On the NP-Completeness of Finding an Optimal Strategy in Games with Common Payoffs

Francis Chu* Joseph Halpern*

Department of Computer Science
Upson Hall, Cornell University
Ithaca, NY 14853-7501, USA

{fcc,halpern}@cs.cornell.edu

August 21, 2002

Abstract

Consider a very simple class of (finite) games: after an initial move by nature, each player makes one move. Moreover, the players have common interests: at each node, all the players get the same payoff. We show that the problem of determining whether there exists a joint strategy where each player has an expected payoff of at least r is NP-complete as a function of the number of nodes in the extensive-form representation of the game.

1 Introduction

In many problems arising in distributed computing, we are free to program the agents so as to achieve a goal of the designer; thus, the agents can be viewed as pursuing a common goal. The interaction becomes a game in which the strategic aspect is of little importance, whereas the coordination aspect becomes crucial.

To give just one example, in recent years, the Federal Aviation Administration (FAA) has introduced the concept of *Free Flight*, which will decentralize the National Airspace System (NAS). Free Flight allows pilots, whenever practical, to choose their own route, instead of following pre-assigned routes. (See `http://www.faa.gov/freeflight/ff_ov.htm`

*Work supported in part by NSF under grant IRI-96-25901.

for more details.) Since a pilot may plot his own course instead of taking a pre-assigned route under Free Flight, the pilot might well want to optimize the route for his payoff function. Of course, the optimal choices will depend both on what other pilots do and on what can be viewed as Nature's moves (e.g., the wind speed).

While pilots are no longer obligated to follow routes pre-assigned by the FAA during Free Flight, they might not want to act completely independently, since there might be incentives to cooperate. For example, pilots from the same organization (e.g., airline, shipping company, etc.) might well want to coordinate their flights so as to optimize the payoff function of the organization itself, since what is optimal for a particular flight might not be the best thing to do for the organization as a whole. Indeed, in the extreme case, the pilots may have the same payoff function (the payoff function of the organization to which they belong).

This type of of situation, where the agents have a common payoff because they are members of the same team or organization (or because their payoff can be taken to be the organization's payoff) arises frequently in distributed systems applications. In games of this sort, it turns out that there is an optimal joint strategy—one that gives all the agents at least as high a payoff (individually) as any other joint strategy. This optimal joint strategy is deterministic and is Pareto optimal, a Nash equilibrium, and a correlated equilibrium (under perhaps the most natural definition of correlated equilibrium in extensive-form games).

An obvious goal is thus to compute the optimal joint strategy. Here, unfortunately, is where the scalability concern bites. We show that the problem of computing the optimal joint strategy is NP-complete as a function of the number of nodes in the extensive-form representation of the game. (More precisely, we show that it is NP-complete to determine whether there is a joint strategy that nets the players a payoff of at least r for a fixed rational number r.) This is true even if there are only two players in the game, each of whom moves once, after an initial move by nature. The role of nature is critical. Without the initial move by nature, it is easy to see that the problem is decidable in polynomial time, no matter how many players there are. It is also easy to see that two players are required for the lower bound; if there is just a single player, it is again easy to figure out the optimal move for that player.

These results also depend on the fact that we are considering the extensive-form representation of the game. Using the normal-form representation of the game, it is trivial to find an optimal joint strategy for the players by looking at the payoff matrix: it is just a joint strategy that nets them the best payoff. (Recall that we are considering games where the players get the same payoff.) There is no contradiction with the NP-completeness result here: the normal-form representation can be exponential in the size of the extensive-form representation.

These results form an interesting contrast to those of Gilboa and Zemel [1989], who considered related questions in the context of games in normal form. They showed that, given a game G and a number r, computing whether there exists a Nash equilibrium where each player gets a payoff of at least r is NP-complete, while computing whether there exists a correlated equilibrium where each player gets at least r is decidable in polynomial time. Note that Gilboa and Zemel are trying to determine whether there exists a Nash or correlated equilibrium where the players do well, whereas we are trying to determine whether there exists a joint strategy where the players do well. However, in our setting, there exists a joint strategy where the players do well iff there is a Nash/correlated equilibrium where the players do well.

It is also interesting to compare these results to those of Koller and Megiddo [1992]. Like us, they consider extensive form and focus on the two-player case but they consider zero-sum games, where the players have diametrically opposite interests, whereas we consider coordination games where the players have identical interests. They show that it is NP-hard to decide whether player 1 can *guarantee* an expected payoff of at least r (no matter what player 2 does), even if there are no chance moves. They show the problem is NP-complete if there are no chance moves or if player 1 has perfect recall, but need to allow player 2 to have imperfect recall. Note that since we restrict players to making one move each, they certainly have perfect recall in our setting.

