Computing Optimal Strategies to Commit to in Extensive-Form Games

Joshua Letchford
Duke University
Dept. of Computer Science
Durham, NC, USA
jcl@cs.duke.edu

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

ABSTRACT

Computing optimal strategies to commit to in general normal-form or Bayesian games is a topic that has recently been gaining attention, in part due to the application of such algorithms in various security and law enforcement scenarios. In this paper, we extend this line of work to the more general case of commitment in extensive-form games. We show that in some cases, the optimal strategy can be computed in polynomial time; in others, computing it is NP-hard.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems; J.4 [**Social and Behavioral Sciences**]: Economics; K.4.4 [**Computers and Society**]: Electronic Commerce

General Terms

Algorithms, Economics, Theory

Keywords

noncooperative game theory, commitment, Stackelberg, extensive-form games

1. INTRODUCTION

Game theory defines solution concepts for strategic situations, in which multiple self-interested agents interact in the same environment. Perhaps the best-known solution concept is that of *Nash equilibrium* [13]. A Nash equilibrium prescribes a strategy for every player, in such a way that no individual player has an incentive to change her strategy. If strategies are allowed to be mixed—a mixed strategy is a probability distribution over pure strategies—then it is known that every finite game has at least one Nash equilibrium. Some games have more than one equilibrium, leading to the *equilibrium selection problem*.

Perhaps the most basic representation of a game is the *normal form*. In the normal-form representation, every player's pure strategies are explicitly listed, and for every combination of pure strategies, every player's utility is explicitly listed.

The problem of *computing* Nash equilibria of a normal-form game has received a large amount of attention in recent years. Finding a Nash equilibrium is PPAD-complete [5, 1]. Finding an optimal equilibrium (for just about any reasonable definition of "optimal"—for instance, maximizing the sum of the players' utilities) is NP-hard [6, 3]; moreover, it is not even possible to find an equilibrium that is approximately optimal in polynomial time, unless P=NP [3]. This holds even for two-player games.

However, one can argue about whether Nash equilibrium is always the right solution concept. In some settings, one player can credibly commit to a strategy, and communicate this to the other player, before the other player can make a decision. To see how this can affect the outcome of a game, consider the following game (which is often used as an example of this).

EXAMPLE 1 (KNOWN). *Consider the normal-form game in Figure 1. For the case where the players move simultaneously (no ability to commit), the unique Nash equilibrium is (U, L): U strictly dominates D, so that the game is solvable by iterated strict dominance. So, player 1 (the row player) receives utility* 1. *However, now suppose that player 1 has the ability to commit. Then, she is better off committing to play D, which will incentivize player 2 to play R, resulting in a utility of* 2 *for player 1. The situation gets even better for player 1 if she can commit to a mixed strategy: in this case, she can commit to the mixed strategy* $(.5 - \epsilon, .5 + \epsilon)$, *which still incentivizes player 2 to play R, but now player 1 receives an expected utility of* $2.5 - \epsilon$. *To ensure the existence of optimal strategies, we assume (as is commonly done and for which good justifications can be given [2, 14]) that player 2 breaks ties in player 1's favor, so that the optimal strategy for player 1 to commit to is* $(.5, .5)$, *resulting in a utility of* 2.5. *(Note that there is never a reason for player 2 to randomize, since he effectively faces a single-agent decision problem.)*

In fact, commitment to a mixed strategy can never hurt a player (relative to Nash equilibrium in the game without commitment), because at worst the player can just commit to her Nash equilibrium strategy. This argument can be extended to show that it cannot even hurt relative to correlated equilibrium [18]. In contrast, committing to a pure strategy is not always beneficial; for example, consider matching pennies.

One may argue that the normal form is not the correct representation for this game and other games with commitment. In game theory, the time structure of games is usually represented in the *extensive form*. Indeed, the above game (with commitment to mixed strategies) can be represented as the extensive-form game in Figure 1. While this is a conceptually useful representation, from a computational perspective it is not helpful: player 1 has an infinite number of strategies, hence (the naïve representation of) the

EC'10, June 7–11, 2010, Cambridge, Massachusetts, USA.

	L	R
U	(1,1)	(3,0)
D	(0,0)	(2,1)

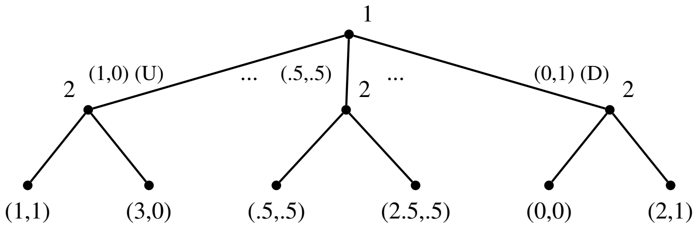


Figure 1: A normal-form game, and the extensive-form representation of its commitment version.

tree has infinite size. Instead, from a computational perspective, it is better to operate directly on the normal form (or some other concise representation of the game).

An optimal strategy to commit to is usually called a *Stackelberg* strategy, after von Stackelberg, who showed that in Cournot's duopoly model [4], a firm that can commit to a production quantity has a strategic advantage [17]. The computation of Stackelberg strategies in general games has recently started to receive significant attention. The first paper on this topic [2] appeared in EC-06 and studied Stackelberg strategies in normal-form and Bayesian games, and showed, among other things, that the optimal mixed strategy to commit to in a two-player normal-form game can be found in polynomial time using linear programming,[1] though this becomes NP-hard in Bayesian games or with three players (a later paper [12] proved inapproximability results for the Bayesian case). Unfazed by the NP-hardness result, Paruchuri et al.[14] developed a mixed-integer programming formulation for the Bayesian case, and this algorithm has been implemented at the core of deployed security applications, specifically the strategic random placement of checkpoints and canine units at Los Angeles International Airport [7, 15]. Later work [8, 10] studied computing Stackelberg strategies in settings where the normal form has exponential size, for example, when player 1 has to allocate multiple resources to defend multiple targets. One motivating domain for this line of work is a new application: the scheduling of Federal Air Marshals [16].

Real-world applications to security and law enforcement are not the only motivation for work on computing optimal strategies to commit to; the notion of commitment plays a key role in many game-theoretic settings. Notably, in mechanism design (or environment design or principal-agent settings), the designer/center/autioneer/principal is assumed to be able to commit to a mechanism before the (other) agents move. For example, if an auctioneer uses a Vickrey auction, it is generally assumed that she will unambiguously commit to this mechanism, rather than (for example) waiting for the bids to come in, and subsequently backtracking on her promise of a Vickrey auction and attempting to charge the winner her own bid (the first price) after all. Consequently, research on computational aspects of commitment in game theory has the potential to serve as a bridge between work by computer scientists on game theory and work by computer scientists on mechanism design.

Our contributions. All of the above work considers settings where in the "original" game, the players move simultaneously—and then we add the ability to commit for one player (which conceptually results in an extensive-form game, though we have argued that that representation is generally not helpful computationally). However, the original game may already have a time structure of its own. In this case, the original game is best represented as an extensive-form game. Perhaps unsurprisingly, commitment can also be helpful in such games, as the following example shows.

EXAMPLE 2. *Consider the game in Figure 2. (Note that player 1 moves* second *in this game; this is because in this paper, we insist that player 1 be the player with the commitment advantage.)*

Without any commitment advantage, this game can be solved by backward induction: player 1 will move Left if player 2 moves Left, and Right if player 2 moves Right; player 2 prefers the former, so the resulting utilities are $(1, 3)$.

