Computing the Optimal Game

Markus Brill
Dept. of Computer Science
Duke University
Durham, NC, USA
brill@cs.duke.edu

Rupert Freeman
Dept. of Computer Science
Duke University
Durham, NC, USA
rupert@cs.duke.edu

Vincent Conitzer
Dept. of Computer Science
Duke University
Durham, NC, USA
conitzer@cs.duke.edu

ABSTRACT

In many multiagent environments, a designer has some, but limited control over the game being played. In this paper, we formalize this by considering incompletely specified games, in which some entries of the payoff matrices can be chosen from a specified set. We show that it is NP-hard for the designer to decide whether she can make her choices so that no action in a given set gets played in equilibrium. Hardness holds even in zero-sum games and even in weak tournament games (which are symmetric zero-sum games whose entries are all -1, 0, or 1). The latter case is closely related to the necessary winner problem for a social-choice-theoretic solution concept. We then give a mixed-integer linear programming formulation for weak tournament games and evaluate it experimentally.

1. INTRODUCTION

Game theory provides the natural toolkit for reasoning about systems of multiple self-interested agents. In some cases, the game is exogenously determined and all that is left to do is to figure out how to play it. For example, if we are trying to solve heads-up limit Texas hold'em poker (as was recently effectively done [5]), there is no question about what the game is. Out in the world, however, the rules of the game are generally not set in stone. Often, there is an agent, to whom we will refer as the *designer* or *principal*, that has some control over the game played. Consider, for example, applications of game theory to security domains [19, 21, 1, 23]. In the long run, the game could be changed, by adding or subtracting security resources [4] or reorganizing the targets being defended.

Mechanism design constitutes the extreme case of this, where the designer typically has complete freedom in choosing the game to be played by the agents (but still faces a challenging problem due to the agents' private information). However, out in the world, we generally also do not find this other extreme. Usually, some existing systems are in place and place constraints on what the designer can do. This is true to some extent even in the contexts where mechanism design is most fruitfully applied. For example, one can imagine that it would be difficult and costly for a major search engine to entirely redesign its existing auction mechanism for allocating advertisement space, because of existing users' expectations, interfacing software, etc. But this does not mean that aspects of the game played by the advertisers cannot be tweaked in the designer's favor.

Appears at: 2nd Workshop on Exploring Beyond the Worst Case in Computational Social Choice. Held as part of the 14th International Conference on Autonomous Agents and Multiagent Systems. May 4th, 2015. Istanbul, Turkey.

In this paper, we introduce a general framework for addressing intermediate cases, where the designer has some but not full control over the game. We focus on *incompletely specified games*, where some entries of the game matrix contain *sets* of payoffs, from among which the designer must choose. The designer's aim is to choose so that the resulting equilibrium of the game is desirable to her. This problem is conceptually related to k-implementation [17] and the closely related internal implementation [2], where one of the parties is also able to modify an existing game to achieve better equilibria for herself. However, in those papers the game is modified by committing to payments, whereas we focus on choosing from a fixed set of payoffs in an entry.

We focus on two-player zero-sum games, both symmetric and not (necessarily) symmetric, and show NP-hardness in both cases. (Due to a technical reason explained later, hardness for the symmetric case does not directly imply hardness for the not-necessarily-symmetric case.) The hardness result for the symmetric case holds even for *weak tournament games*, in which the payoffs are all -1, 0, or 1. To show this, we prove hardness of a related problem in *computational social choice*, another important research area in multiagent systems.

In social choice, we take as input a vector of rankings of the alternatives (e.g., $a \succ c \succ b$) and as output return some subset of the alternatives. Some social choice functions are based on the *pairwise majority graph* which has a directed edge from one alternative to another if most voters prefer the former. One attractive concept is that of the *essential set (ES)* [14, 10], which can be thought of as based on the following game. Two abstract players simultaneously pick an alternative, and if one player's chosen alternative has an edge to the other's, the former wins. Then, the essential set consists of all alternatives that are played with positive probability in some equilibrium. Furthermore, an important computational problem in social choice is the *possible (necessary) winner problem* [7, 13, 15, 22, 3]: given only *partial* information about the voters' preferences—for example, because we have yet to elicit the preferences of some of the voters—is a given alternative potentially (necessarily) one of the chosen ones?

We conclude the paper by formulating and evaluating the efficacy of a mixed-integer linear programming formulation for the possible ES winner problem.

2. EXAMPLES

The following is an incompletely specified two-player symmetric zero-sum game with actions a, b, c, d.

	a	b	c	d
a	0	1	0	$\{-1,0,1\}$
b	-1	0	1	0
c	0	-1	0	1
d	$\{-1,0,1\}$	0	-1	0

Here, each entry specifies the payoff to the row player (since the game is zero-sum, the column player's payoff is implicit), and the set notation indicates that the payoff in an entry is not yet fully specified. E.g., $\{-1, 0, 1\}$ indicates that the designer may choose either -1, 0, or 1 for this entry. In the case of symmetric games, we require that the designer keep the game symmetric, so that if she sets[1] $u_1(d, a) = 1$ then she must also set $u_1(a, d) = -1$. Thus, our example game has three possible completions. The goal for the designer, then, is to choose a completion in such a way that the equilibrium of the resulting game is desirable to her. For example, the designer may aim to have only actions a and c played with positive probability in equilibrium. Can she set the payoffs so that this happens? The answer is yes, because the completion with $u_1(a, d) = 1$ has this property. On the other hand, the completion with $u_1(a, d) = -1$ does have Nash equilibria in which b and d are played with positive probability (for example, both players mixing uniformly is a Nash equilibrium of this game).

