i(x_{N(i)}) \equiv \sum_{C \in \mathcal{C}_i} M'_{i,C}(x_C)$ *and* $M_i(x) \equiv M'_i(x_{N(i)})$*, respectively.*

Connections to other game classes. If for each player, each clique set is a singleton, we obtain a (possibly asymmetric) *graphical game*, and the single clique in the set defines the neighborhood of the player (i.e., in that case, $\mathcal{C}_i = \{N(i)\}$ for all i). If, in addition, each clique is the complete set of players, then the game is a standard *normal-form game*, also called strategic- or matrix-form game (i.e., in that case, $N(i) = V$ for all i). A GMhG becomes a classical, standard *polymatrix game* (Janovskaja, 1968) if for each player i, $\mathcal{C}_i = \{\{i, j\} \mid j \in V, j \neq i\}$, which is the set of cliques of pairs of nodes involving the player and every other player. [10] In contrast to *hypergraphical games* (Papadimitriou & Roughgarden, 2008), a GMhG is more expressive, in part because a GMhG does not require that the same "sub-game" (i.e., local-clique payoff hypermatrix) be shared among all players in the clique of the "sub-game." For example, a local-clique payoff hypermatrix may appear in the summation defining the local payoff hypermatrix of exactly one player. A GMhG has the polynomial intersection property and thus a polynomial correlated equilibrium scheme (Papadimitriou & Roughgarden, 2008).

Representation size. The representation size of a GMhG is $\Theta\left(\sum_{i \in V} \sum_{C \in \mathcal{C}_i} \prod_{j \in C} |A_j|\right) = O(n\,\kappa\,m^c)$, where $\kappa \equiv \max_{i \in V} |\mathcal{C}_i|$ and $c \equiv \max_{i \in V} \max_{C \in \mathcal{C}_i} |C|$. Hence, the size is dominated by the representation of the local-clique payoff matrices, which are each of size exponential in their respective clique size. However, this representation size could be considerably smaller than for a graphical game, which is exponential in the neighborhood size. For example, if for each i, we have $|\mathcal{C}_i| \leq k$, and for each $C \in \mathcal{C}_i$, we have $|C| = 2$, then the GMhG becomes a *graphical polymatrix game*, with representation size $O(n\,k\,m^2)$, *linear* in the maximum number of neighbors k, compared to $O(nm^k)$ for a standard graphical game with "tabular" representations, which is *exponential* in k.

Payoff Scale. Normalizing the payoff of a GG in standard local strategic/normal-form takes linear time in the representation size of the game. This is because we can find the minimum and maximum local payoff values for each local payoff hypermatrix simply by going over each payoff value in the hypermatrix in sequence. [11] However, such an approach is intractable in GMhGs in general. Denote the maximum and minimum payoff values for each player $i \in V, u_i \equiv \max_{x_{N_i}} \sum_{C \in \mathcal{C}_i} M'_{i,C}(x_C)$ and $l_i \equiv \min_{x_{N_i}} \sum_{C \in \mathcal{C}_i} M'_{i,C}(x_C)$, respectively. Computing *both* u_i and l_i is NP-hard. To see this, note that u_i and l_i are the result of a max and min operation over an additive function of the set of the player's hyperedges and its possible joint-actions. It is easy to reduce the problem of finding a solution to an arbitrary contraint network to that of computing *both* u_i and l_i for each player i. Hence, in general,

10. Note that this is the standard definition of polymatrix games. It requires *symmetry* in the hyperedges: for all pair of players $i, j \in V, i \neq j, \{i, j\} \in \mathcal{C}_i \cap \mathcal{C}_j$.
11. Recall that the "hypermatrix" of each player in a GG, as originally defined, reduces to a standard multi-dimensional matrix that is exponential in the size of the player's neighborhood.

we do not have much of a choice but to assume that the payoffs of all players are in the same scale. That way, using a global approximation-quality value ϵ is meaningful. Here, we compute the individual maximum and minimum values of each hypermatrix payoff of each player as a way to set up the sparse uniform-discretization.