Now, suppose that player 1 can commit to a pure strategy. Then, she can commit to the strategy: Right if player 2 moves Left; Left if player 2 moves Right. In this case, player 2 will prefer to move Right, resulting in utilities $(2, 2)$*—a better outcome for player 1.*

Furthermore, suppose that player 1 can commit to a mixed behavioral strategy.[2] *Then, she can commit to the strategy: right if player 2 moves left, 50% left and 50% right if player 2 moves right. This leaves player 2 indifferent between going left and right; by the tiebreaking assumption, he will move right, resulting in expected utilities of* $(2.5, 1)$.

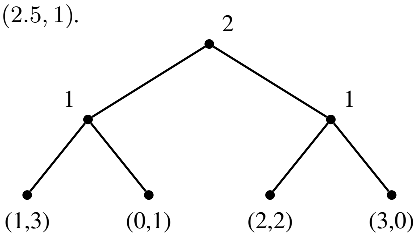


Figure 2: Extensive-form game for example 2.

The focus of this paper is to determine for which extensive-form games the Stackelberg solution can be found efficiently.

Of course, once again, we can incorporate the commitment stage into the game form, resulting in an even larger extensive-form game. This approach scales even more poorly for extensive-form games than for normal-form games; not only is there still the problem of an infinite-size tree when commitment to mixed strategies is allowed, but now even commitment to pure strategies does not scale, because if there are n information sets for player 1 (with, say, 2 actions each), then there are 2^n possible pure strategies to which player 1 can commit. That is, the size of the extensive-form representation that incorporates the commitment stage is exponential in the size of the original extensive-form representation.

Alternatively, we can convert the original extensive-form game into normal form, and then apply the standard techniques for computing Stackelberg strategies there. This approach runs into the difficulty that the size of the normal form will be exponential.[3]

[1] This algorithm was presented in [2], and then again in [18].

[2] We recall that a behavioral strategy gives a probability distribution over actions at each information set. In extensive-form games of perfect recall, any mixed strategy is equivalent to a behavioral strategy [11].

[3] It may be interesting to investigate whether sequence-form

The rest of this paper is laid out as follows. In Section 2, we lay out the different variants of the problem. Section 3 contains our positive results—those variants of extensive-form games where it is possible to solve for a Stackelberg strategy in polynomial time. Section 4 contains our negative results—those variants of extensive-form games where solving for a Stackelberg strategy is NP-hard. Section 5 concludes and gives a summary of our results in tree form.

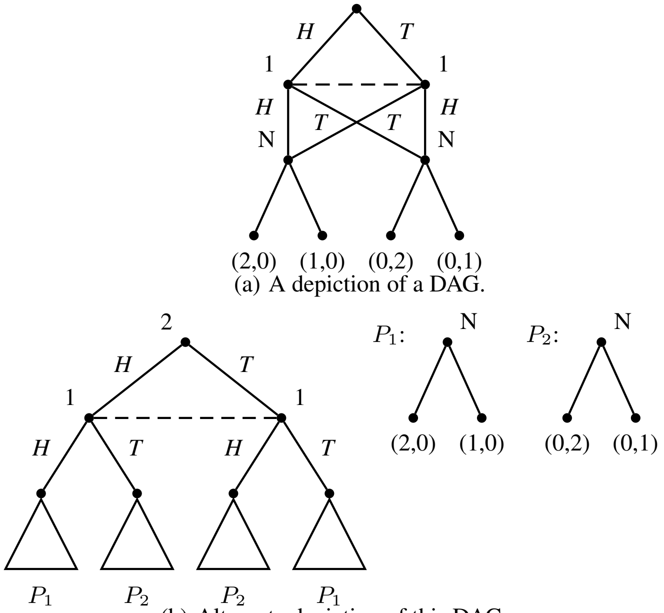
2. THE VARIANTS OF THE PROBLEM

We will refer to player 1 as the *leader* and the other player(s) as the *follower(s)*. An extensive-form game consists of a rooted tree with additional information. Each leaf node of the tree specifies the payoffs for all players. Each internal node of the tree is associated with a player, who acts at that node, resulting in a move to one of that node's children. A player's nodes are partitioned into *information sets*, and the player cannot distinguish among nodes in an information set (this implies that the player has the same set of possible actions at every node in the same information set). Let m be the number of leaf nodes in the tree, and n the number of internal nodes.

We will consider commitment to *behavioral strategies*: a behavioral strategy for a player associates with each information set for that player a probability distribution over the actions associated with that information set. A behavioral strategy is *pure* if all of its probabilities are 0 or 1; in general, a behavioral strategy is *mixed.*

We examine how the following aspects affect the difficulty of solving for a Stackelberg (behavioral) strategy in an extensive-form game.

- *Chance nodes (moves by Nature).* Does the game include moves by Nature? (Nature is a player that plays according to a fixed behavioral strategy and has no stake in the game; when we count the number of players, Nature is not included.)
- *Perfect or imperfect information.* A game has perfect information if all information sets have size 1.
- *Commitment to pure strategies vs. mixed strategies.* Is the leader able to commit to a mixed behavioral strategy, or only to a pure one?
- *Tree or DAG.* Conceptually, in extensive-form games, there is no loss of generality in assuming that the game is represented as a tree: if two different paths of play both lead us to the same state of the game, it is always possible to simply have two copies of that state, corresponding to the different paths that may have led us there. However, this duplication can result in an exponential blowup in representation size. Moreover, when we consider commitment to behavioral strategies, it may not be reasonable to allow a player to commit to one action at one node, and another at a different node that in reality represents the same state. For this reason, we consider not only the standard representation of an extensive-form game using a tree (and additional information), but also using a directed acyclic graph (DAG). When a game is represented using a DAG, the probabilities to which a player commits at a node cannot depend on the path taken to that node. Figure 3 shows an example a modified form of matching pennies where the winner has a chance of winning an additional unit of utility. This game is depicted as a DAG in two different equivalent ways (in the second way(b), when two nodes are both shown as parents of a subtree, for example P_1, this means that they are both parents of a single copy of that subtree). The second way will be how DAGs will be drawn for the rest of this paper.
- *Number of players.* Some of the variants will turn out to become hard at two players, some at three players, and others are easy for any number of players.
- *Restricted or costly commitment.* It may be the case that the leader is able to commit only at some nodes. Alternatively, it may be the case that for each node, there is a non-negative cost associated with committing at that node, which will be subtracted from the leader's utility at the end of the game. (Restricted commitment is the special case where all costs are 0 or ∞; hence, a positive result for costs implies a positive result for restrictions, and vice versa for negative results.)



(a) A depiction of a DAG.

(b) Alternate depiction of this DAG.

Figure 3: Example of DAG depiction methods.

One issue that deserves some discussion is the following: does the leader commit at every information set, or can the leader selectively choose to commit at some, but not other, information sets? In many cases, it will be without loss of generality to assume that the leader commits at every information set (assuming that there are no restrictions or costs to doing so). To argue this, we first need to consider what happens when the leader does not commit everywhere. After the commitment phase, an extensive-form game results, where every node at which the leader committed is replaced by a move by Nature, in accordance with the mixture to which the leader committed there. We assume that some Nash equilibrium of this modified game will result; since we are only interested in games of perfect recall, we can assume that the equilibrium is in behavioral strategies [11]. Moreover, we assume that if the extensive-form game is one of perfect information, then a *pure* Nash equilibrium will be played.