Next, consider the following incompletely specified asymmetric zero-sum game:

	ℓ	r
t	-2	1
b	$\{-1,1\}$	0

Suppose the designer's goal is to *avoid* row t being played in equilibrium. One might think that the best way to achieve this is to make row b (the only other row) look as good as possible, and thus set $u_1(b, \ell) = 1$. This results in a fully mixed equilibrium where t is played with probability $\frac{1}{4}$ (and ℓ with $\frac{1}{4}$). On the other hand, setting $u_1(b, \ell) = -1$ results in ℓ being a strictly dominant strategy for the column player, and thus the row player would actually play b with probability 1.

3. PRELIMINARIES

In this section, we formally introduce the concepts and computational problems studied in the paper. For a natural number n, let $[n]$ denote the set $\{1, \ldots, n\}$.

3.1 Games

A matrix $M \in \mathbb{Q}^{m\times n}$ defines a two-player zero-sum game (or *matrix game*) as follows. Let the rows of M be indexed by $I = [m]$ and the columns of M be indexed by $J = [n]$, so that $M = (m(i, j))_{i\in I, j\in J}$. Player 1, the *row player*, has action set I and player 2, the *column player*, has action set J. If the row player plays action $i \in I$ and the column player plays action $j \in J$, the payoff to the row player is given by $m(i, j)$ and the payoff to the column player is given by $-m(i, j)$. A *(mixed) strategy* of the row (resp., column) player is a probability distribution over I (resp., J). Payoffs are extended to mixed strategy profiles in the usual way.

A matrix game $M = (m(i, j))_{i\in I, j\in J}$ is *symmetric* if $I = J$ and $m(i, j) = -m(j, i)$ for all $(i, j) \in I \times J$. A *weak tournament game* is a symmetric matrix game in which all payoffs are from the set $\{-1, 0, 1\}$. Weak tournament games naturally correspond to directed graphs $W = (A, \succ)$ as follows: vertices correspond to actions and there is a directed edge from action a to action b (denoted $a \succ b$) if and only if the payoff to the row player in action profile (a, b) is 1.

3.2 Incomplete Games

An *incompletely specified matrix game* (short: *incomplete matrix game*) is given by a matrix $M \in (2^{\mathbb{Q}})^{m\times n}$. That is, every entry of the matrix $M = (m(i, j))_{i\in I, j\in J}$ is a subset $m(i, j) \subseteq \mathbb{Q}$. If $m(i, j)$ consists of a single element, we say that the payoff for action profile (i, j) is *specified*, and write $m(i, j) = m$ instead of the more cumbersome $m(i, j) = \{m\}$. For an incomplete matrix game M, the set of *completions of* M is given by the set of all matrix games that arise from selecting a number from the corresponding set for every action profile for which the payoffs are unspecified.

An *incomplete symmetric game* is an incomplete matrix game with $m(j, i) = \{-m : m \in m(i, j)\}$ for all $i \in I$ and $j \in J$. The set of *symmetric completions* of an incomplete symmetric game M is given by the set of all completions of M that are symmetric. When considering incomplete symmetric games, we will restrict attention to symmetric completions, which is the reason hardness results do not transfer from the symmetric case to the general case. An *incomplete weak tournament game* is an incomplete matrix game for which every symmetric completion is a weak tournament game. Every incomplete weak tournament game corresponds to a directed graph in which the relation for certain pairs (i, j) of distinct vertices have not been determined.

3.3 Equilibrium Concepts

The standard solution concept for matrix games is Nash equilibrium. A strategy profile (σ, τ) is a Nash equilibrium of a matrix game M if the strategies σ and τ are best responses to each other, i.e., $m(\sigma, j) \geq m(\sigma, \tau) \geq m(i, \tau)$ for all $i \in I$ and $j \in J$. The payoff to the row player is identical in all Nash equilibria, and is known as the *value* of the game.

We are interested in the question whether an action is played with positive probability in at least one Nash equilibrium. For improved readability, the following definitions are only formulated for the row player; definitions for the column player are analogous. The support $supp(\sigma)$ of a strategy σ is the set of actions that are played with positive probability in σ. An action $i \in I$ is called *essential* if there exists a Nash equilibrium (σ, t) with $i \in supp(\sigma)$. By $ES_{\text{row}}(M)$ we denote the set of all actions $i \in I$ that are essential.

Definition 1. *The* essential set $ES(M)$ *of a matrix game* M *contains all actions that are essential, i.e.,* $ES(M) = ES_{row}(M) \cup ES_{column}(M)$.