Some additional notation is necessary before stating the theorem, whose proof appears in Appendix A. Denote by $u_{i,C} \equiv \max_{x_C \in A_C} M'_{i,C}(x_C)$ and $l_{i,C} \equiv \min_{x_C \in A_C} M'_{i,C}(x_C)$ the largest and smallest payoff values achieved by the local-grid payoff hypermatrix $M'_{i,C}$, respectively; and by $R_{i,C} \equiv u_{i,C} - l_{i,C}$ its *largest range* of values.

Theorem 3. (Sparse Discretization for Graphical Multi-hypermatrix Games) *For any graphical multi-hypermatrix game and any ϵ such that*

$$0 < \epsilon \leq 2 \min_{i \in V} \frac{\sum_{C \in \mathcal{C}_i} R_{i,C} \left(|C| - 1\right)}{\max_{C' \in \mathcal{C}_i} |C'| - 1},$$

a (uniform) discretization with

$$s_i = \left\lceil \frac{2 \left|A_i\right| \max_{j \in \mathcal{N}_i} \sum_{C \in \mathcal{C}_j} R_{j,C} \left(|C| - 1\right)}{\epsilon} \right\rceil$$

for each player i is sufficient to guarantee that for every *exact Nash equilibrium of the game, its closest (in ℓ_∞ distance) joint mixed strategy in the induced discretized space is an ϵ-Nash equilibrium of the game.*

The generalized bound has already proved useful to derive and analyze dynamic-programming algorithms for computing approximate Nash equilibria in non-trivial special cases of *interdependent defense (IDD) games* with specific, practical graph structures. Chan, Ceyko, and Ortiz (2012) introduced IDD games to model, study, and analyze security and defense mechanisms for deterrence in network-structured interdependent security systems. In an IDD game, each node in the graph represents an internal agent or "site" in the network graph under the risk of a deliberate attack from an external agent, also called the "attacker," as a model abstraction. Each site individually and voluntarily decides whether it is cost-effective to protect itself against a potential attack. One objective is the potential to study the effect of minimal interventions in overall system security for deterrence purposes. Chan and Ortiz (2015) use a corollary of Theorem 3 to design an FPTAS for computing an approximate Nash equilibrium in IDD games with directed-tree-structured graphs over the "sites" in the network. [12] The key point of the result is that the number of actions/pure-strategies of the attacker is *not constant*, but *linear* in the number of players (i.e., $O(n)$, using the notation here). Typically, under such circumstances, naively applying an algorithm such as **TreeProp** or **NashProp** would require worst-case running-time *exponential* in $n \log n$ (i.e., $O(n^n)$). And that is just to compute each message-passing multi-dimensional table. In addition, each table itself would require space *exponential* in n (i.e., $\Omega(2^n)$), even under the special cases we considered.

Unlike for standard GGs, the broader algorithmic and computational implications of the sparse discretization in GMhGs remain open.

12. This result extends easily to IDD games with graphs of bounded (hyper)tree width, by following the standard approach used to derive algorithms based on serial dynamic programming in constraint (Dechter, 2003) and probabilistic (Koller & Friedman, 2009) graphical models in AI.

Acknowledgements

I would like to give warm thanks to the Action Editor handling a previous version of this research note Amy Greenwald, former JAIR Editor-in-Chief Dan Roth, the Action Editor for this final version Edith Elkind, and the anonymous reviewers for their support, feedback, comments, and suggestions. Particular thanks go to the anonymous reviewer who pointed out the refined expression for the special case of GGs of the running time of state-of-the-art algorithms for solving CSPs given their hypertree decomposition, leading to the refined expressions given in Theorem 1. I would also like to thank Michael Kearns for posting and acknowledging my original research note, as well as my handwritten lecture notes on graphical games, in the webpage for his Spring 2003 course on computational game theory at the University of Pennsylvania (`http://www.cis.upenn.edu/~mkearns/teaching/cgt/`). This manuscript was supported in part by National Science Foundations Faculty Early Career Development Program (CAREER) Award IIS-1643006 (transferred from IIS-1054541).