With these assumptions, if commitment to mixed behavioral strategies is possible, then it does not hurt the leader to commit to the

techniques [9], which are used to solve extensive-form games efficiently (without blowing them up to normal form), can be extended to be useful in settings with a commitment advantage. In this paper, however, we do not investigate this approach: we are primarily interested in whether versions of our problem can be solved in polynomial time or are NP-hard, and we manage to settle nearly all of these questions.

probabilities prescribed by the Nash equilibrium (of the modified game) that is best for her; assuming ties are broken in the leader's favor, she will do no worse. Even if commitment to mixed behavioral strategies is not allowed, then still, if the extensive-form game is one of perfect information, since we assume that a pure Nash equilibrium will be played, the leader can simply commit to the actions prescribed by the best such equilibrium for her.

Nevertheless, in games of imperfect information where commitment to mixed strategies is not allowed, we must admit that it can be optimal to not commit everywhere (for example, matching pennies; it is also easy to construct examples where the leader wants to commit at some, but not all, information sets). Our hardness result here assumes that the leader can choose to commit at a subset of the information sets. Throughout, if there are multiple equilibria in the game that results after the commitment stage, we assume that one of the equilibria that is optimal for player 1 will be played (with the caveat that if the game has perfect information, we assume that the best *pure* equilibrium for player 1 will be played). This is consistent with the common assumption that the follower breaks ties in the leader's favor.

A summary of our results is given in Figure 8 in the conclusion.

3. POSITIVE RESULTS

In this section we give polynomial-time algorithms for the following problems:

Case 1. Perfect-information games in tree form with no chance nodes and no costs/restrictions on commitment, where we allow for any number of players, but restrict the problem to pure strategy commitment (Theorem 1).

Case 2. Perfect-information games in tree form with no chance nodes and costs/restrictions on commitment, where we allow for any number of players, but restrict the problem to pure strategy commitment (Theorem 2).

Case 3. Perfect-information games in DAG form with no chance nodes and no costs/restrictions on commitment, where we restrict the problem to two players and pure strategy commitment (Theorem 3).

Case 4. Perfect-information games in tree form with no chance nodes and no costs/restrictions on commitment, where we restrict the problem to two players but allow for mixed strategy commitment (Theorem 4).

3.1 No chance, pure strategies, trees, no costs or restrictions, perfect information

THEOREM 1. *In perfect-information games in tree form with no chance nodes and no costs/restrictions on commitment, the optimal pure strategy to commit to can be found in $O(nm)$ time (for any number of players).*

PROOF. The algorithm contains two main steps. The first step will be a bottom-up dynamic program for determining, for each node v, a set S_v, which is the subset of v's descendant leaf nodes that can be achieved by an appropriate commitment by player 1. We then choose the best outcome $l^* \in S_r$ for player 1, where r is the root of the tree. Then, in the second (top-down) stage, we determine the appropriate strategy to commit to to achieve l^*.

Dynamic programming step (upward pass)

For each leaf node l, $S_l = \{l\}$. For every internal node v at which player 1 takes an action, 1 can achieve any leaf that can be obtained from any of its children, by committing to the corresponding child and committing appropriately from there on. Hence, $S_v = \bigcup_{w \text{ child of } v} S_w$.

The case of an internal node v at which player $i \neq 1$ takes an action is slightly more complicated. Here, it will be true that $S_v \subseteq \bigcup_{w \text{ child of } v} S_w$, but in general not all elements of this union will be included, because it may not feasible to incentivize player i to choose every one of them. For each child w of v, let $m(w) \in \arg\min_{l \in S_w} u_i(l)$; this leaf corresponds to the most we can "punish" i for going to w. Then, $S_v = \bigcup_{w \text{ child of } v} \{l \in S_w : \max_{w' \neq w} u_i(m(w')) \leq u_i(l)\}$.

These recursive equalities allow us to use dynamic programming to compute S_r; we can then choose $l^* \in \arg\max_{l \in S_r} u_1(l)$.

Stackelberg strategy determination (downward pass)

We need a procedure $\mathsf{strategy}(v, l)$ that, given a node v and one of its descendant leaves $l \in S_v$, specifies how to commit in the subtree rooted at v so that the outcome l results. If player 1 acts at v, then let w be the child of v that is an ancestor of l; simply commit to going to w and recursively call $\mathsf{strategy}(w, l)$. (It does not matter how 1 commits at other children of v since they will not be reached.) If player $i \neq 1$ acts at v, then let w be the child of v that is an ancestor of l. For w, recursively call $\mathsf{strategy}(w, l)$. For all other children $w' \neq w$ of v, recursively call $\mathsf{strategy}(w', m(w'))$.

Runtime analysis

We recall that there are m leaves and n internal nodes. Each time we reach an internal node, we require a constant number of operations on a subset of the leaves; each of these operations requires at most linear time; we reach every internal node twice (once on the upward and once on the downward pass). □

3.2 No chance, pure strategies, trees, perfect information

In this subsection, we extend Theorem 1 to the case where commitment at an information set can come at a cost (or where there are restrictions on commitment at some information sets, which is the special case where the cost is ∞). This comes at the cost of an additional factor m in the runtime.

THEOREM 2. *In perfect-information games in tree form with no chance nodes, the optimal pure strategy to commit to can be found in $O(nm^2)$ time (with any number of players, and with costs or restrictions on commitment).*

The main difference between the algorithm for this case from the one used in the proof of Theorem 1 is that, in the upward pass, we now compute the optimal cost to achieve each outcome for each subtree. Furthermore, we now need to take into account that, for a subtree in which player 1 moves first, the cheapest way to obtain a given outcome may not be to commit at the root of that subtree, but rather to use commitments further down to make the desired move at the root optimal. Due to space constraints, the detailed proof for Theorem 2 appears only in the full version of the paper.

3.3 No chance, pure strategies, two players, no costs or restrictions, perfect information

So far, we have focused on commitment to pure strategies in perfect-information games of no chance, represented in tree form. The following result extends this to DAG games, though here we must restrict to the two-player case without costs and restrictions (our later hardness results show that we cannot hope for more, unless P=NP).

THEOREM 3. *In two-player perfect-information games in DAG form with no chance nodes and no costs/restrictions on commitment, the optimal pure strategy to commit to can be found in time $O(mn(m + n))$.*

The algorithm for this case revolves around determining whether a given outcome is feasible to achieve, by minimizing player 2's utility at nodes that do not lead to the desired outcome. Due to space constraints, the detailed proof for Theorem 3 appears only in the full version of the paper.

3.4 No chance, trees, two-player, no costs or restrictions, perfect information

So far, we have focused on commitment to pure strategies in perfect-information games of no chance. We now give a positive result for commitment to mixed behavioral strategies, although this requires that there be only two players and that the game be represented in tree form. (Again, our later hardness results show that we cannot hope for more, unless P=NP.) Our next result also assumes that there are no costs or restrictions.

THEOREM 4. *In two-player perfect-information games in tree form with no chance nodes and no costs/restrictions on commitment, the optimal mixed strategy to commit to can be found in time $O(nm^2)$.*

PROOF. The algorithm contains two main steps. First, we will determine through a bottom-up dynamic program, for each node v, a set S_v, which is the set of all mixtures over v's descendant leaf nodes that can be achieved by an appropriate commitment by player 1. To aid in visualizing S_v, we picture each leaf node l as a point p in two-dimensional space, where the x-dimension is player 2's utility and the y-dimension is player 1's utility. Since the goal is a commitment strategy that maximizes utility for player 1, we need only maintain the upper envelope of S_v, which we can represent as a set of line segments, each of which can be stored compactly as pairs of two points (p, p'). Let S_v^1 be the set of line segments that we maintain. (In fact, for computational simplicity, we will maintain more than just the line segments corresponding to the upper envelope, but the upper envelope of the line segments in S_v^1 will correspond to the upper envelope of S_v.) Finally, for computational reasons, we maintain a set S_v^2, which is always a subset of the endpoints of the line segments in S_v^1 (but excludes some that we will not need).