There is a useful connection between the essential set and *quasi-strict* Nash equilibria. Quasi-strictness is a re-

[1]Let $u_1(x, y)$ denote the payoff to the row player in row x and column y.

finement of Nash equilibrium that requires that every best response is played with positive probability [12]. Formally, a Nash equilibrium (σ, t) of a matrix game M is *quasi-strict* if $m(\sigma, j) > m(\sigma, \tau) > m(i, \tau)$ for all $i \in I \setminus supp(\sigma)$ and $j \in J \setminus supp(\tau)$. Since the set of Nash equilibria of a matrix game M is convex, there always exists a Nash equilibrium (σ, τ) with $supp(\sigma) \cup supp(\tau) = ES(M)$. Moreover, it has been shown that all quasi-strict equilibria of a matrix game have the same support [6]. Thus, an action is contained in the essential set of a matrix game if and only if it is played with positive probability in some quasi-strict Nash equilibrium. Brandt and Fischer [6] have shown that quasi-strict equilibria, and thus the essential set, can be computed in polynomial time.

3.4 Computational Problems

We are interested in the question whether the designer can set the unspecified payoffs in such a way that all equilibria of the resulting game use only actions in a prespecified set.

- **Equilibrium Containment Problem (ECP):** Given an incomplete matrix game M, a subset $I' \subseteq I$ of rows, and a subset $J' \subseteq J$ of columns, does there exist a completion M' of M with $ES(M') \subseteq I' \cup J'$?

One may wonder why this is the right problem to solve. One motivation is the following. A general setup would be that the designer has a cost in $[0, \infty)$ for each possible outcome of the game and tries to minimize her expected cost in the worst-case equilibrium for her. The next proposition shows that hardness of the Equilibrium Containment Problem immediately implies hardness, and in fact inapproximability, of the problem of minimizing the designer's cost.

PROPOSITION 1. *Suppose ECP is NP-hard. Then no multiplicative approximation guarantee for the problem of minimizing the designer's expected cost (in the worst-case equilibrium for her) can be given in polynomial time unless P=NP.*

PROOF. Suppose, for the sake of contradiction, that a polynomial-time algorithm giving such a guarantee did exist. Then, given an instance of ECP, we can use this algorithm to determine whether there is a completion where $ES(M') \subseteq I' \cup J'$, as follows. Simply give the designer cost 0 for any outcome $(i, j) \in I' \times J'$, and cost 1 for any other outcome. Then run the algorithm. If there is a completion where $ES(M') \subseteq I' \cup J'$, then the designer has cost 0 in any equilibrium of this completion, and so our algorithm should return a cost of 0 because the approximation is multiplicative. If there is no such completion, then every completion has an equilibrium with positive cost for the designer, and so our algorithm should return a positive cost. Therefore, we would be able to use our approximation algorithm to solve ECP. □

In the context of weak tournament games, where the essential set is interpreted as a social choice function identifying desirable elements from a set of vertices A, a variant of ECP corresponds to the *necessary winner problem*, which asks whether a given vertex a is in the essential set for *all* completions. Clearly, this is the case if and only if there does *not* exist a completion for which the essential set is contained in $A \setminus \{a\}$. This motivates us to study the necessary winner problem for the essential set and its natural counterpart, the possible winner problem.

- **Possible ES Winner:** Given an incomplete weak tournament game W and vertex a, is there a (symmetric) completion W' of W such that $a \in ES(W')$?
- **Necessary ES Winner:** Given an incomplete weak tournament game W and vertex a, is $a \in ES(W')$ for all (symmetric) completions W' of W?

4. ZERO-SUM GAMES

In this section, we analyze the computational complexity of the Equilibrium Containment Problem. In the proof, we will make use of a class of games that we call *alternating games*. Intuitively, an alternating game is a generalized version of Rock-Paper-Scissors that additionally allows "tiebreaking payoffs" which are small payoffs in cases where both players play the same action. Formally, consider a triple (n, d, H), where n is an odd natural number, $d = (d_1, \ldots, d_n) \in \{-1, 0, 1\}^n$, and $H \in \mathbb{Q}_{>1}$. The *alternating game* $C(n, d, H)$ is the matrix game given by $M = (m(i, j))_{i \in I, j \in J}$ with $I = J = [n]$ and

$$m(i,j) = \begin{cases} d_i & \text{, if } i = j \\ (-1)^{j-i-1}H & \text{, if } i < j \\ -m_{ji} & \text{, if } i > j. \end{cases}$$

For example, the alternating game $C(5, (-1, 1, 0, 0, 1), 10)$ has payoff matrix

$$\begin{pmatrix} -1 & 10 & -10 & 10 & -10 \\ -10 & 1 & 10 & -10 & 10 \\ 10 & -10 & 0 & 10 & -10 \\ -10 & 10 & -10 & 0 & 10 \\ 10 & -10 & 10 & -10 & 1 \end{pmatrix}.$$

An alternating game $C(n, d, H)$ is symmetric (and equivalent to a weak tournament game) if and only if $d = (0, \ldots, 0)$. Let $tr(C)$ denote the trace of the payoff matrix of C, i.e., $tr(C) = \sum_{i \in [n]} d_i$. We call C *balanced* if $tr(C) = 0$.

LEMMA 1. *Let $C = C(n, d, H)$ be an alternating game.*

(i) The value of C has the same sign as $tr(C)$ and is monotonically increasing in $tr(C)$.

(ii) C has a unique Nash equilibrium. In this equilibrium, both players play completely mixed strategies.

(iii) As $H \to \infty$, the equilibrium strategies of both players converge to $(\frac{1}{n}, \ldots, \frac{1}{n})$.

In particular, part (i) implies that every balanced alternating game has value zero. We are now ready to prove that the Equilibrium Containment Problem is intractable.