The rest of the paper is organized as follows. In Section 2 we briefly review the relevant definitions and explain the difficulty of computing an optimal solution. In Section 3, we prove the main NP-completeness result of the paper. In Section 4 we offer some concluding remarks.

2 Preliminaries

We first describe the games of interest to us in somewhat nonstandard notation (which nevertheless seems to us fairly natural and appropriate for our application). We then describe the easy conversions to normal and extensive form. For us, an n-player game, G_n, is a $2n+3$-tuple $(W, A_1, \ldots, A_n, O_1, \ldots, O_n, \Pr, \textsc{obs}, \textsc{pay})$ such that

- W is the set of possible worlds,
- A_i is the set of possible actions of agent i,
- O_i is the set of possible observations of agent i,
- $\Pr$ is the probability distribution on W,
- $\textsc{obs}(w, i) \in O_i$ is the observation agent i makes in world w, and
- $\textsc{pay}(w, (a_1, \ldots, a_n)) \in \mathbf{R}^n$ is the joint payoff of the joint action $(a_1, \ldots, a_n)$ in world w.

Note that each world $w \in W$ determines the observation each agent makes (via $\textsc{obs}$) and the joint payoff given any joint action (via $\textsc{pay}$). We assume that the game (i.e., the tuple) is common knowledge among the agents. Intuitively, if $\text{PAY}(w, (a_1, \dots, a_n)) = (b_1, \dots, b_n)$, then b_i is the payoff to agent i if the world is w and agent j performs action a_j, $j = 1, \dots, n$. A *game with common payoffs* is one where all joint payoffs have the form $(b, \dots, b)$: i.e., each agent gets the same payoff.

Each agent i decides what action to take based on his observation and his strategy. Formally, a *strategy* for agent i is a function from O_i to A_i. That is, for each observation, the strategy prescribes an action. (Note that this is a *deterministic* strategy. In general, *randomized* strategies are also possible. However, as we show shortly, in the games with common payoffs that we are interested in, there is no loss of generality in considering only on deterministic strategies.) A *joint strategy* is an n-tuple $(S_1, \dots, S_n)$ such that S_i is the strategy for agent i. (For brevity, we will sometimes write $(x_1, \dots, x_k)$ as $\vec{x}$.) Given a joint strategy $\vec{S}$, we can compute its expected joint payoff

$$\mathbf{E}^{\text{PAY}}(\vec{S}) = \sum_{w \in W} \Pr(w) \cdot \text{PAY}(w, (S_1(\text{OBS}(w, 1)), \dots, S_n(\text{OBS}(w, n)))).$$

Clearly such a game can be easily converted to extensive form. Nature makes the first move, by choosing a world. Then agent 1 moves (by choosing an action in A_1), agent 2 moves (by choosing an action in A_2), and so on. Agent i cannot distinguish two nodes n and n' in the game tree if nature's move in the path to n is w, nature's move in the path to n' is w', and $\text{OBS}(w, i) = \text{OBS}(w', i)$. That is, agent i's information sets are determined by the $\text{OBS}(\cdot, i)$. Note that the size of the game tree in the extensive-form representation is essentially the same as the size of the description of the PAY function. Thus, the conversion from the representation that we have chosen to the extensive-form representation is polynomial.

It is also easy to convert a game to normal form, by describing the matrix of expected payoffs for each joint strategy. However, in general, this conversion is exponential. For example, it is easy to construct a two-player game where $|W| = n^2$, $|O_1| = |O_2| = n$, and $|A_1| = |A_2| = 2$. The description of this game is quadratic in n, and the game tree has $4n^2$ nodes. However, since a strategy for player i is a function O_i to A_i, there are 2^n strategies for each player, and the normal-form representation of the game is of size exponential in n.

As is well known, finite games always have Pareto optimal joint strategies and always have strategies that are in Nash equilibrium (if we allow randomized strategies). However, in general, the set of Pareto optimal strategies may be disjoint from the strategies in Nash equilibrium. This is not the case with common payoffs. Indeed, with common payoffs, $\preceq$ is a linear (pre)ordering, so there is a joint strategy with the highest expected (joint) payoff. (There may be more than one, since there could be ties.) Thus, there is always a deterministic joint strategy that dominates every joint strategy (with respect to the $\preceq$ ordering); such a joint strategy is *a fortiori* Pareto optimal and a Nash equilibrium, since no agent can do (strictly) better with any other joint strategy. Moreover, since the payoff of a randomized strategy is a convex combination of the payoffs of deterministic strategies, there can be no randomized strategy with a higher payoff. Thus, in games with common payoffs, a (deterministic) joint strategy with highest expected payoff is both a Nash equilibrium and Pareto optimal.