Dynamic programming step (upward pass)

For each leaf node l, $S_l^2 = \{l\}$ and $S_l^1 = \{(l, l)\}$. At a node v where player 1 acts, for any child w of v, any point in S_w is also in S_v. Moreover, if we take one point from S_w for every child w, every mixture over these points is also in S_v. However, because there are only two players, it will never be helpful to mix over points from more than two children (for any point obtained that way, there will be a point in S_v that is obtained by mixing over points from only two children that has the same utility for player 2 and at least the same utility for player 1). This allows us to compute the set of endpoints: $S_v^2 = \bigcup_{w \text{ child of } v} S_w^2$. The line segments in S_v^1 will come from two different sources. First, some line segments will come directly from S_w^1 for some child w, via a pure commitment to w. The second source is by generating segments by mixing between any two children of v, w and $w' \neq w$; in this case, we can restrict attention to mixing over one end point from S_w^2 and one from $S_{w'}^2$ (points obtained by mixing over other points will be dominated). We can define the set resulting from this second source as $\hat{S}_v^1 = \bigcup_{p \in S_w^2, p' \in S_{w'}^2 \text{ s.t. } w < w'} \{(p, p')\}$ (we define an arbitrary order $<$ over the child nodes to avoid duplication). Thus, we get that $S_v^1 = (\bigcup_{w \text{ child of } v} S_w^1) \cup \hat{S}_v^1$.

For a node v at which player 2 acts, it can happen that, for at least one child w of v, there are values in S_w that it is impossible to incentivize player 2 to play. Let $m_{w'} \in \arg\min_{p \in S_{w'}^2} u_2(p)$; this point corresponds to the most we can "punish" player 2 for going to w'. Let $i(w) = \max_{w' \neq w} u_2(m_{w'})$; this value corresponds to the smallest utility that player 2 will accept when going to w.

Based on $i(w)$, we will shrink S_w^1 into a set $\hat{S}_w^1$, so that $S_v^1 = \bigcup_{w \text{ child of } v} \hat{S}_w^1$. For every $(p, p') \in S_w^1$, we do the following:

- If $u_2(p) \geq i(w)$ and $u_2(p') \geq i(w)$, add (p, p') to $\hat{S}_w^1$.
- If $u_2(p) < i(w)$ and $u_2(p') < i(w)$, do not add (p, p') to $\hat{S}_w^1$.
- If $u_2(p) < i(w)$ and $u_2(p') \geq i(w)$, find the point p'' where $u_2(p'') = i(w)$ and $p'' = (\alpha p + (1 - \alpha)p')$ for some $\alpha \in [0, 1]$. Then, add (p'', p') to $\hat{S}_w^1$ and add p'' to a set of new potential endpoints $\hat{S}_w^2$. The case where $u_2(p) \geq i(w)$ and $u_2(p') < i(w)$ is similar.

To calculate S_v^2, we include all points p that are in S_w^2 for some child w and for which $u_2(p) \geq i(w)$. Additionally, for each child w, if $\hat{S}_w^2$ is nonempty, we add one point p to S_v^2, namely the point in $\arg\max_{p \in \hat{S}_w^2} u_1(p)$.

Finally, at the root node r, we calculate the solution $p^* \in \arg\max_{p \in S_r^2} u_1(p)$.

Commitment determination (downward pass)

We need a procedure $\mathsf{strategy}(v, p'')$ that, given a node v and a point p'' that lies on one of the lines $(p, p') \in S_v^1$, specifies how to commit in the subtree rooted at v so that the outcome p results. If player 1 acts at v, we calculate α and (p, p') where $p'' = \alpha p + (1 - \alpha)p'$, $\alpha \in [0, 1]$. Then, we find w and w' where $p \in S_w^2$ and $p' \in S_{w'}^2$. If $w = w'$, simply commit to going to w and recursively call $\mathsf{strategy}(w, p'')$. If $w \neq w'$, then at v commit to the mixture αw and $(1 - \alpha)w'$, and recursively call $\mathsf{strategy}(w, p)$ and $\mathsf{strategy}(w', p')$. (It does not matter how 1 commits at other children of v since they will not be reached.) For a node v where player 2 acts, find the descendant w that contains p'' (more precisely, find w such that $(p, p') \in \hat{S}_w^1$ where $p'' = \alpha p + (1 - \alpha)p'$ for some $\alpha \in [0, 1]$). Then, recursively call $\mathsf{strategy}(w, p'')$, and, for all other children $w' \neq w$, call $\mathsf{strategy}(w', m_w)$.

Runtime analysis

If we consider all the S_v^2 on a single level of the tree, we can bound the sum of the sizes of these sets by m, because there are only two types of points in these sets. First, any leaf node can appear in them (but can appear only once at each level of the tree). Second, at any player 2 node at or below this level, a new point can be created, but if so, another point from below this node will be removed.

Next, let us consider the potential sum of the sizes of all the S_v^1 on a single level. New line segments can only be generated at internal nodes v from some $p \in S_w^2$ and some $p' \in S_{w'}^2$ where w and $w' \neq w$ are children of v, and thus at the same level. If the only end points ever generated were leaf nodes (which is not true in general), then there can be at most $\binom{m}{2}$ line segments at a level (since there will be no duplicates). More generally, if a new end point is created at a player 2 node, it replaces another end point; therefore, every end point corresponds to a leaf that it replaced (or that was replaced by the end point it replaced, etc.). Therefore the $\binom{m}{2}$ bound on the number of line segments at a level holds generally.

At a node v where player 1 acts, taking an existing line segment from one of its children or generating a new one from two of its children is a constant-time operation per line segment. Hence, this step takes $O(m^2)$ time per node. At a node v where player 2 acts, it also takes $O(m^2)$ time to evaluate all the line segments from all of its children to see which have to be altered and which can be included without change. Thus, the upward pass requires $O(nm^2)$

time. The downward pass merely has to find, at each node v, which line segment(s) intersects with the desired point, which can be done in $O(m^2)$ time, as there are at most $O(m^2)$ line segments in its children's feasible sets. Again, because we are doing this for at most $O(n)$ nodes, this requires at most $O(nm^2)$ time in total. □

4. NEGATIVE RESULTS

In this section we give NP-hardness results for the following problems:

Case 1. It is NP-hard to solve for the optimal strategy to commit to in a game with chance nodes, even if the game has only two players, it is in tree form, it is a game of perfect information with no costs/restrictions, and regardless of whether commitment to mixed strategies is allowed (Theorem 5).

Case 2. It is NP-hard to solve for the optimal mixed strategy to commit to in a three-player game, even if the game is in tree form and it is a game of perfect information with no costs/restrictions or chance nodes (Theorem 6).

Case 3. It is NP-hard to solve for the optimal strategy to commit to in a game with imperfect information, even if the game has only two players, it is in tree form, it has no costs/restrictions or chance nodes, and regardless of whether commitment to mixed strategies is allowed (Theorem 7).

Case 4. It is NP-hard to solve for the optimal mixed strategy to commit to in a game in DAG form, even if the game has only two players and it is a game of perfect information with no costs/restrictions or chance nodes (Theorem 8).

Case 5. It is NP-hard to solve for the optimal strategy to commit to in a three-player game in DAG form, even if the game is a game perfect information with no costs/restrictions or chance nodes, regardless of whether commitment to mixed strategies is allowed (Theorem 9).

Case 6. It is NP-hard to solve for the optimal strategy to commit to in a game in DAG form with commitment restrictions, even if the game has only two players, it is a game of perfect information with no chance nodes and regardless of whether commitment to mixed strategies is allowed (Theorem 10).