THEOREM 1. *The Equilibrium Containment Problem (in matrix games that are not necessarily symmetric) is NP-complete.*

PROOF. Membership in NP is straightforward, as we can guess a completion M' and compute $ES(M')$.

For hardness, we give a reduction from SETCOVER. An instance of SETCOVER is given by a collection $\{S_1, \ldots, S_n\}$ of subsets of a universe U, and an integer k; the question is whether we can cover U using only k of the subsets. We may assume that k is odd (it is always possible to add a

	c_1	c_2	c_3	c_4	c_5	s_1	s_2	s_3	s_4	s_5	t
$S_{1,1}$	0	H	$-H$	H	$-H$	B	B	0	0	0	N_1
$S_{1,2}$	$\{-1,1\}$	H	$-H$	H	$-H$	-1	-1	0	0	0	N_2
$S_{2,1}$	$-H$	0	H	$-H$	H	0	0	B	B	0	N_1
$S_{2,2}$	$-H$	$\{-1,1\}$	H	$-H$	H	0	0	-1	-1	0	N_2
$S_{3,1}$	H	$-H$	0	H	$-H$	0	0	B	0	B	N_1
$S_{3,2}$	H	$-H$	$\{-1,1\}$	H	$-H$	0	0	-1	0	-1	N_2
$S_{4,1}$	$-H$	H	$-H$	0	H	0	B	0	B	0	N_1
$S_{4,2}$	$-H$	H	$-H$	$\{-1,1\}$	H	0	-1	0	-1	0	N_2
x_1	H	$-H$	H	H	-1	0	0	0	0	0	0

Figure 1: The incomplete matrix game M used in the proof of Theorem 1 for the SetCover instance given by $|U| = 5$, $n = 4$, $k = 3$, $S_1 = \{s_1, s_2\}$, $S_2 = \{s_3, s_4\}$, $S_3 = \{s_3, s_5\}$, and $S_4 = \{s_2, s_4\}$. The double line separates L and R.

singleton subset with an element not covered by anything else and increase k by 1). Define an incomplete matrix game M where the row player has $3n - k$ actions, and the column player has $2n-k+|U|+1$ actions. We denote the row player's actions by $\{S_{i,j} \ : \ i \in [n], j \in [2]\} \cup \{x_i : i \in [n-k]\}$.

Let L denote the restriction of the game to the first $2n-k$ columns. We denote the column player's actions in this part of the game by $\{c_1, \dots, c_{2n-k}\}$. We set $m(S_{i,1}, c_i) = 0$ and $m(S_{i,2}, c_i) = \{-1, 1\}$ for all $i \in [n]$ and $m(x_i, c_{2n+i}) = -1$ for all $i \in [n-k]$. We fill in the remaining entries with H and $-H$, where $H > 0$ is a large constant, so that L acts as an alternating game when exactly one of the actions $S_{i,1}$ and $S_{i,2}$ is removed for each i. At this point the assumption that k is odd is important, since this means that $2n - k$ is also odd and we can create an alternating game.

Let R denote the restriction of the game to the remaining $|U| + 1$ columns. Here, we denote the columns by $\{s_1, \dots, s_{|U|}, t\}$. For column s_j, we set the payoff to be B (where $B > 0$ is a large constant) in row $S_{i,1}$ if $s_j \in S_i$, and 0 otherwise. We set the entry in row $S_{i,2}$ to be -1 if $s_j \in S_i$ and 0 otherwise. Next we find N_1 and N_2 such that $kN_1 + (n-k)N_2 > 0$ and $(k+1)N_1 + (n-k-1)N_2 < -1$ (note that $N_1 < 0$ and $N_2 > 0$). In column t we set the entry in row $S_{i,1}$ to N_1 and the entry in row $S_{i,2}$ to N_2, for each i. We set all other entries of t to be zero. Figure 1 illustrates this construction for a small instance of SetCover.

Observe that in any completion $M' = (m'(\cdot, \cdot))$ of M, if the column player plays only actions from L, then for each pair $(S_{i,1}, S_{i,2})$, the row player will play at most one of them with positive probability in equilibrium, since one of the rows will weakly dominate the other in L and the column player's strategy will have full support in L (see Lemma 1). When $m'(S_{i,2}, c_i) = -1$, $S_{i,1}$ weakly dominates $S_{i,2}$, and the opposite holds when $m'(S_{i,2}, c_i) = 1$. Intuitively, setting the entry $m'(S_{i,2}, c_i)$ to -1 corresponds to choosing S_i for the set cover, and setting $m'(S_{i,2}, c_i)$ to 1 corresponds to not choosing S_i for the set cover. We show that there exists a set cover of size k if and only if the essential set of the corresponding completion is contained within L.

First, suppose that we complete the game in a way that corresponds to a set cover of size k. We will show that there exists an equilibrium that lies completely within L. If the column player plays only columns from L, then we have already observed that M behaves as an alternating game from the perspective of the row player. By Lemma 1, both the row and the column player can guarantee themselves a payoff of zero since the row player has $n - k$ undominated actions with a 1 on the diagonal, and the same number of undominated actions with a -1 on the diagonal. Now let us examine the payoff for the column player from playing any action not in L. For any action s_i there is some chosen set S_j that covers element s_i and for which there is a probability close to $\frac{1}{2n-k}$ (by Lemma 1, as long as H is large enough) that the row player plays $S_{j,1}$, in which case the column player obtains a payoff of $-B$; in all other cases she obtains a payoff of at most 1. Thus, by setting B sufficiently large, the column player will not be incentivized to play any action s_i. Likewise for sufficiently large H the payoff for the column player from playing column t is arbitrarily close to $-(\frac{k}{2n-k}N_1 - \frac{n-k}{2n-k}N_2) < 0$ (by the definition of N_1 and N_2), so she will not play t.