We are interested in characterizing the difficulty of computing the optimal strategy. To make this a decision problem, we consider the problem of deciding whether the optimal strategy gives a payoff of at least r. As we said in the introduction, we show that this problem is NP-complete for our representation (and hence the extensive-form representation) of the game. Why should it be so hard?

Clearly, it is trivial in the case of normal-form games: we simply scan the possible payoffs and find whether there is one that gives an expected payoff of at least r. However, since the normal-form representation is exponential in the size of the extensive-form representation, this will not help. If there were no chance moves, the problem would also be trivial. We simply scan the payoffs; if there is one with a payoff of at least r, the answer is yes, since the players can just play the strategy that puts them on this path. However, with chance moves, this approach will not work, since we must take nature's move into account. This turns out to be hard, even if nature has only polynomially many possible moves.

3 The NP-Completeness Result

We now show the problem of determining the optimal strategy in a game with common payoffs is NP-complete. For simplicity, we restrict to two-player games. It will be clear from the proof that the result holds for arbitrary n-player games. For technical reasons, we further restrict to games whose probabilities and payoffs are rational numbers, and that the rational number $\frac{p}{q}$ is represented by pairs of integers (p, q), where $q > 0$. (This ensures that the games are finitely *described*, so that we can finish reading their descriptions in finite amount of time. If we are truly pedantic, we should actually fix an encoding of the games, but we will not get into such tedious details. Readers not familiar with NP-completeness results and the techniques by which they are established may wish to consult [Garey and Johnson 1979] for an introduction.) If the game is a common payoff game, we write $\text{PAY}(w, (a_1, \ldots, a_n)) = p$, where p is the common payoff. We then take $\mathbf{E}^{\text{PAY}}(\vec{S})$ to be a single number, rather than a tuple (all of whose components are identical).

The problem of finding the optimal strategy is an optimization problem. So that it fits into the standard framework of complexity classes, we convert it to a decision problem:

UTIL: Given a two-player game (with common payoff), is there a joint strategy $\vec{S}$ such that $\mathbf{E}^{\text{PAY}}(\vec{S}) \geq 1$?[1]

[1] Of course, there is nothing special about the choice of 1 here. The same result holds for an arbitrary (rational) r or if we take r as part of the input. In general, for hardness results, it suffices to focus on a special case.

The following theorem shows that UTIL is NP-complete.

Theorem 3.1: *UTIL is NP-complete.*

Proof: UTIL is clearly in NP, since it suffices to guess a joint strategy and verify that the expected joint payoff is indeed at least 1. To show that UTIL is NP-hard, we reduce 3SAT to UTIL. Recall that an *instance* of 3SAT consists of a Boolean formula which is a conjunction of *clauses*, each of which is a disjunction of three *literals*, each of which is either a (Boolean) variable or the negation of a (Boolean) variable. An instance of UTIL is simply a (finite) two-player game (with common payoff). A *positive instance of 3SAT* is a formula (of 3SAT) which is satisfiable (i.e., there is a truth assignment that satisfies all the clauses) while a *positive instance of UTIL* is a (finite) two-player game (with common payoff) for which there exists a joint strategy whose expected payoff is at least 1. Recall that to show that 3SAT reduces to UTIL, we need to give a polynomial-time transformation f that maps instances of 3SAT to instances of UTIL such that φ is a positive instance of 3SAT iff $f(\varphi)$ is a positive instance of UTIL.

Let φ be an instance of 3SAT with n clauses. Let n be the number of clauses in φ and let m be the number of variables in φ. (Note that $m \leq 3n$, since each clause contains at most three variables.) Let $z_1, \ldots, z_m$ be the variables and $C_1, \ldots, C_n$ be the clauses. Thus $\varphi = C_1 \wedge \cdots \wedge C_n$, where $C_i = (\ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3})$ is a clause, where $\ell_{i,j}$ is a literal. If $\ell_{i,j} = z_k$ or $\ell_{i,j} = \neg z_k$, we say that z_k is the variable *associated with* $\ell_{i,j}$; we denote the variable associated with a literal ℓ by $v(\ell)$.