4.1 Chance nodes

Our first hardness result shows that when chance nodes are involved, even the simplest of problems become NP-hard.

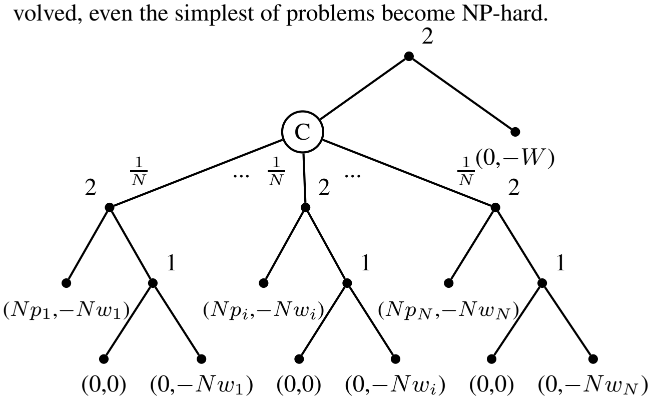


Figure 4: Two-player game with chance nodes used in the hardness reduction of Theorem 5.

THEOREM 5. *It is NP-hard to solve for the optimal strategy to commit to in a game with chance nodes, even if the game has only two players, it is in tree form, it is a game of perfect information with no costs/restrictions, and regardless of whether commitment to mixed strategies is allowed.*

PROOF. In the KNAPSACK problem, we are given a set of N items, and for each of them, a value p_i and a weight w_i; additionally, we are given a weight limit W. We are asked to find a subset of the items with total weight at most W that maximizes the sum of the p_i in the subset. We reduce an arbitrary KNAPSACK instance to an extensive-form game, in such a way that the maximal utility obtainable by the leader with commitment (whether pure or mixed) is equal to the optimal solution value in the KNAPSACK instance. The extensive-form game is illustrated in Figure 4, and defined formally below.

Top of the tree

The first move is by player 2, who chooses between an outcome of $(0, -W)$, and a chance node which randomizes uniformly over the item subtrees, defined next.

Item subtrees

Each item subtree t^i has two levels. At the top level, there is a node where player 2 acts. It has two children: one is a leaf node with an outcome of $(Np_i, -Nw_i)$, the other is a node where player 1 acts. The latter node also has two children: leaf nodes $(0, 0)$ and $(0, -Nw_i)$.

Proof of equivalence to KNAPSACK instance

If, at the lowest internal node of an item subtree, player 1 commits to playing 100% Right, then player 2, breaking ties in 1's favor, will move Left in this subtree, resulting in payoffs $(Np_i, -Nw_i)$ if this subtree is reached. Otherwise, player 2 will move Right, and player 1 will get 0 (and player 2 at most 0). Because player 1 wants player 2 to move to the chance node at the top, there is no benefit to player 1 in moving Right with probability strictly between 0 and 100%, since this will only make the chance node less desirable to player 2 without benefiting player 1, so we can assume without loss of optimality that player 1 commits to a pure strategy.

Let S be the set of indices of subtrees where player 1 commits to playing Right. Then, player 2's expected utility for the chance node is $(1/N)\sum_{i \in S} -Nw_i = -\sum_{i \in S} w_i$. Player 2 will choose to move to the chance node if and only if $\sum_{i \in S} w_i \leq W$. Given this, player 1's expected utility is $(1/N)\sum_{i \in S} Np_i = \sum_{i \in S} p_i$. Hence, finding player 1's optimal strategy to commit to is equivalent to solving the KNAPSACK instance. □

4.2 Chance node gadget

Before we get to the next reduction, let us introduce a gadget that we will use in two of the following reductions. This gadget allows us to create a pseudo-chance node in games of no chance, when commitment to mixed behavioral strategies is allowed. The construction of a pseudo-chance node with N equally likely outcomes requires $2(N - 1)$ internal nodes and is as pictured in Figure 5. The outcomes of the pseudo-chance node correspond to the leaves immediately following an action by player 1.

To illustrate how the construction works, let us solve for the optimal mixed behavioral strategy for player 1 to commit to.

PROPOSITION 1. *The optimal mixed behavioral strategy for player 1 to commit to in the game in Figure 5 is the following: At the* i*th node for player 1 from the bottom, commit to the mixture* $(1/(i+1), i/(i+1))$. *Given this, player 2 will always move right, and each leaf node directly connected to a player 1 node is reached with probability* $1/n$. *Moreover, if the utilities for player 1 are perturbed by a sufficiently small amount, the optimal mixed behavioral strategy for player 1 remains exactly the same.*

PROOF. In an optimal solution, player 1 will incentivize player 2 to move Right everywhere, because player 1 gets 0 when player 2 moves Left. Let us consider player 1's optimal commitment at the bottom internal node of the tree. Player 1 needs to commit to

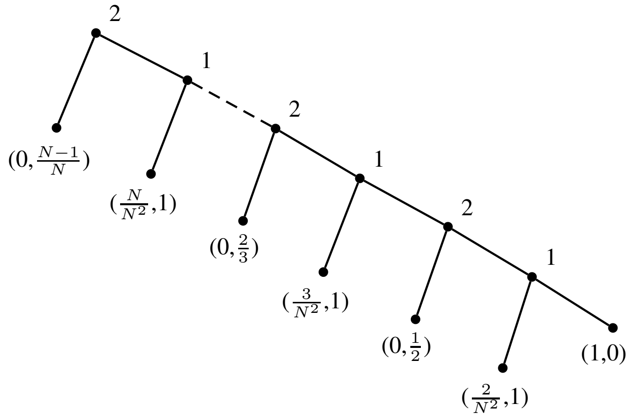


Figure 5: Pseudo-chance node gadget used in Theorem 6 and Theorem 8.

placing at least probability $1/2$ on Left in order for player 2 to want to move Right at the node before. An additional benefit for player 1 of putting probability on Left here is that it helps incentivize player 2 to move Right at higher nodes. However, player 1 will not want to place more than probability $1/2$ on Left, because incentives for player 2 to move right higher up in the tree are "cheaper" at higher-up player 1 nodes (because player 1 gets higher utility for moving Left higher up in the tree, while player 2's utility for this is always 1). Given this, at the next higher node for player 1, by similar reasoning, it is optimal for her to commit to going left with probability $1/3$, etc. Given the optimal strategy, the probability that player 1 ends up moving Left at her ith node from the bottom (and not before) is $\frac{1}{i+1}\prod_{j=i+1}^{N-1} j/(j+1) = 1/N$; and the probability of player 1 never moving Left is $\prod_{j=1}^{N-1} j/(j+1) = 1/N$. $\square$

Thus, this construction allows us to generate a uniform distribution. This may at first appear not nearly as useful as being able to use a true chance node, because we can only get a uniform distribution over specially chosen leaves, rather than over any collection of subtrees. Nevertheless, in the proofs below, we will be able to modify some of these leaves into more interesting subtrees without affecting the strategic result of a uniform distribution.

4.3 Three-player, mixed strategy commitment

We use the pseudo-chance node gadget to prove that optimal commitment to a mixed behavioral strategy is NP-hard in three-player games.

THEOREM 6. *It is NP-hard to solve for the optimal mixed strategy to commit to in a three-player game, even if the game is in tree form and it is a game of perfect information with no costs/restrictions or chance nodes.*

PROOF. In the KNAPSACK problem, we are given a set of N items, and for each of them, an integer value p_i and an integer weight w_i; additionally, we are given an integer weight limit W. (Without loss of generality, $w_i \leq W$ for all i.) We are asked to find a subset of the items with total weight at most W that maximizes the sum of the p_i in the subset. Let $p^* = \max_i p_i$. We reduce an arbitrary KNAPSACK instance to a three-player extensive-form game, in such a way that the maximum utility obtainable by the leader with mixed strategy commitment corresponds to the maximum value obtainable in the KNAPSACK problem. The extensive-form game is illustrated in Figure 6, and defined formally below.

