

Approachability in repeated games: Computational aspects and a Stackelberg variant[☆]

Shie Mannor^a, John N. Tsitsiklis^{b,*}

^a Department of Electrical and Computer Engineering, McGill University, Montreal, PQ, H3A 2A7 Canada

^b Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Received 3 March 2006

Available online 11 April 2008

Abstract

We consider a finite two-player zero-sum game with vector-valued rewards. We study the question of whether a given polyhedral set D is “approachable,” that is, whether Player 1 (the “decision maker”) can guarantee that the long-term average reward belongs to D , for any strategy of Player 2 (the “adversary”). We examine Blackwell’s necessary and sufficient conditions for approachability, and show that the problem of checking these conditions is NP-hard, even in the special case where D is a singleton. We then consider a Stackelberg variant whereby, at each stage, the adversary gets to act after observing the decision maker’s action. We provide necessary and sufficient conditions for approachability, and again establish that checking these conditions is NP-hard, even when D is a singleton. On the other hand, if the dimension of the reward vector is fixed, an approximate version of these conditions can be checked in polynomial time.

© 2008 Elsevier Inc. All rights reserved.

JEL classification: C73; C79

1. Introduction

We consider a decision maker (Player 1, P_1) who interacts repeatedly with the environment, modeled as an adversary (Player 2, P_2). At each stage (time step), each player chooses an action from given finite sets and a vector-valued reward is realized, as a function of the pair of actions chosen. We are given a polyhedral set D , and we are interested in the question of whether there exists a strategy for P_1 under which the long-term average of the reward vector is guaranteed to belong to D , for every strategy of P_2 (in which case, we say that D is “approachable”). This problem was introduced and studied by Blackwell (1956), using the tools of what became known as “approachability theory.” In particular, Blackwell established necessary and sufficient conditions for the case of a convex set D , as well as a prescription for the strategy of P_1 . However, despite the pervasiveness and the renewed interest in approachability theory

[☆] This research was supported by the National Science Foundation under contract ECS-0312921, and by the Natural Sciences and Engineering Research Council of Canada.

* Corresponding author.

E-mail addresses: shie.mannor@mcgill.ca (S. Mannor), jnt@mit.edu (J.N. Tsitsiklis).

in the context of learning in games, the computational aspects of this theory do not seem to have been considered before. The objective of this paper is to close this gap.

We start in Section 2 with a description of the model, and background results from approachability theory (Blackwell's conditions). Even though in some cases it is fairly straightforward to check whether a set is approachable or not (see Hart and Mas-Colell, 2001 for a representative example), we show that checking Blackwell's conditions is an NP-hard problem, even in the special case where the set D is a singleton. On the other hand, when the dimension of the reward vector is fixed, we establish that the question of approachability can be decided “approximately” in time which is polynomial (though exponential in the dimension of the reward vector).

Blackwell's formulation and conditions refer to the case where, at each stage, the two players act simultaneously, without knowledge of the other player's action. In Section 3, we introduce a Stackelberg variant in which, at each stage, P_1 acts first and P_2 (the adversary) is informed of P_1 's action before choosing her own action. We establish that the question of approachability is NP-hard, even in the special case where the set D is a singleton, but can be decided in polynomial time if the dimension of the reward vector is held fixed.

2. Approachability for the case of simultaneous actions

2.1. Model and background

We consider a repeated game where a decision maker wishes to guarantee that the long-term average of a vector-valued reward belongs to a prespecified target set. The stage game is a finite game involving two players, P_1 (the decision maker) and P_2 (the adversary). This naturally abstracts the case where there are multiple players and we are only concerned with the reward obtained by P_1 .