Basically, the game proceeds as follows: nature chooses a clause C_i and literal $\ell_{i,j}$ in that clause. Each of the $3n$ choices of nature is equally likely. Player 1 observes the variable $v(\ell_{i,j})$ associated with the literal chosen by nature, and must then choose a truth value for that variable. Note that a strategy for player 1 is a truth assignment. Optimal play by player 1 will amount to choosing a truth assignment that satisfies φ. Player 2 observes the clause C_j chosen by nature and must choose a literal in that clause. Intuitively, this should be a literal that evaluates to **true** in the truth assignment chosen by player 1. The players get a payoff of 3 if the literal chosen by player 2 is the same as the one chosen by nature and it evaluates to **true** under the truth value for the variable chosen by player 1; otherwise they get 0.

Notice that the maximum expected payoff the players can get in this game is 1, since for each clause, in exactly one of the three worlds corresponding to that clause, player 2's choice will match nature's choice. In the n worlds where player 2's choice and nature's choice match, the players can get a payoff of 3 if player 2's choice evaluates to true under player 1's truth assignment. In all other worlds, they get a payoff of 0. Moreover, if φ is satisfiable, there is a joint strategy that gives the players an expected payoff of 1. Player 1 chooses a truth assignment α that satisfies φ (so that when player 1 observes the variable z, he plays $\alpha(z)$) and player 2, when given clause C_i, chooses a literal j in C_i that evaluates to true under truth assignment α. On the other hand, if φ is not satisfiable, there is no joint strategy that nets an expected payoff of 1, since for each truth assignment chosen by player 1, there is at least one clause where no literals are satisfied.

More formally, let $f(\varphi)$ be the following game:

- $W = \{w_{i,j} : 1 \leq i \leq n \text{ and } 1 \leq j \leq 3\}$ (the world $w_{i,j}$ corresponds to the literal $\ell_{i,j}$),
- $A_1 = \{\textbf{true}, \textbf{false}\}$,
- $A_2 = \{1, 2, 3\}$,
- $O_1 = \{z_1, \ldots, z_m\}$,
- $O_2 = \{C_1, \ldots, C_n\}$, and
- $\Pr(w_{i,j}) = \frac{1}{3n}$ for all i, j.
- $\textsc{obs}(w_{i,j}, 1) = v(\ell_{i,j})$ and $\textsc{obs}(w_{i,j}, 2) = C_i$; that is, in world $w_{i,j}$, the observation of agent 1 is $v(\ell_{i,j})$ and the observation of agent 2 is C_i.
- $\textsc{pay}$ is defined as follows:

$$\textsc{pay}(w_{i,j}, (\textbf{true}, j')) = \begin{cases} 3 & \text{if } j = j' \text{ and } \ell_{i,j} = v(\ell_{i,j}) \\ 0 & \text{otherwise} \end{cases}$$

$$\textsc{pay}(w_{i,j}, (\textbf{false}, j')) = \begin{cases} 3 & \text{if } j = j' \text{ and } \ell_{i,j} = \neg v(\ell_{i,j}) \\ 0 & \text{otherwise} \end{cases}$$

Note that the size of $f(\varphi)$ is linear in the number of clauses, so it is easy to implement f in polynomial time.

As an example, let $\varphi = (z_1 \vee \neg z_2 \vee z_3) \wedge (z_2 \vee z_4 \vee \neg z_1)$. Then $f(\varphi) = (W, A_1, A_2, O_1, O_2, \Pr, \textsc{obs}, \textsc{pay})$, where

- $W = \{w_{1,1}, w_{1,2}, w_{1,3}, w_{2,1}, w_{2,2}, w_{2,3}\}$,
- $A_1 = \{\textbf{true}, \textbf{false}\}$,
- $A_2 = \{1, 2, 3\}$,
- $O_1 = \{z_1, z_2, z_3, z_4\}$,
- $O_2 = \{(z_1 \vee \neg z_2 \vee z_3), (z_2 \vee z_4 \vee \neg z_1)\}$, and
- $\Pr(w_{i,j}) = \frac{1}{6}$.
- $\textsc{obs}$ is defined as follows:

– $\text{OBS}(w_{1,1}, 1) = z_1$, $\text{OBS}(w_{1,2}, 1) = z_2$, $\text{OBS}(w_{1,3}, 1) = z_3$,
  $\text{OBS}(w_{2,1}, 1) = z_2$, $\text{OBS}(w_{2,2}, 1) = z_4$, $\text{OBS}(w_{2,3}, 1) = z_1$