Appendix A. Proof of Theorem 3

This appendix presents a derivation for the most general case of GMhGs. It is important to note that the derivation of the proof is considerably simpler if one specifically considers 2-action GGs, as Kearns et al. (2001) originally did.

To simplify notation, given any joint mixed-strategy (i.e., a product distribution) p, for all $B \subset V$, and $x_B \in A_B$, we denote by $p(x_B) \equiv \prod_{i \in B} p_i(x_i) = \sum_{x_{-B}} p(x_B, x_{-B})$ the joint mixed-strategy over players in B only (i.e., marginal product-distributions of p over the joint-actions of players in B). Let p and q be two joint mixed strategies and, for each player i and each action x_i, denote by $\Delta_i(x_i) \equiv p_i(x_i) - q_i(x_i)$. In a slight abuse of notation, let $\Delta(x_S) \equiv \prod_{k \in S} \Delta_k(x_k)$.

The following very simple lemma is a cornerstone of the proof.

Lemma 1. (Product-Distribution Differences) *For any clique $B \subset V$ of players, for any clique joint action x_B,*

$$p(x_B) - q(x_B) = \sum_{S \in 2^B - \emptyset} \Delta(x_S)\, q(x_{B-S}) \; .$$

Proof. The lemma follows by applying a binomial expansion:

$$\begin{aligned} p(x_B) =& \prod_{j \in B} \left(q_j(x_j) + \Delta_j(x_j)\right) \\ =& \sum_{S \in 2^B} \Delta(x_S)\, q(x_{B-S}) \\ =& q(x_B) + \sum_{S \in 2^B - \emptyset} \Delta(x_S)\, q(x_{B-S}) \; . \end{aligned}$$

□

To further simplify the presentation of the proof, it is convenient to introduce a slight abuse of notation: for all, $i \in C, C \in \mathcal{C}_i, B, S \subset C, B \cap S = \emptyset, x_S \in A_S, p_{C-B-S} \in \mathcal{P}_{C-B-S}$, let $M'_{i,C}(x_S, \Delta_B, p_{C-B-S}) \equiv \sum_{x_B \in A_B} \Delta(x_B) M'_{i,C}(x_S, x_B, p_{C-B-S})$.

The following useful claim follows immediately from the last lemma of joint product distribution differences (Lemma 1).

Claim 1. *Under the conditions of Lemma 1, for all $i \in V$, $C \in \mathcal{C}_i$, $B \subset C$, $x_B \in A_B$ and $p_{C-B}, q_{C-B} \in \mathcal{P}_{C-B}$, we have*

$$M'_{i,C}(x_B, p_{C-B}) - M'_{i,C}(x_B, q_{C-B}) = \sum_{S \in 2^{C-B} - \emptyset} M'_{i,C}(x_B, \Delta_S, q_{C-B-S}) \; .$$

Proof. Applying the last lemma on the differences between product distributions (Lemma 1), we have

$$\begin{aligned} M'_{i,C}(x_B, p_{C-B}) - M'_{i,C}(x_B, q_{C-B}) &= \sum_{x_{C-B}} \left[\sum_{S \in 2^{C-B} - \emptyset} \Delta(x_S) q(x_{C-B-S}) \right] M'_{i,C}(x_C) \\ &= \sum_{S \in 2^{C-B} - \emptyset} \sum_{x_S} \Delta(x_S) \sum_{x_{C-B-S}} q(x_{C-B-S}) M'_{i,C}(x_C) \\ &= \sum_{S \in 2^{C-B} - \emptyset} M'_{i,C}(x_B, \Delta_S, q_{C-B-S}) \; . \end{aligned}$$

□

Using some algebra we can show another useful claim.