The game is defined by a triple $(\mathcal{A}, \mathcal{B}, M)$ where:

- (a) \mathcal{A} is the finite set of actions for P_1 ; we will assume that $\mathcal{A} = \{1, 2, \dots, m\}$.
- (b) \mathcal{B} is the finite set of actions for P_2 ; we will assume that $\mathcal{B} = \{1, 2, \dots, n\}$.
- (c) M is an $n \times m$ matrix with vector-valued entries, with $M(a, b)$ denoting the reward obtained by P_1 , when P_1 chooses action $a \in \mathcal{A}$, and P_2 chooses action $b \in \mathcal{B}$; we will assume that $M(a, b) \in \mathbb{R}^k$.¹

The game is played in stages. At each stage t , P_1 chooses an action $a_t \in A$, P_2 chooses an action $b_t \in B$, and P_1 obtains a reward $m_t = M(a_t, b_t)$. We define P_1 's average reward, at time t , as

$$\hat{m}_t = \frac{1}{t} \sum_{\tau=1}^t m_\tau.$$

We further assume that P_1 has a prespecified target set $D \in \mathbb{R}^k$, assumed to be a polyhedron.² The goal of P_1 is to have the average reward \hat{m}_t approach this set D , as t increases, in a sense to be made precise below.

For a finite set \mathcal{C} , we let $\Delta(\mathcal{C})$ be the set of all probability measures on a set \mathcal{C} , which is identified with the set of $|\mathcal{C}|$ -dimensional nonnegative vectors whose entries sum to one, and which will be referred to as the set of possible mixed actions on \mathcal{C} . A strategy for P_1 (respectively, P_2) is a mapping from all possible histories of the form $(a_1, b_1, \dots, a_{t-1}, b_{t-1})$ to the set of mixed actions on \mathcal{A} (respectively, \mathcal{B}). Given the strategies of the two players, we assume that the randomizations involved are all independent. We use $\|\cdot\|$ to denote the Euclidean norm in \mathbb{R}^k , and define the point-to-set distance $\rho(x, D) = \inf_{y \in D} \|x - y\|$. We now define formally the goal of P_1 .

Definition 2.1. A set D is *approachable* if there exists a strategy σ of P_1 such that for every $\varepsilon, \delta > 0$, there exists t_0 such that for every strategy τ of P_2 ,

$$\mathbf{P}\left(\sup_{t \geq t_0} \rho(\hat{m}_t, D) > \varepsilon\right) < \delta. \quad (1)$$

¹ All of the subsequent development also applies to the case where the rewards are random variables, sampled independently at each time, with mean $M(a_t, b_t)$ and finite second moment. We restrict to the deterministic case for simplicity.

² We restrict to polyhedral sets, as opposed to the general convex sets considered by Blackwell, because we wish to focus on algorithmic aspects.

A set D is *excludable* if there exists a strategy τ of P_2 and some $\varepsilon > 0$ such that for every $\delta > 0$, there exists t_0 such that for every strategy σ of P_1 ,

$$\mathbf{P}\left(\inf_{t \geq t_0} \rho(\hat{m}_t, D) < \varepsilon\right) < \delta. \quad (2)$$

In both cases, the probability measure is induced by the randomness in the strategies of P_1 and P_2 .

Obviously, a set cannot be both approachable and excludable. But when D is a polyhedron (as assumed in this paper), it is guaranteed to be either approachable or excludable (this is not true for general sets).

For any $a \in \mathcal{A}$ and $q \in \Delta(\mathcal{B})$, we define

$$M(a, q) = \sum_{b \in \mathcal{B}} M(a, b) q_b,$$

and for any $q \in \Delta(\mathcal{B})$, we define

$$Q(q) = \text{conv}(\{M(a, q)\}_{a \in \mathcal{A}}),$$

so that $Q(q)$ is the convex hull of the set of expected rewards that can be obtained by varying a . The necessary and sufficient conditions for approachability are as follows (Blackwell, 1956).

Theorem 2.1. *The following are equivalent:*

- (a) *The polyhedron D is not approachable.*
- (b) *The polyhedron D is excludable.*
- (c) *There exists some $q \in \Delta(\mathcal{B})$, such that $D \cap Q(q) = \emptyset$.*
- (d) *There exists some $d \in \mathbb{R}^k$ and some $q \in \Delta(\mathcal{B})$, such that*

$$\max_{x \in D} d^\top x < d^\top M(a, q), \quad \forall a \in \mathcal{A}.$$

The equivalence of conditions (a)–(c) above was established in (Blackwell, 1956).³ The equivalence with condition (d) is an immediate consequence of the separating hyperplane theorem.