Top of the tree

The first move is by player 3, who can choose either an outcome of $(0, 0, -W)$, or to move to the pseudo-chance node. The pseudo-chance node will (in the optimal mixed commitment solution) randomize uniformly over N item subtrees and an additional outcome of $(1, 0, 0)$. (In the representation of the tree in Figure 6, the children of the pseudo-chance node PC correspond to all the leaves of the tree in Figure 5 where player 1 moved last. The leaves where player 2 moved last still need to be part of the game, but we do not draw them in Figure 6 since they will not be reached in the optimal solution—they are "hidden" as part of the pseudo-chance node construction. This is the reason that player 2 does not occur at all in Figure 6—he is "hidden in the pseudo-chance node." Also note that the use of the pseudo-chance node places restrictions on the utilities below it: player 2's utilities must be identical to those in Figure 5 (that is, they must all equal 1), and player 1's utilities must be very close to those in Figure 5. Note that the N in Figure 5 equals $N + 1$ in Figure 6.)

Item subtrees

We define a subtree t^{x_i} for each item. This subtree consists of two levels. At the top node of the subtree, player 3 chooses between two children: one is a leaf node with an outcome of $(\frac{i+1}{(N+1)^2} + \frac{p_i}{2p^* W(N+1)^2}, 1, -(N + 1)w_i)$, the other is a node where player 1 acts. This latter node also has two children: leaf nodes $(\frac{i+1}{(N+1)^2}, 1, 0)$ and $(\frac{i+1}{(N+1)^2}, 1, -(N + 1)w_i)$.

Proof of equivalence to KNAPSACK instance

If at the lowest internal node of an item subtree, player 1 commits to playing 100% Right, then player 3 will move Left in this subtree, resulting in payoffs $(\frac{i+1}{(N+1)^2} + \frac{p_i}{2p^* W(N+1)^2}, 1, -N + 1w_i)$. Otherwise, player 3 will move Right, and player 1 will get $\frac{i+1}{(N+1)^2}$ (and player 3 at most 0). Because player 1 wants player 3 to move to the pseudo-chance node at the top, there is no benefit to player 1 in moving Right with probability strictly between 0 and 100%, since this will only make the pseudo-chance node less desirable to player 3, without benefiting player 1, so we can assume without loss of optimality that player 1 commits to a pure strategy in each item subtree.

Let us assume, for now, that in the optimal solution, the distribution over the pseudo-chance node's children is uniform. Let S be the set of indices of subtrees where player 1 commits to playing Right. Then, player 3's expected utility for the chance node is $(1/(N + 1)) \sum_{i \in S} -(N + 1)w_i = -\sum_{i \in S} w_i$. Player 3 will choose to move to the chance node if and only if $\sum_{i \in S} w_i \leq W$. Given this, player 1's expected additional utility (relative to the case where $S = \emptyset$) is $(1/(N + 1)) \sum_{i \in S} \frac{p_i}{2p^* W(N+1)^2} = \frac{1}{(2p^* W(N+1)^3)} \sum_{i \in S} p_i$. This quantity is maximized when $\sum_{i \in S} p_i$ is maximized (nothing else depends on S), so finding player 1's optimal mixed strategy to commit to is equivalent to solving the KNAPSACK instance—under the assumption that the distribution over the pseudo-chance node's children is uniform.

Now, let us revisit that assumption—can player 1 obtain even higher utility by committing inside the pseudo-chance node in a way that results in the pseudo-chance node not mixing uniformly?

First, suppose that the choice of player 1's commitment in the item subtrees is fixed to a feasible solution S, that is, $\sum_{i \in S} w_i \leq W$. Then, by the analysis in the proof of Proposition 1, player 1 is best off committing in the pseudochance node in a way that results in a uniform distribution, because player 1's utilities are only slightly perturbed relative to Figure 5.

The more difficult case is when player 1's commitment in the item subtrees is fixed to an *in*feasible solution S, with $\sum_{i \in S} w_i >$

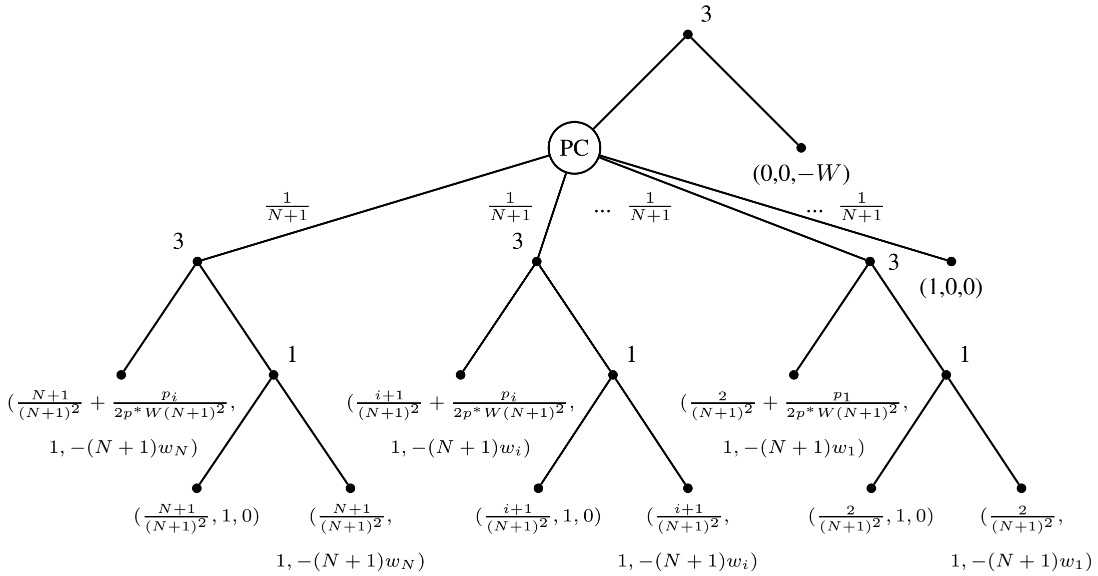


Figure 6: Three-player tree with pseudo-chance node used in the hardness reduction of Theorem 6.

W. While this is clearly not beneficial for player 1 if the pseudochance node results in a uniform distribution (because player 3 will move Right at the beginning), it may be possible for player 1 to commit to a different strategy in the pseudo-chance node so that player 3 will in fact move Left at the beginning of the game. We will show that player 1 cannot benefit from such an approach, in the sense that in this case player 1 would be better off reducing S to the point where it is feasible, and committing in a way such that the pseudo-chance node results in a uniform distribution. Roughly, the idea is the following. Suppose that $\sum_{i \in S} w_i > W$. By integrality, we know that $\sum_{i \in S} w_i \geq W + 1$, that is, the sum of the included items' weights is bounded away from W. Hence, in order to make player 3 play Left at the root, player 1's strategy at the pseudo-chance node must be significantly different, so that the resulting distribution over the children of the pseudo-chance node must be signficantly different from the uniform distribution. We then show that the cost of this difference to player 1 is larger than what she could ever obtain from including more items.

Specifically, in any distribution over the pseudo-chance node's children that makes player 3 play Left when $\sum_{i \in S} w_i > W$, at least $1/W$ of the probability mass must have shifted relative to the uniform distribution over the pseudo-chance node's children. This is because player 3's expected utility for going Left must be increased by at least 1 relative to the uniform distribution, but for each unit of mass that is shifted, the expected utility of player 3 can change by at most $\max_i w_i \leq W$.