Claim 2. *Under the conditions of Lemma 1, for all $i \in V$, and $C \in \mathcal{C}_i$, we have*

$$\sum_{S \in 2^C - \emptyset} M'_{i,C}(\Delta_S, q_{C-S}) = \sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) \; .$$

Proof. First note that we can decompose the left-hand side of the equation in the claim as

$$\begin{aligned} \sum_{S \in 2^C - \emptyset} M'_{i,C}(\Delta_S, q_{C-S}) = &\sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(q_i, \Delta_B, q_{C-B-\{i\}}) + \\ &\sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(\Delta_i, \Delta_B, q_{C-B-\{i\}}) \; . \end{aligned}$$

Now note that, using the definition of Δ_i, we have

$$M'_{i,C}(\Delta_i, \Delta_B, q_{C-B-\{i\}}) = M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) - M'_{i,C}(q_i, \Delta_B, q_{C-B-\{i\}}) \; .$$

The claim follows after making the appropriate substitution and simplifying:

$$\begin{aligned} \sum_{S \in 2^C - \emptyset} M'_{i,C}(\Delta_S, q_{C-S}) = &\sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(q_i, \Delta_B, q_{C-B-\{i\}}) + \\ &\sum_{B \in 2^{C-\{i\}} - \emptyset} (M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) - M'_{i,C}(q_i, \Delta_B, q_{C-B-\{i\}})) \\ = &\sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) \; . \end{aligned}$$

□

Suppose p is a Nash equilibrium of the game, which must exist by Nash's Theorem (Nash, 1951). Applying the last two claims above, we obtain

$$\begin{aligned}
M_i(p) =& M_i(q) + \sum_{C \in \mathcal{C}_i} \sum_{S \in 2^C - \emptyset} M'_{i,C}(\Delta_S, q_{C-S}) \\
=& M_i(q) + \sum_{C \in \mathcal{C}_i} \sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) \\
\geq& \max_{x'_i} M_i(x'_i, p_{-i}) \\
=& \max_{x'_i} \left[M_i(x'_i, q_{-i}) + \sum_{C \in \mathcal{C}_i} \sum_{B \in 2^{C-\{i\}} - \emptyset} M'_{i,C}(x'_i, \Delta_B, q_{C-B-\{i\}}) \right].
\end{aligned}$$

Rearranging and simplifying, we obtain the following expression:

$$\begin{aligned}
M_i(q) \geq& \max_{x'_i} M_i(x'_i, q_{-i}) + \\
& \sum_{C \in \mathcal{C}_i} \sum_{B \in 2^{C-\{i\}} - \emptyset} \left(M'_{i,C}(x'_i, \Delta_B, q_{C-B-\{i\}}) - M'_{i,C}(p_i, \Delta_B, q_{C-B-\{i\}}) \right).
\end{aligned}$$

Let q be the closest (in ℓ_∞ distance) joint mixed strategy in $\widetilde{I}$, defined using sizes s_i as given in the statement of the theorem, to exact Nash equilibrium p. Hence, we have

$$|\Delta_i(x_i)| \leq \frac{\epsilon}{2 |A_i| \max_{j \in \mathcal{N}_i} \sum_{C \in \mathcal{C}_j} R_{j,C} (|C| - 1)}.$$

Consider the conditional expected hypermatrix payoff difference in parenthesis within the innermost summation inside the maximization of the equilibrium condition above. Noting that

$$|M'_{i,C}(x'_i, x_B, q_{C_{-i,B}}) - M'_{i,C}(p_i, x_B, q_{C_{-i,B}})| \leq R_{i,C},$$

we obtain the following lower bound on that innermost summation:

$$\begin{aligned}
\left(M'_{i,C}(x'_i, \Delta_B, q_{C_{-i,B}}) - M'_{i,C}(p_i, \Delta_B, q_{C_{-i,B}}) \right) \geq& \sum_{x_B} \left[\prod_{k \in B} |\Delta_k(x_k)| \right] (-R_{i,C}) \\
=& - R_{i,C} \sum_{x_B} \prod_{k \in B} |\Delta_k(x_k)|. \qquad (1)
\end{aligned}$$