We now specialize to the case where the set D is a singleton. Without loss of generality, we will assume that $D = \{0\}$.

Corollary 2.1. *The set $\{0\}$ is excludable if and only if there exists some $d \in \mathbb{R}^k$ and some $q \in \Delta(\mathcal{B})$, such that $0 < d^\top M(a, q)$, $\forall a \in \mathcal{A}$.*

2.2. NP-hardness

Since $M(a, q)$ is linear in q , we see that checking the condition in Corollary 2.1 is equivalent to solving the following problem:

EXCLUDABILITY. Given a finite collection of $k \times n$ matrices A_a , with rational entries, determine whether there exists a solution $(d, q) \in \mathbb{R}^k \times \mathbb{R}^n$ to the system of inequalities

$$q \geq 0, \quad \sum_{b \in \mathcal{B}} q_b = 1, \quad 0 < d^\top A_a q, \quad \forall a, \quad (3)$$

where the inequality $q \geq 0$ is interpreted componentwise.

It is well known that solving a system of general quadratic inequalities is NP-hard (see Murty and Kabadi, 1987). Our result below boils down to a proof that the special case that involves only bilinear inequalities and nonnegativity

³ Blackwell's model considers strategies that depend on the history of past rewards, as opposed to the history of past actions of the two players, but the same result (with essentially the same proof) is valid for our setting as well.

constraints is also NP-hard. It is not known whether the problem belongs to NP, because if the inequalities have a feasible solution, there is no guarantee that there exists a feasible solution with a polynomial number of bits.

Theorem 2.2. EXCLUDABILITY is NP-hard. In particular, the problem of deciding whether the set $D = \{0\}$ is approachable is NP-hard.

Proof. We provide a reduction of the 3-satisfiability problem (3SAT), which is NP-complete, to EXCLUDABILITY. An instance of satisfiability consists of n variables, x_1, \dots, x_n , and m clauses C_1, \dots, C_m , where each clause is a disjunction of three literals, and where a literal is a variable or its negation. The question is whether there exists an assignment of truth values to the literals so that all clauses are satisfied.

Given an instance of 3SAT with n literals and m clauses, we construct an instance of EXCLUDABILITY, as follows. We let the dimension of d and of q be equal to $2n$, and use the notation $d = (d_1, \bar{d}_1, d_2, \dots, \bar{d}_n)$, $q = (q_1, \bar{q}_1, q_2, \dots, \bar{q}_n)$. We also let $d_0 = \sum_{i=1}^n (d_i + \bar{d}_i)$ and $q_0 = \sum_{i=1}^n (q_i + \bar{q}_i)$. In addition to the conditions $q \geq 0$ and $q_0 = \sum_{i=1}^n (q_i + \bar{q}_i) = 1$, we introduce the following bilinear inequalities:

$$d_i q_i + \bar{d}_i \bar{q}_i < \frac{1}{3n} d_0 (q_i + \bar{q}_i), \quad i = 1, \dots, n, \quad (4)$$

(note that d_0 is a linear combination of the d_i and \bar{d}_i , which makes (4) a bilinear inequality) and

$$d_i q_i > 0, \quad \bar{d}_i \bar{q}_i > 0, \quad i = 1, \dots, n. \quad (5)$$

For each clause of the form $(x_i \vee x_j \vee x_k)$, we introduce the inequality

$$(d_i + d_j + d_k) q_0 > \frac{d_0}{n} q_0. \quad (6)$$

In case a variable x_i appears negated in a clause, we write down a similar constraint, except that d_i is replaced by \bar{d}_i . For example, given the clause $(\bar{x}_i \vee x_j \vee \bar{x}_k)$, we introduce the inequality

$$(\bar{d}_i + d_j + \bar{d}_k) q_0 > \frac{d_0}{n} q_0. \quad (7)$$

Suppose that the instance of 3SAT is satisfiable, and consider a satisfying assignment. If x_i is “true,” we set

$$d_i = 10, \quad \bar{d}_i = 1, \quad q_i = \frac{1}{10n}, \quad \bar{q}_i = \frac{9}{10n}.$$

If x_i is “false,” we set

$$d_i = 1, \quad \bar{d}_i = 10, \quad q_i = \frac{9}{10n}, \quad \bar{q}_i = \frac{1}{10n}.$$

Constraints (4) and (5) are satisfied by construction. Since every clause is satisfied, we see that in every constraint associated with a clause (e.g., constraint (6) or (7)), at least one of the d_i or \bar{d}_i variables in the left-hand side is set to 10, so that the left-hand side is at least equal to 12. Since the right-hand side evaluates to 11, such a constraint is satisfied. Thus, all constraints are satisfied, and we conclude that we have a “yes” instance of EXCLUDABILITY.