Now suppose that we complete the game in a way that does not correspond to a set cover of size k. There are three cases: we set k payoffs to -1 but they do not correspond to a set cover; we set too many unspecified payoffs to -1; or, we set too few unspecified payoffs to -1. We show that in each case the essential set is *not* contained in L (or—another possibility in the third case—a set cover in fact exists).

Case 1: Exactly k unspecified entries are set to -1, but the corresponding sets do not cover all of U. Suppose for the sake of contradiction that there exists an equilibrium entirely within L. Since there are exactly k sets chosen, Lemma 1 tells us that each player receives exactly zero payoff and that the row player plays exactly the set of undominated actions, as reasoned above. But there is some element, say s_i, that is not covered, so that for every S_j with $s_i \in S_j$ the row player does not put any probability on $S_{j,1}$. Therefore, the column player could deviate to s_i and obtain some strictly positive payoff, since all entries in rows that are actually played by the row player are at most zero (with at least one entry being strictly negative, under the assumption that all elements are contained in at least one set). Therefore there does not exist an equilibrium that stays within L, because the column player would have incentive to deviate.

Case 2: More than k entries, say $\ell > k$ entries, are set to -1. Note that the value of L will be negative, but will be no less than $-\frac{1}{2n-k}$ (the row player could guarantee herself at least $-\frac{1}{2n-k}$ by simply playing every action with probability $\frac{1}{2n-k}$). Again suppose that there exists an equilibrium within L. By the same argument as before, for sufficiently large H the row player plays $S_{i,1}$ actions with total probability arbitrarily close to $\frac{\ell}{2n-k}$ and plays $S_{i,2}$ actions with total probability arbitrarily close to $\frac{n-\ell}{2n-k}$. Therefore the payoff for the column player playing column t is arbitrarily close to $-(\frac{\ell}{2n-k}N_1 + \frac{(n-\ell)}{2n-k}N_2) \geq -(\frac{k+1}{2n-k}N_1 + \frac{(n-k-1)}{2n-k}N_2) > \frac{1}{2n-k}$, by the choice of N_1 and N_2. So the column player can obtain more payoff by playing t than playing only actions in L.

Case 3: Fewer than k unspecified entries are set to -1. There are two possibilities. First, the way of setting the entries corresponds to a set cover. In this case there is necessarily a set cover of size k as well, meaning that M is a yes instance. Second, the payoffs do not correspond to a set cover. Then, by the same argument as for Case 1, the column player can obtain a positive payoff by playing some action s_i. Because the row player can guarantee a value greater than 0 if the game is restricted to L (see Lemma 1), it follows that the equilibrium is not contained within L.

Together, this shows that the equilibrium can be contained in L if and only if there exists a set cover of size k. $\square$

5. WEAK TOURNAMENT GAMES

We now turn to weak tournament games and analyze the computational complexity of possible and necessary ES winners.

THEOREM 2. *The possible ES winner problem (in weak tournament games) is NP-complete.*

PROOF SKETCH. Membership in NP is straightforward as we can guess a completion W' of the incomplete weak tournament game and verify that the action is in $ES(W')$.

For hardness, we provide a reduction from SAT. Let $\varphi = C_1 \wedge \ldots \wedge C_m$ be a Boolean formula in conjunctive normal form over a finite set $V = \{v_1, \ldots, v_n\}$ of variables. We define an incomplete weak tournament $W_\varphi = (A, \succ)$ as follows.[2] The set A of vertices is given by $A = \cup_{i=1}^{n} X_i \cup \{c_1, \ldots, c_m\} \cup \{d\}$, where $X_i = \{x_i^1, \ldots, x_i^6\}$ for all $i \in [n]$. Vertex c_j corresponds to clause C_j and the set X_i corresponds to variable v_i.

Within each set X_i, there is a cycle $x_i^1 \succ x_i^2 \succ x_i^3 \succ x_i^4 \succ x_i^5 \succ x_i^6 \succ x_i^1$ and an unspecified edge between x_i^1 and x_i^4. If variable v_i occurs as a positive literal in clause C_j, we have edges $c_j \succ x_i^3$ and $x_i^5 \succ c_j$. If variable v_i occurs as a negative literal in clause C_j, we have edges $c_j \succ x_i^6$ and $x_i^2 \succ c_j$. Moreover, there is an edge from c_j to d for every $j \in [m]$. For all pairs of vertices for which neither an edge has been defined, nor an unspecified edge declared, we have a tie. See Figure 2 for an example.

[2]We use the notation $a \succ b$ to denote a directed edge from a to b.