Moreover, player 1's commitment in the pseudo-chance node must still incentivize player 2 to always move Right, because player 1 gets 0 after player 2 moves Left. It is not difficult to show that in order for this to be true, it must be the case that the first i children of the pseudo-chance node receive probability at most i/N. Because of this, if $1/W$ mass is shifted in the distribution over the pseudochance node's children, it must be the case that the expected index of the child chosen from the pseudo-chance node increases by $1/W$. As a result, the expectation of the term $(i+1)/(N+1)^2$ in player 1's payoff decreases by $1/(W(N+1)^2)$. (Note that $N+2-i$ is the index of the child.) On the other hand, the expectation of the term $x_i \cdot p_i/(2p^* W(N+1)^2)$ in player 1's expected utility—where $x_i \in \{0, 1\}$ indicates whether item i has been included in S, so that this term corresponds to his utility from including items—can increase by at most $\max_i p_i/(2p^* W(N+1)^2) = 1/(2W(N+1)^2)$, which is less than $1/(W(N+1)^2)$. Hence it follows that it is never helpful to player 1 to change the distribution from the pseudochance node in order to include items beyond the limit W. It follows that the problem of finding the optimal mixed strategy to commit to in the game in Figure 6 is equivalent to the KNAPSACK instance. □

4.4 Two players, imperfect information, pure or mixed strategy commitment

Next, we will show that the addition of imperfect information is enough to make the problem NP-hard even with two players.

THEOREM 7. *It is NP-hard to solve for the optimal strategy to commit to in a game with imperfect information, even if the game has only two players, it is in tree form, it has no costs/restrictions or chance nodes, and regardless of whether commitment to mixed strategies is allowed. This holds even if only player 1 has nonsingleton information sets.*

PROOF. We will first consider the case of mixed-strategy commitment. We reduce an arbitrary instance of 3SAT to an extensiveform game such that the leader can obtain utility 1 if the 3SAT instance is satisfiable and 0 otherwise. The 3SAT instance consists of N variables x_1,...,x_N and M clauses, C_1,...,C_M. This game is illustrated in Figure 7 and the details of its construction follow.

The top of the tree and the information sets

Connected to the root (at which player 2 acts), for each variable x_i, the game contains a subtree t^{x_i}; for each clause C_i, a subtree t^{C_j}; and finally, a "target" subtree t. Player 1's objective is to incentivize player 2 to choose t, as this is the only subtree where she can get positive utility. The roots of all of these subtrees belong to player 1, and they are all in the same information set S. Thus, they have the same possible actions for player 1, specifically, one action for each variable x_i. Furthermore, we have one information set S_i for each variable x_i; the commitment decision here will correspond to the choice of a literal ($+x_i$ or $-x_i$).

Target subtree

The root of the t subtree has a child for each variable x_i; each of these corresponds to a simple coordination game, where 1 and 2 both receive 1 if and only if they choose the same action. There is

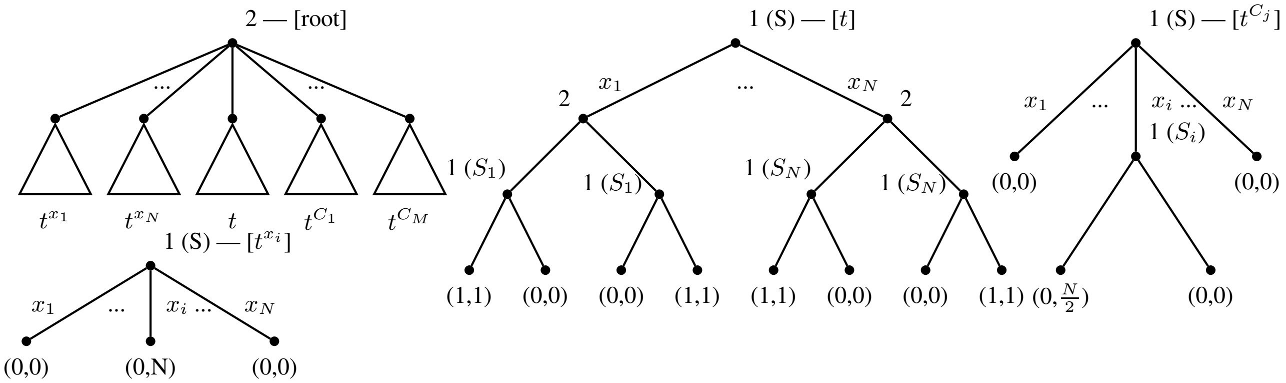


Figure 7: Two-player game of imperfect information used in the hardness reduction of Theorem 7. The notation (X) next to a node indicates that the node is part of the information set X. In t^{C_j}, $+x_i$ appears in C_j.

one twist: player 1's information set S_i in the coordination game for x_i also includes some nodes from the clause subtrees.

Clause subtrees

The root of each t^{C_j} subtree also has a child for each variable x_i. If x_i does not appear in C_j at all, then this child is a leaf node with utilities $(0, 0)$. If $+x_i$ or $-x_i$ appears in C_j, then the child is part of the information set S_i, and player 1 must move Left or Right. Moving Right corresponds to setting the variable to *true*, and results in utilities $(0, \frac{N}{2})$ if $-x_i$ appears in C_j and utilities $(0, 0)$ if $+x_i$ appears in C_j. Moving Left corresponds to setting the variable to *false*, and results in utilities $(0, 0)$ if $-x_i$ appears in C_j and utilities $(0, \frac{N}{2})$ if $+x_i$ appears in C_j.

Variable subtrees

For each variable subtree t^{x_i}, the action corresponding to x_i leads to a leaf node with a payoff of $(0, N)$, and all other actions lead to a leaf node with a payoff of $(0, 0)$.

Proof of equivalence to 3SAT instance (with mixed-strategy commitment)

First, in order for player 2 to choose t, player 1 must commit to a mixed behavioral strategy that places equal ($\frac{1}{N}$) probability on each of the choices in the information set S. Otherwise, some x_i receives more than $\frac{1}{N}$ probability, and as a result player 2 would receive strictly more than 1 expected utility from t^{x_i}, whereas t can give player 2 at most 1. Given this, player 1 also must commit to a pure strategy in each of the information sets S_i. This is because otherwise, player 2 would get expected utility strictly less than 1 for t (because with some positive probability there would be a coordination failure), causing him again to prefer some t^{x_i}.

Thus, the only decision remaining for player 1 is whether to commit to Left or Right at each S_i (that is, whether to choose $-x_1$ or $+x_i$, respectively). We now argue that player 2 will choose t rather than one of the clause subtrees, if and only if player 1's choices correspond to a satisfying assignment. If, in information set S_i, player 1 has committed to going Right ($+x_i$), then all clause subtrees t^{C_j} with $+x_i \in C_j$ will give player 2 a utility of at most 1. This is because in this case there will be at most two subtrees of the clause subtree where player 2 receives $N/2$, resulting in an expected utility of at most 1 for him for the clause subtree. The case where player 1 has committed to going Left ($-x_i$) has an identical effect on the clause subtrees t^{C_j} with $-x_i \in C_j$. However, if all three literals of a clause C_j are set to *false*, then player 2 will prefer this unsatisfied clause tree t^{C_j} to t, because his expected utility for t^{C_j} is $\frac{3}{2} > 1$.

Thus, the only way in which player 1 can obtain a utility of 1 is by committing to a uniform distribution in information set S, and a pure action at every information set S_i, where these actions correspond to a satisfying assignment for the 3SAT instance.