– $\text{OBS}(w_{1,*}, 2) = (z_1 \vee \neg z_2 \vee z_3)$,
  $\text{OBS}(w_{2,*}, 2) = (z_2 \vee z_4 \vee \neg z_1)$

- PAY is defined as follows:

	1	2	3
true	3	0	0
false	0	0	0

$w_{1,1}$

	1	2	3
true	0	0	0
false	0	3	0

$w_{1,2}$

	1	2	3
true	0	0	3
false	0	0	0

$w_{1,3}$

	1	2	3
true	3	0	0
false	0	0	0

$w_{2,1}$

	1	2	3
true	0	3	0
false	0	0	0

$w_{2,2}$

	1	2	3
true	0	0	0
false	0	0	3

$w_{2,3}$

As discussed earlier, it is easy to see that there is a joint strategy that gives the players an expected payoff of 1 in this game iff φ is satisfiable. Thus φ is a positive instance of 3SAT iff $f(\varphi)$ is a positive instance of UTIL, so we are done. ∎

Although we restricted to two-player games in Theorem 3.1, it should be clear that the analogous problem is also NP-complete for n-player games. The upper bound is clear, since it suffices to guess a joint strategy just as before. And it is easy to modify our lower bound proof to deal with n-player games; we leave details to the reader.

There is one technical point worth observing. Say that player i *considers* w_j *possible in* w_k iff $\text{OBS}(w_k, i) = \text{OBS}(w_j, i)$ (i.e., player i makes the same observation in both worlds). Intuitively, if a player considers many worlds possible, he has a lot of uncertainty. Note that in the game constructed in the proof of Theorem 3.1, player 2 considers only three worlds possible in any given world (since there are only three literals in each clause) while player 1 may consider many worlds possible in some worlds (since a variable may appear in many clauses). Is it necessary for one of the players to have much uncertainty for our result to hold? It turns out that the problem remains NP-complete even if player 1 considers at most three worlds possible in each world as well. The reason that player 1 may consider many worlds possible is because a variable may occur in many clauses. It is easy to convert a formula φ in which a variable may occur many times to a formula φ' in which each variable occurs in at most three clauses such that φ' is satisfiable iff φ is satisfiable and φ' can be constructed in polynomial time. (This can be done via a technique very similar to the reduction of SAT to 3SAT.) Thus the problem remains NP-complete even if both players consider at most three worlds possible in each world. While we know that 2SAT is in P and that SAT restricted to formulas in which each variable occurs at most twice is in P, we have not investigated whether UTIL remains

NP-complete if both players consider at most two worlds possible in each world. Clearly if both players consider only one world possible (i.e., they have perfect information), then we can find the optimal joint strategy in linear time.

4 Conclusion

We have shown that the problem of determining whether there is a joint strategy that nets at least r in a common payoff game in extensive form is NP complete, even if the there are only two players, each of whom makes only one move (following a move by nature). Essentially the same argument shows that it is NP-complete to find a strategy that is within a fixed fraction of optimal. Thus, we cannot even find approximately optimal strategies in polynomial time. What does this say about problems such as Free Flight? Should we necessarily give up on finding optimal strategies? Recent successes in finding solutions to NP-complete problems [Hogg, Huberman, and Williams 1996; Monasson, Zecchina, Kirkpatrick, Selman, and Troyansky 1999] suggest that there may be some reason to hope; NP-complete problems may not be so infeasible in practice. Of course, further research needs to be done to see if problems such as Free Flight can in fact be represented in a reasonable way as a game that is in practice soluble.

Acknowledgements

We would like to thank Ken Birman for bringing Free Flight to our attention, and Tzachi Gilboa for many thoughtful comments and useful suggestions for improving the paper.

References

Garey, M. and D. S. Johnson (1979). *Computers and Intractability: A Guide to the Theory of NP-completeness*. San Francisco, Calif.: W. Freeman and Co.

Gilboa, I. and E. Zemel (1989). Nash and correlated equilibrium: some complexity considerations. *Games and Economic Behavior 1*, 80–93.

Hogg, T., B. Huberman, and C. Williams (Eds.) (1996). *Artificial Intelligence*, Volume 81. Elsevier. Special Issue on *Phase Transitions and Complexity*.

Koller, D. and N. Megiddo (1992). The complexity of two-person zero-sum games in extensive form. *Games and Economic Behavior 4*(4), 528–552.

Monasson, R., R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky (1999). Typical-case complexity results from a new type of phase transition. *Nature 400*(8), 133–137.