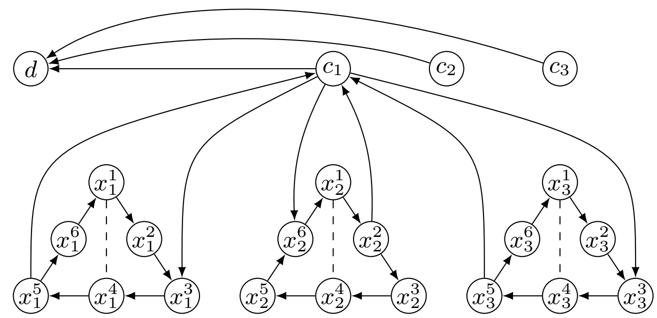

Figure 2: Graphical representation of weak tournament game W_φ for formula $\varphi = C_1 \wedge C_2 \wedge C_3$ with $C_1 = x_1 \vee \neg x_2 \vee x_3$. Dashed lines indicate unspecified edges. For improved readability, edges connecting c_2 and c_3 to X have been omitted.

We make two observations about the weak tournament W_φ. First, for every completion W of W_φ, we have $d \in ES(W)$ if and only if $ES \cap \{c_1, \ldots, c_m\} = \emptyset$. Second, for each i, there is exactly one unspecified edge within (and thus exactly three possible completions of) the subtournament $W_\varphi|_{X_i}$. If the we set a tie between x_i^1 and x_i^4, then all Nash equilibria p of the subtournament $W_\varphi|_{X_i}$ satisfy $p(x_i^1) = p(x_i^3) = p(x_i^5)$ and $p(x_i^2) = p(x_i^4) = p(x_i^6)$. If we set $x_i^1 \succ x_i^4$, then every quasi-strict equilibrium p of $W_\varphi|_{X_i}$ satisfies $p(x_i^2) = p(x_i^4) = p(x_i^6) = 0$, $p(x_i^5) > p(x_i^1) > p(x_i^3) > 0$, and $p(x_i^1)+p(x_i^3) > p(x_i^5)$. By symmetry, setting $x_i^4 \succ x_i^1$ results in quasi-strict equilibria p with $p(x_i^1) = p(x_i^3) = p(x_i^5) = 0$, $p(x_i^4) > p(x_i^6) > p(x_i^2) > 0$, and $p(x_i^2) + p(x_i^6) > p(x_i^4)$.

We can now show hat φ is satisfiable if and only if there is a completion W of W_φ with $d \in ES(W)$. For the direction from left to right, let α be a satisfying assignment and consider the completion W of W_φ as follows: if v_i is set to true, under α add edge $x_i^1 \succ x_i^4$; otherwise, add edge $x_i^4 \succ x_i^1$. It can be shown that $ES(W) = \cup_{i\in[n]} ES(X_i) \cup \{d\}$.

For the direction from right to left, let W be a completion of W_φ with $d \in ES(W)$. Define the assignment α by setting variable v_i to true if $x_i^1 \succ x_i^4$ and to false if $x_i^4 \succ x_i^1$. If there is a tie between x_i^1 and x_i^4, we set the truth value of v_i arbitrarily. Since $d \in ES(W)$, we know by the first observation above that $c_j \notin ES(W)$ for all $j \in [m]$. It can now be shown that every c_i has an incoming edge from a vertex in $ES(W)$, and that this vertex corresponds to a literal that appears in C_i and that is set to true under α. $\square$

We get hardness for the necessary winner problem by slightly modifying the construction used in the proof above.

THEOREM 3. *The necessary ES winner problem (in weak tournament games) is coNP-complete.*

It can actually be shown that the problems considered in Theorems 2 and 3 remain intractable even in the case where unspecified can be chosen from the continuous interval $[-1, 1]$ (while still maintaining symmetry).

As discussed in Section 3.4, the necessary ES winner problem is a special case of the ECP for symmetric matrix games. Therefore, Theorem 3 implies that (the symmetric version of) ECP is intractable even in weak tournament games.

COROLLARY 1. *The Equilibrium Containment Problem is NP-complete in weak tournament games.*

6. MIXED-INTEGER PROGRAM FOR WEAK TOURNAMENT GAMES

Of course, the fact that a problem is NP-hard does not make it go away; it is still desirable to find algorithms that scale reasonably well (or very well on natural instances). NP-hard problems in game theory often allow such algorithms. In particular, formulating the problem as a *mixed-integer program (MIP)* and calling a general-purpose solver often provides good results. Examples include computing optimal Nash equilibria, which is NP-hard (even to approximate) [11, 9] but for which the MIP approach is quite effective [20], and computing Stackelberg strategies in Bayesian games, which again is NP-hard (even to approximate) [8, 16] but for which again the MIP approach is quite effective [18]. In this section, we give a MIP formulation for the possible ES winner problem.

6.1 MIP Formulation

Let $W = (w(i,j))_{i,j\in A}$ be an incomplete weak tournament game. For every entry $w(i,j)$ of W, we define two binary variables x_{ij}^{pos} and x_{ij}^{neg}. Setting $w(i,j)$ to $w_{ij} \in \{-1,0,1\}$ corresponds to setting x_{ij}^{pos} and x_{ij}^{neg} in such a way that $(x_{ij}^{\text{pos}}, x_{ij}^{\text{neg}}) \neq (1,1)$ and $x_{ij}^{\text{pos}} - x_{ij}^{\text{neg}} = w_{ij}$. For each action j, there is a variable p_j corresponding to the probability that the column player assigns to j. Finally, z_{ij} is a variable that, in every feasible solution, equals $w_{ij}p_j$.