We can upper-bound the last factor in the right-hand side of the last expression as

$$\begin{aligned}
\sum_{x_B} \prod_{k \in B} |\Delta_k(x_k)| &= \prod_{k \in B} \sum_{x_k} |\Delta_k(x_k)| \\
&\leq \prod_{k \in B} \sum_{x_k} \frac{\epsilon}{2\,|A_k|\,\max_{j \in \mathcal{N}_k} \sum_{C' \in \mathcal{C}_j} R_{j,C'}\,(|C'|-1)} \\
&= \prod_{k \in B} \frac{\epsilon}{2\,\max_{j \in \mathcal{N}_k} \sum_{C' \in \mathcal{C}_j} R_{j,C'}\,(|C'|-1)} \\
&\leq \prod_{k \in B} \frac{\epsilon}{2 \sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \\
&= \left(\frac{\epsilon}{2 \sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \right)^{|B|} .
\end{aligned}$$

Thus, using the resulting lower bound on the expression given in (1), we obtain

$$M_i(q) \geq \max_{x'_i} M_i(x'_i, q_{-i}) - \sum_{C \in \mathcal{C}_i} \sum_{B \in 2^{C-\{i\}} - \emptyset} R_{i,C} \left(\frac{\epsilon}{2 \sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \right)^{|B|} .$$

The second term in the right-hand side of the last expression equals

$$\begin{aligned}
&\sum_{C \in \mathcal{C}_i} R_{i,C} \sum_{B \in 2^{C-\{i\}} - \emptyset} \left(\frac{\epsilon}{2 \sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \right)^{|B|} \\
&= \sum_{C \in \mathcal{C}_i} R_{i,C} \left[\left(1 + \frac{\epsilon}{2 \sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \right)^{|C|-1} - 1 \right] \\
&\leq \sum_{C \in \mathcal{C}_i} R_{i,C} \left[\frac{\epsilon\,(|C|-1)}{\sum_{C' \in \mathcal{C}_i} R_{i,C'}\,(|C'|-1)} \right] = \epsilon\,.
\end{aligned}$$

The last inequality follows from using the upper-bound condition on ϵ given in the statement of the theorem and applying the well-known inequality, $1 + z \leq \exp(z) \leq 1 + z + z^2$ for $|z| < 1$ (Cormen, Leiserson, & Rivest, 1990). This completes the proof of the theorem.

References

Becker, A., & Geiger, D. (2001). A sufficiently fast algorithm for finding close to optimal clique trees. *Artificial Intelligence*, *125*(1-2), 3 – 17.

Bollobas, B. (1979). *Graph Theory*. Springer.

Chan, H., Ceyko, M., & Ortiz, L. E. (2012). Interdependent defense games: Modeling interdependent security under deliberate attacks. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, UAI'12, pp. 152–162, Arlington, Virginia, United States. AUAI Press.

Chan, H., & Ortiz, L. (2015). Computing Nash equilibrium in interdependent defense games. In *AAAI Conference on Artificial Intelligence*.

Cormen, T., Leiserson, C., & Rivest, R. (1990). *Introduction to Algorithms*. The MIT Press, Cambridge, MA.

Daskalakis, C., Goldberg, P. W., & Papadimitriou, C. H. (2009). The complexity of computing a Nash equilibrium. *Commun. ACM*, *52*(2), 89–97.

Daskalakis, C., & Papadimitriou, C. H. (2006). Computing pure Nash equilibria in graphical games via Markov random fields. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, EC '06, pp. 91–99, New York, NY, USA. ACM.

Daskalakis, C., & Papadimitriou, C. H. (2008). Discretized multinomial distributions and nash equilibria in anonymous games. In *Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '08, pp. 25–34, Washington, DC, USA. IEEE Computer Society.

Dechter, R. (2003). *Constraint Processing*. Morgan Kaufmann.