Commitment to pure actions only

We now argue that player 1 can still obtain utility 1 with a satisfying assignment even if she can only commit to pure actions (but is free to not commit at some of the information sets). Commitment at information sets (S_1, ..., S_N) is already restricted to pure actions in order for player 1 to obtain utility 1. However, player 1 is now unable to commit to a uniform distribution at information set S, and we have already seen that commitment to anything other than a uniform distribution here results in utility 0 for player 1. However, if player 1 chooses to *not* commit in information set S, it turns out that player 1 playing a uniform distribution in information set S and player 2 playing choosing t is part of a Nash equilibrium, if and only player 1 plays in a way that satisfies all clauses in the lower information sets. (In fact, player 1 does not need to commit in these information sets either, in the sense that there is an equilibrium in which she will play the satisfying assignment—of course, there are many other equilibria as well.) Thus, even in the pure-strategy case (where commitment is not required), player 1 can obtain a utility of 1 if and only if there is a satisfying assignment to the 3SAT problem. □

4.5 DAG results

Due to space constraints, the proofs from this subsection appear only in the full version of the paper. We again use the pseudo-chance node gadget from Subsection 4.2 to prove that optimal commitment to a mixed behavioral strategy is NP-hard in two-player games on a DAG.

THEOREM 8. *It is NP-hard to solve for the optimal mixed strategy to commit to in a game in DAG form, even if the game has only two players and it is a game of perfect information with no costs/restrictions or chance nodes.*

If we have more than two players, solving for the optimal strategy to commit to in DAGs becomes NP-hard even when we only allow for pure strategy commitment.

THEOREM 9. *It is NP-hard to solve for the optimal strategy to commit to in a three-player game in DAG form, even if the game is a game perfect information with no costs/restrictions or chance nodes, regardless of whether commitment to mixed strategies is allowed.*

Finally, if we allow for restrictions, solving for the optimal commitment strategy in DAGs becomes NP-hard even in the two-player, pure-strategy commitment case.

THEOREM 10. *It is NP-hard to solve for the optimal strategy to commit to in a game in DAG form with commitment restrictions, even if the game has only two players, it is a game of perfect information with no chance nodes and regardless of whether commitment to mixed strategies is allowed.*

5. CONCLUSIONS

A summary of our results appears in Figure 8, in the form of a decision tree. (This is the only tree in this paper that does not correspond to a game.)

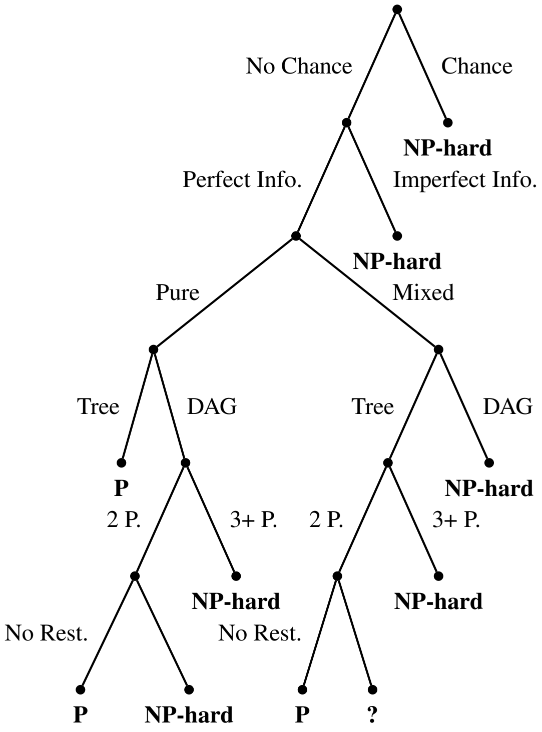


Figure 8: Summary of results. At any node, some of the aspects of the problem are fixed, others are not. Every leaf node states whether the problem is in P or NP-hard at that leaf; this result then applies for *any* way of fixing the remaining aspects of the problem.

We note that we have left one case open. Besides this, future research can investigate questions such as the following. Are there good approximation algorithms for the NP-hard cases? Are there other restrictions on the games that allow for fast algorithms? Can the runtime of the algorithms in this paper be improved, perhaps through better use of data structures?

Acknowledgments

This work has been supported by NSF CAREER 0953756 and IIS-0812113, by ARO 56698-CI, and an Alfred P. Sloan fellowship.

6. REFERENCES

[1] Xi Chen and Xiaotie Deng. Settling the complexity of two-player Nash equilibrium. In *Proceedings of FOCS*, pages 261–272, 2006.

[2] Vincent Conitzer and Tuomas Sandholm. Computing the optimal strategy to commit to. In *Proceedings of EC*, pages 82–90, Ann Arbor, MI, USA, 2006.

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

[4] Antoine Augustin Cournot. *Recherches sur les principes mathématiques de la théorie des richesses (Researches into the Mathematical Principles of the Theory of Wealth)*. Hachette, Paris, 1838.

[5] Constantinos Daskalakis, Paul Goldberg, and Christos H. Papadimitriou. The complexity of computing a Nash equilibrium. In *Proceedings of STOC*, pages 71–78, 2006.

[6] Itzhak Gilboa and Eitan Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1:80–93, 1989.

[7] Manish Jain, James Pita, Milind Tambe, Fernando Ordóñez, Praveen Paruchuri, and Sarit Kraus. Bayesian Stackelberg games and their application for security at Los Angeles International Airport. *SIGecom Exch.*, 7(2):1–3, 2008.

[8] Christopher Kiekintveld, Manish Jain, Jason Tsai, James Pita, Fernando Ordóñez, and Milind Tambe. Computing optimal randomized resource allocations for massive security games. In *Proceedings of AAMAS*, pages 689–696, Budapest, Hungary, 2009.

[9] Daphne Koller and Nimrod Megiddo. Finding mixed strategies with small supports in extensive form games. *International Journal of Game Theory*, 25:73–92, 1996.

[10] Dmytro Korzhyk, Vincent Conitzer, and Ronald Parr. Complexity of computing optimal stackelberg strategies in security resource allocation games. In *Proceedings of AAAI*, Atlanta, GA, USA, 2010.

[11] H. W. Kuhn. Extensive games and the problem of information. In H. W. Kuhn and A. W. Tucker, editors, *Contributions to the Theory of Games*, volume 2 of *Annals of Mathematics Studies, 28*, pages 193–216. Princeton University Press, 1953.

[12] Joshua Letchford, Vincent Conitzer, and Kamesh Munagala. Learning and approximating the optimal strategy to commit to. In *Proceedings of SAGT*, pages 250–262, Paphos, Cyprus, 2009.

[13] John Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36:48–49, 1950.

[14] Praveen Paruchuri, Jonathan P. Pearce, Janusz Marecki, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. Playing games for security: an efficient exact algorithm for solving Bayesian Stackelberg games. In *Proceedings of AAMAS*, pages 895–902, Estoril, Portugal, 2008.

[15] James Pita, Manish Jain, Fernando Ordóñez, Christopher Portway, Milind Tambe, and Craig Western. Using game theory for Los Angeles airport security. *AI Magazine*, 30(1):43–57, 2009.

[16] Jason Tsai, Shyamsunder Rathi, Christopher Kiekintveld, Fernando Ordonez, and Milind Tambe. IRIS - a tool for strategic security allocation in transportation networks. In *AAMAS - Industry Track*, 2009.

[17] Heinrich von Stackelberg. *Marktform und Gleichgewicht*. Springer, Vienna, 1934.

[18] Bernhard von Stengel and Shmuel Zamir. Leadership games with convex strategy sets. *Games and Economic Behavior*, 2009.