To determine whether an action $k \in A$ is a possible ES winner of W, we solve the following MIP. Every feasible solution of this MIP corresponds to a completion of W and a Nash equilibrium of this completion.

$$
\begin{aligned}
\text{maximize}\quad & p_k \\
\text{subject to}\quad & x_{ij}^{\text{neg}} - x_{ji}^{\text{pos}} = 0, \quad \forall i,j \\
& x_{ij}^{\text{pos}} + x_{ij}^{\text{neg}} \leq 1, \quad \forall i,j \\
& x_{ij}^{\text{pos}} = 1, \quad \text{if } w(i,j) = 1 \\
& x_{ij}^{\text{neg}} = 1, \quad \text{if } w(i,j) = -1 \\
& z_{ij} \geq p_j - 2(1 - x_{ij}^{\text{pos}}), \quad \forall i,j \\
& z_{ij} \geq -p_j - 2(1 - x_{ij}^{\text{neg}}), \quad \forall i,j \\
& z_{ij} \geq -2x_{ij}^{\text{pos}} - 2x_{ij}^{\text{neg}}, \quad \forall i,j \\
& \textstyle\sum_{j\in A} z_{ij} \leq 0, \quad \forall i \\
& x_{ij}^{\text{neg}} \in \{0,1\}, \quad \forall i,j \\
& x_{ij}^{\text{pos}} \in \{0,1\}, \quad \forall i,j \\
& \textstyle\sum_{j\in A} p_j = 1 \\
& p_i \geq 0, \quad \forall i
\end{aligned}
$$

Here, indices i and j range over the set A of actions. Most interesting are the constraints on z_{ij}; we note that exactly one of the three will be binding depending on the values of x_{ij}^{pos} and x_{ij}^{neg}. The net effect of these constraints is to ensure that $z_{ij} \geq w_{ij}p_j$. (Since we also have the constraint $\sum_{j\in A} z_{ij} \leq 0$ and because the value of every completion is zero, $z_{ij} = w_{ij}p_j$ in every feasible solution.) All other constraints containing x_{ij}^{pos} or x_{ij}^{neg} are to impose symmetry and consistency on the entries. The remaining constraints make sure that p is a well-defined probability distribution and that no row yields positive payoff for player 1.

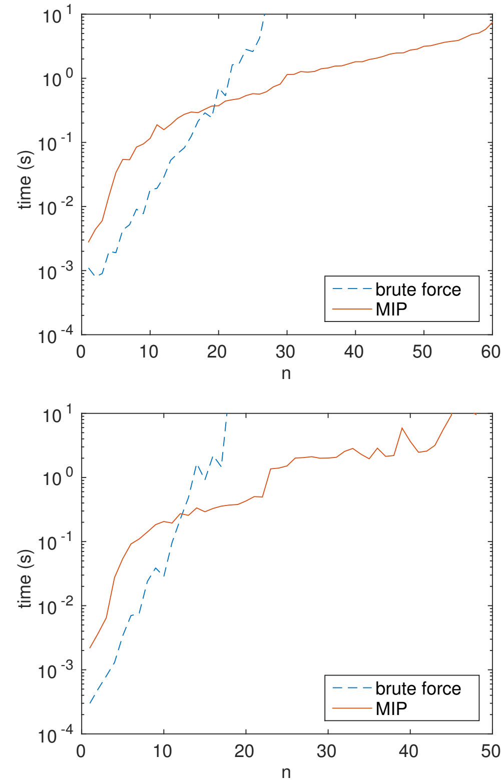

Figure 3: Average runtime (log scale) for $\frac{n}{2}$ unspecified entries (top) and n unspecified entries (bottom).

6.2 Experimental Results

We tested our MIP on weak tournament games containing either $\frac{n}{2}$ or n unspecified entries, where $n = |A|$ is the number of actions available to each player. For each n, we examined the average time required to solve 100 random instances[3] of size n, using CPLEX 12.6 to solve the MIP. Results are shown in Figure 3, with algorithms cut off once the average time to find a solution exceeds 10 seconds.

We compared the performance of our MIP with a simple brute force algorithm. The brute force algorithm performs a depth-first search over the space of all completions, terminating when it finds a certificate of a yes instance or after it has exhausted all completions. We observe that for even relatively small values of n, the MIP begins to significantly outperform the brute-force algorithm. Indeed, it solves random instance of size 60 in around 2 minutes. However, CPLEX on the same machine solves completely specified games of size 1000 in a matter of seconds!

7. CONCLUSION

Often, a designer has some, but limited, control over the game being played, and wants to exert this control to her advantage. In this paper, we studied how computationally hard it is for the designer to decide whether she can choose payoffs in an incompletely specified game to achieve some

[3]Random instances were generated by randomly choosing each entry from $\{-1,0,1\}$ and imposing symmetry, then randomly choosing the fixed number of entries to be unspecified.

goal in equilibrium, and found that this is NP-hard even in quite restricted cases of two-player zero-sum games. Future work may address the following questions. Are there classes of games for which these problems are efficiently solvable? Can we extend the MIP approach to broader classes of games? What results can we obtain for general-sum games? Note that just as hardness for symmetric zero-sum games does not imply hardness for zero-sum games in general (because in the latter the game does not need to be kept symmetric), in fact hardness for zero-sum games does not imply hardness for general-sum games (because in the latter the game does not need to be kept zero-sum). However, this raises the question of which solution concept should be used—Nash equilibrium, correlated equilibrium, Stackelberg mixed strategies, etc. (All of these coincide in two-player zero-sum games.) All in all, we believe that models where a designer has limited, but not full, control over the game are a particularly natural domain of study for AI researchers and computer scientists in general, due to the problems' inherent computational complexity and potential to address real-world settings.