Gottlob, G., Greco, G., & Scarcello, F. (2005). Pure Nash equilibria: Hard and easy games. *JAIR*, *24*, 357–406.

Gottlob, G., Leone, N., & Scarcello, F. (2000). A comparison of structural CSP decomposition methods. *Artificial Intelligence*, *124*(2), 243 – 282.

Gottlob, G., Leone, N., & Scarcello, F. (2001). Hypertree decompositions: A survey. In Sgall, J., Pultr, A., & Kolman, P. (Eds.), *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings*, pp. 37–57, Berlin, Heidelberg. Springer Berlin Heidelberg.

Gottlob, G., Leone, N., & Scarcello, F. (2002). Hypertree decompositions and tractable queries. *Journal of Computer and System Sciences*, *64*, 579–627.

Janovskaja, E. B. (1968). Equilibrium situations in multi-matrix games. *Litovsk. Mat. Sb.*, *8*, 381–384.

Kearns, M. (2007). Graphical games. In Nisan, N., Roughgarden, T., Tardos, E., & Vazirani, V. V. (Eds.), *Algorithmic Game Theory*, chap. 7, pp. 159–180. Cambridge University Press.

Kearns, M. J., Littman, M. L., & Singh, S. P. (2001). Graphical models for game theory. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, UAI '01, pp. 253–260, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Lipton, R. J., Markakis, E., & Mehta, A. (2003). Playing large games using simple strategies. In *Proceedings of the 4th ACM Conference on Electronic Commerce*, EC '03, pp. 36–41, New York, NY, USA. ACM.

Montanari, U. (1974). Networks of constraints: Fundamental properties and applications to picture processing. *Information Sciences*, *7*, 95 – 132.

Nash, J. (1951). Non-cooperative games. *Annals of Mathematics*, *54*, 286–295.

Ortiz, L. E. (2002). Approximation bound refinement of KLS. Unpublished Note, `http://www.cis.upenn.edu/~mkearns/teaching/cgt/revised_approx_bnd.pdf`.

Ortiz, L. E. (2015). Graphical potential games. *CoRR*, *abs/1505.01539*.

Ortiz, L. E., & Kearns, M. (2003). Nash propagation for loopy graphical games. In Becker, S., Thrun, S., & Obermayer, K. (Eds.), *Advances in Neural Information Processing Systems 15*, pp. 817–824. MIT Press.

Papadimitriou, C. H., & Roughgarden, T. (2008). Computing correlated equilibria in multi-player games. *J. ACM*, *55*(3), 14:1–14:29.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems. Networks of Plausible Inference*. Morgan Kaufmann, Palo Alto, CA.

Reed, B. A. (1992). Finding approximate separators and computing tree width quickly. In *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, pp. 221–228, New York, NY, USA. ACM.

Russell, S. J., & Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson Education.

Singh, S., Soni, V., & Wellman, M. (2004). Computing approximate Bayes-Nash equilibria in tree-games of incomplete information. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, EC '04, pp. 81–90, New York, NY, USA. ACM.

Soni, V., Singh, S., & Wellman, M. P. (2007). Constraint satisfaction algorithms for graphical games. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '07, pp. 435–442, New York, NY, USA. ACM.

Vazirani, V. V. (2001). *Approximation Algorithms*. Springer.

Vickrey, D., & Koller, D. (2002). Multi-agent algorithms for solving graphical games. In *Eighteenth National Conference on Artificial Intelligence*, pp. 345–351, Menlo Park, CA, USA. American Association for Artificial Intelligence.

Waltz, D. (1975). Generating semantic descriptions from drawings of scenes with shadows. In Winston, P. H. (Ed.), *The Psychology of Computer Vision*, pp. 19–91. McGraw-Hill, New York.

Williamson, D. P., & Shmoys, D. B. (2011). *The Design of Approximation Algorithms*. Cambridge.

Yu, S., & Berthod, M. (1995). A game strategy approach for image labeling. *Computer Vision and Image Understanding*, *61*(1), 32 – 37.