For the converse, suppose that we have a solution (d, q) to the system of inequalities. We construct a truth assignment by setting each variable x_i to “true” if and only if $d_i \geq \bar{d}_i$. Since every d_i is positive (due to (5)), inequality (4) implies that either $d_i < d_0/3n$ or $\bar{d}_i < d_0/3n$. Therefore, if x_i is set to false (respectively, true), then $d_i < d_0/3n$ (respectively, $\bar{d}_i < d_0/3n$). Consider a typical clause, say, $(\bar{x}_i \vee x_j \vee \bar{x}_k)$, and the associated constraint (7). Since the constraint is satisfied, at least one of the variables $\bar{d}_i, d_j, \bar{d}_k$ must be at least $d_0/3n$, which implies that at least one of the literals $\bar{x}_i, x_j, \bar{x}_k$ is true and the clause is satisfied. We conclude that the instance of 3SAT is satisfiable, which completes the reduction. \square

As an immediate corollary of Theorem 2.2, we see that the problem of deciding excludability of a polyhedral set D is NP-hard. In many contexts, the specifications of desired performance, as captured by the set D , consist of minimum requirements for each component of the reward vector. Equivalently, the set D is a translated orthant. This case is addressed by the following result.

Corollary 2.2. EXCLUDABILITY is NP-hard even for the special case where the set D is the nonnegative orthant, $\{y \in \mathbb{R}^k \mid y \geq 0\}$.

Proof. Consider condition (d) in Theorem 2.1. If d has a positive component then $\max_{x \in D} d^\top x = \infty$. Therefore, for the case where D is the nonnegative orthant, excludability is equivalent to the condition that there exist some $d \leq 0$ and some $q \in \Delta(\mathcal{B})$ such that $0 < d^\top M(a, q)$ for all $a \in \mathcal{A}$.

The proof of Theorem 2.2 applies without change, except that the inequalities $d_i q_i > 0$, $\bar{d}_i \bar{q}_i > 0$ are no longer necessary. \square

If we restrict the reward dimension k , or the cardinality m or n of either action set, it is not known whether the problem remains NP-hard or whether a polynomial time algorithm is possible. However, as we show in the next section, approachability can be decided in an approximate sense, when the dimension is held fixed.

2.3. Approximate approachability

In this section, we provide an algorithm that determines whether a set is approximately approachable or not, and quantifies (approximately) the degree to which a set is not approachable. In particular, we are interested in approximating the smallest constant ε such that there exists a strategy that guarantees that the distance between the average reward and the set D is asymptotically less than ε . We will use the following definitions.

Definition 2.2.

(a) For any $\varepsilon > 0$, the ε -expansion of a set D , denoted D^ε , is

$$D^\varepsilon = \{x: \rho(x, D) \leq \varepsilon\}.$$

(b) A set of vectors $\mathcal{C} = \{d_1, \dots, d_\ell\}$ is a δ -covering of a set $U \subset \mathbb{R}^k$ if for every $u \in U$ there exists $d_i \in \mathcal{C}$ such that $\|u - d_i\| \leq \delta$.

The following algorithm receives as input two threshold levels, ε_1 and ε_2 (where $\varepsilon_1 < \varepsilon_2$), and returns one of the following: that D^{ε_1} is excludable, or that D^{ε_2} is approachable. Note that these two statements are *not* mutually exclusive; if both are true, either statement may be returned. The algorithm essentially consists of checking the excludability condition (d) in Theorem 2.1 only for a finite set of direction vectors d .