Acknowledgments

This work was supported by NSF and ARO under grants CCF-1101659, IIS-0953756, CCF-1337215, W911NF-12-1-0550, and W911NF-11-1-0332, and by a Feodor Lynen research fellowship of the Alexander von Humboldt Foundation.

8. REFERENCES

[1] B. An, E. Shieh, M. Tambe, R. Yang, C. Baldwin, J. DiRenzo, B. Maule, and G. Meyer. PROTECT - A deployed game theoretic system for strategic security allocation for the United States Coast Guard. *AI Magazine*, 33(4):96–110, 2012.

[2] A. Anderson, Y. Shoham, and A. Altman. Internal implementation. In *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 191–198, Toronto, Canada, 2010.

[3] H. Aziz, M. Brill, F. Fischer, P. Harrenstein, J. Lang, and H. G. Seedig. Possible and necessary winners of partial tournaments. In *Proceedings of the 11th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 585–592. IFAAMAS, 2012.

[4] S. Bhattacharya, V. Conitzer, and K. Munagala. Approximation algorithm for security games with costly resources. In *Proceedings of the Seventh Workshop on Internet and Network Economics (WINE)*, pages 13–24, Singapore, 2011.

[5] M. Bowling, N. Burch, M. Johanson, and O. Tammelin. Heads-up limit hold'em poker is solved. *Science*, 347(6218):145–149, 2015.

[6] F. Brandt and F. Fischer. On the hardness and existence of quasi-strict equilibria. In *Proceedings of the 1st International Symposium on Algorithmic Game Theory (SAGT)*, volume 4997 of *Lecture Notes in Computer Science (LNCS)*, pages 291–302. Springer-Verlag, 2008.

[7] V. Conitzer and T. Sandholm. Vote elicitation: Complexity and strategy-proofness. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 392–397, Edmonton, AB, Canada, 2002.

[8] V. Conitzer and T. Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 82–90, Ann Arbor, MI, USA, 2006.

[9] V. Conitzer and T. Sandholm. New complexity results about Nash equilibria. *Games and Economic Behavior*, 63(2):621–641, 2008.

[10] B. Dutta and J.-F. Laslier. Comparison functions and choice correspondences. *Social Choice and Welfare*, 16(4):513–532, 1999.

[11] I. Gilboa and E. Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1:80–93, 1989.

[12] J. C. Harsanyi. Oddness of the number of equilibrium points: A new proof. *International Journal of Game Theory*, 2:235–250, 1973.

[13] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proceedings of the Multidisciplinary Workshop on Advances in Preference Handling*, pages 124–129. 2005.

[14] G. Laffond, J.-F. Laslier, and M. Le Breton. The bipartisan set of a tournament game. *Games and Economic Behavior*, 5(1):182–201, 1993.

[15] J. Lang, M. S. Pini, F. Rossi, D. Salvagnin, K. B. Venable, and T. Walsh. Winner determination in voting trees with incomplete preferences and weighted votes. *Journal of Autonomous Agents and Multi-Agent Systems*, 25(1):130–157, 2012.

[16] J. Letchford, V. Conitzer, and K. Munagala. Learning and approximating the optimal strategy to commit to. In *Proceedings of the Second Symposium on Algorithmic Game Theory (SAGT-09)*, pages 250–262, Paphos, Cyprus, 2009.

[17] D. Monderer and M. Tennenholtz. K-Implementation. *Journal of Artificial Intelligence Research*, 21:37–62, 2004.

[18] P. Paruchuri, J. P. Pearce, J. Marecki, M. Tambe, F. Ordóñez, and S. Kraus. Playing games for security: An efficient exact algorithm for solving Bayesian Stackelberg games. In *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 895–902, Estoril, Portugal, 2008.

[19] J. Pita, M. Jain, F. Ordóñez, C. Portway, M. Tambe, and C. Western. Using game theory for Los Angeles airport security. *AI Magazine*, 30(1):43–57, 2009.

[20] T. Sandholm, A. Gilpin, and V. Conitzer. Mixed-integer programming methods for finding Nash equilibria. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 495–501, Pittsburgh, PA, USA, 2005.

[21] J. Tsai, S. Rathi, C. Kiekintveld, F. Ordóñez, and M. Tambe. IRIS - A tool for strategic security allocation in transportation networks. In *Proceedings of the Eighth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 37–44, Budapest, Hungary, 2009.

[22] L. Xia and V. Conitzer. Determining possible and necessary winners under common voting rules given

partial orders. *Journal of Artificial Intelligence Research*, 41:25–67, 2011.

[23] Z. Yin, A. X. Jiang, M. Tambe, C. Kiekintveld, K. Leyton-Brown, T. Sandholm, and J. P. Sullivan. TRUSTS: Scheduling randomized patrols for fare inspection in transit systems using game theory. *AI Magazine*, 33(4):59–72, 2012.