Algorithm: APPROACHorEXCLUDE($\varepsilon_1, \varepsilon_2; D$)

Input:

(a) A polyhedron D specified either in terms of linear inequality constraints or in terms of its extreme points, and an $m \times n$ matrix M with entries $M(a, b) \in \mathbb{R}^k$.

(b) Parameters ε_1 and ε_2 , with $0 < \varepsilon_1 < \varepsilon_2$.

Steps:

(a) Let

$$\delta \triangleq \frac{\varepsilon_2 - \varepsilon_1}{2 \max\{\max_{x \in D} \|x\| + \varepsilon_1, \max_{a,b} \|M(a, b)\|\}},$$

and construct a δ -covering $\mathcal{C} = \{d_1, d_2, \dots, d_\ell\}$ of the unit ball in \mathbb{R}^k .

(b) For every $d_i \in \mathcal{C}$, check whether there exists some $q \in \Delta(\mathcal{B})$ such that

$$\max_{x \in D^{\varepsilon_1}} d_i^\top x < d_i^\top M(a, q), \quad \forall a \in \mathcal{A}. \tag{8}$$

If there exists some $d_i \in \mathcal{C}$ and some $q \in \Delta(\mathcal{B})$ such that (8) is satisfied, declare that D^{ε_1} is excludable and terminate.

(c) If the algorithm did not terminate earlier, declare that D^{ε_2} is approachable.

Theorem 2.3. *The APPROACH or EXCLUDE algorithm is correct and its complexity is polynomial in the parameters of D (number of extreme points or inequalities, and the diameter of D), the parameters of the game (m and n), and $1/(\varepsilon_2 - \varepsilon_1)$. It is, however, exponential in k .*

Proof. We start with the proof of correctness. If the algorithm declares that D^{ε_1} is excludable, correctness follows because condition (d) of Theorem 2.1 is satisfied.

Suppose now that the algorithm declares that D^{ε_2} is approachable, so that for every $d_i \in \mathcal{C}$ there exists no $q \in \Delta(\mathcal{B})$ for which the condition (8) is satisfied. Since D is convex, so is D^{ε_2} . It follows that D^{ε_2} is either approachable or excludable. Suppose, in order to derive a contradiction, that D^{ε_2} is excludable. Then there exist $d \in \mathbb{R}^k$ and $q \in \Delta(\mathcal{B})$ such that

$$\max_{x \in D^{\varepsilon_2}} d^\top x < d^\top M(a, q), \quad \forall a \in \mathcal{A}. \quad (9)$$

Fix some d and q with this property, and normalize d so that $\|d\| = 1$. We have

$$\begin{aligned} \max_{x \in D^{\varepsilon_2}} d^\top x &= \max_{x \in D^{\varepsilon_1}} d^\top x + (\varepsilon_2 - \varepsilon_1) \\ &= \max_{x \in D^{\varepsilon_1}} ((d - d_i)^\top x + d_i^\top x) + (\varepsilon_2 - \varepsilon_1) \\ &\geq \max_{x \in D^{\varepsilon_1}} d_i^\top x - \|d - d_i\| \max_{x \in D^{\varepsilon_1}} \|x\| + (\varepsilon_2 - \varepsilon_1), \end{aligned} \quad (10)$$

where the first equality holds because if x^* maximizes $d^\top x$ on D^{ε_1} , the vector $x^* + d^\top(\varepsilon_2 - \varepsilon_1)$ belongs to D^{ε_2} and maximizes $d^\top x$ on D^{ε_2} . Since \mathcal{C} is a δ -covering, we can now choose some $d_i \in \mathcal{C}$ that satisfies $\|d - d_i\| \leq \delta$. Using the definition of δ , Eq. (10) becomes

$$\max_{x \in D^{\varepsilon_2}} d^\top x \geq \max_{x \in D^{\varepsilon_1}} d_i^\top x + (\varepsilon_2 - \varepsilon_1)/2.$$

Since the condition (8) was not satisfied for this particular d_i and q , there exists some a such that

$$\max_{x \in D^{\varepsilon_1}} d^\top x \geq d_i^\top M(a, q).$$

It follows that

$$\begin{aligned} \max_{x \in D^{\varepsilon_2}} d^\top x &\geq d_i^\top M(a, q) + (\varepsilon_2 - \varepsilon_1)/2 \\ &= d^\top M(a, q) + (d_i - d)^\top M(a, q) + (\varepsilon_2 - \varepsilon_1)/2 \\ &\geq d^\top M(a, q), \end{aligned}$$

where in the last inequality we used again the condition $\|d - d_i\| \leq \delta$ and the definition of δ . This contradicts (9), and concludes the proof of correctness.

We now consider the complexity of the algorithm. The number ℓ of elements in the covering \mathcal{C} is

$$O\left(\left(\frac{(\max_{a,b} \|M(a, b)\| + \max_{x \in D} \|x\| + \varepsilon_1)}{\varepsilon_2 - \varepsilon_1}\right)^k\right),$$

where the $O(\cdot)$ notation hides a constant that only depends on the dimension k . For every $d_i \in \mathcal{C}$ we need to check whether there exists some $q \in \Delta(\mathcal{B})$ for which (8) is satisfied. This is equivalent to checking whether

$$\max_{x \in D^{\varepsilon_1}} d_i^\top x < \max_{q \in \Delta(\mathcal{B})} \min_{a \in \mathcal{A}} d_i^\top M(a, q).$$

As noted earlier, to perform the maximization on the left-hand side, it suffices to maximize $d_i^\top x$ over the polyhedron D , which is a linear programming problem.

As for the right-hand side, it is the value of a finite zero-sum game, which can also be computed using linear programming. The complexity of solving each linear program is independent of ε_1 or ε_2 , and depends polynomially on the parameters of the game and of D . \square

Suppose now that we are given a set D , and let η be the smallest ε such that D^ε is approachable (equivalently, the largest ε such that D^ε is excludable for every $\varepsilon < \eta$). The following algorithm calculates η approximately. Its parameters are a desired approximation level $\delta > 0$ and an upper bound κ on η .

Algorithm: RANGEAPPROACH($\delta, \kappa; D$)

Input:

- (a) A polyhedron D specified either in terms of linear inequality constraints, or in terms of its extreme points, and an $m \times n$ matrix M with entries $M(a, b) \in \mathbb{R}^k$.
- (b) Parameters δ and κ , with $0 < \delta < \kappa$.

Steps:

- (a) Set $L = 0$ and $U = \kappa$.
- (b) Repeat until $U - L \leq \delta$:
 - (b-i) Run APPROACHorEXCLUDE($(U + L)/3, 2(U + L)/3; D$).
 - (b-ii) If it returns that $D^{(U+L)/3}$ is excludable, set $L := (U + L)/3$.
 - (b-iii) If it returns that $D^{2(U+L)/3}$ is approachable, set $U := (U + L)/3$.
- (c) Return U and L .

Theorem 2.4. *RANGEAPPROACH($\delta, \kappa; D$) returns U and L such that $L \leq \eta \leq U$. The number of times APPROACHorEXCLUDE is called is $\lceil \log_{2/3}(\delta/\kappa) \rceil$. The complexity of the RANGEAPPROACH algorithm is polynomial in the parameters of the game and of D , and in κ/δ . It is, however, exponential in k .*

Proof. This a straightforward consequence of Theorem 2.3. \square

3. Stackelberg variants

In this section we consider a variant in which P_1 acts as the leader, and P_2 as a follower who can take into account the action chosen by P_1 at the current stage. We provide conditions for approachability (Section 3.1), and establish NP-completeness of the problem of checking these conditions (Section 3.2). In contrast, we present a polynomial algorithm for the special case where the dimension of the reward vector is fixed (Section 3.3). Finally, the variant where P_1 is the follower and P_2 is the leader, is briefly discussed in Section 3.4.

3.1. The model and the approachability conditions

Formally, a strategy for the leader, P_1 (respectively, the follower, P_2), is a function that maps any possible history of the form $(a_1, b_1, \dots, a_{t-1}, b_{t-1})$ (respectively, $(a_1, b_1, \dots, a_{t-1}, b_{t-1}, a_t)$) to a mixed action on \mathcal{A} (respectively, \mathcal{B}). This setting can be viewed as a special case of the setting in the preceding section, except that we now have a new action set $\bar{\mathcal{B}}$ for P_2 , namely the set $\bar{\mathcal{B}} = \mathcal{B}^{\mathcal{A}}$, of mappings from \mathcal{A} to \mathcal{B} . Accordingly, we define the reward vector to be

$$\bar{M}(a, f) = M(a, f(a)), \quad a \in \mathcal{A}, \quad f \in \bar{\mathcal{B}} = \mathcal{B}^{\mathcal{A}}.$$

As before, we allow the players to choose mixed actions. A mixed action for P_2 will now be an element of $\Delta(\mathcal{B}^{\mathcal{A}})$.⁴ For any $q \in \Delta(\mathcal{B}^{\mathcal{A}})$, we let

$$\bar{M}(a, q) = \sum_{f \in \mathcal{B}^{\mathcal{A}}} M(a, f(a))q_f.$$

We have the following counterpart of Theorem 2.1.

⁴ With our definition, P_2 randomizes between different responses to the action of P_1 . In an alternative formulation, we could have P_2 carry out a separate randomization for every action of P_1 . The resulting set of mixed actions would then be $(\Delta(\mathcal{B}))^{\mathcal{A}}$. It can be checked that the two formulations are equivalent.

Theorem 3.1. For the Stackelberg variant in which player P_1 leads, the polyhedron D is excludable if and only if

$$\max_{x \in D} d^\top x < d^\top M(a, f(a)), \quad \forall a \in \mathcal{A}. \quad (11)$$

Proof. Comparing condition (11) with condition (d) in Theorem 2.1, we only need to show that if there exists some $d \in \mathbb{R}^k$ and some $q \in \Delta(\mathcal{B}^{\mathcal{A}})$ such that $\max_{x \in D} d^\top x < d^\top \bar{M}(a, q)$ for all $a \in \mathcal{A}$, then there also exists a q that corresponds to a pure action (i.e., an element of $\mathcal{B}^{\mathcal{A}}$) and for which the same condition holds. Indeed, if such a $q \in \Delta(\mathcal{B}^{\mathcal{A}})$ exists, then for any $a \in \mathcal{A}$, we have

$$\begin{aligned} \min_{x \in D} d^\top x &< d^\top \bar{M}(a, q) \\ &= d^\top \sum_f M(a, f(a)) q_f \\ &= d^\top \sum_b \sum_{\{f: f(a)=b\}} M(a, f(a)) q_f \\ &= d^\top \sum_b M(a, b) \sum_{\{f: f(a)=b\}} q_f \\ &\leq \max_{b \in \mathcal{B}} d^\top \sum_b M(a, b). \end{aligned}$$

This implies that for every $a \in \mathcal{A}$ there exists some $b \in \mathcal{B}$ such that $\max_{x \in D} d^\top x < d^\top M(a, b)$, which is equivalent to condition (11) in the statement of the theorem. \square

3.2. NP-completeness

We now establish that checking the conditions for excludability, for this Stackelberg variant, is also an NP-complete problem, even in the special case where D is a singleton, e.g., if $D = \{0\}$. Condition (11) in Theorem 3.1 leads to the following problem.

S-EXCLUDABILITY. Given finite sets \mathcal{A} and \mathcal{B} , and a rational vector-valued function $M : \mathcal{A} \times \mathcal{B} \rightarrow \mathbb{R}^k$, does there exist some $d \in \mathbb{R}^k$ and a function $f : \mathcal{A} \rightarrow \mathcal{B}$ such that $d^\top M(a, f(a)) < 0$ for all $a \in \mathcal{A}$?

Theorem 3.2. S-EXCLUDABILITY is NP-complete.

Proof. We first establish that the problem is in NP. If we have a “yes” instance of S-EXCLUDABILITY, then there exists a mapping $f : \mathcal{A} \rightarrow \mathcal{B}$ (which will be the certificate) such that the zero vector does not belong to the polyhedron with vertices $M(a, f(a))$, $a \in \mathcal{A}$, and the latter condition can be checked in polynomial time using a linear programming algorithm.

For the proof of NP-completeness, we introduce the following auxiliary problem, which will be shown shortly to be NP-complete.

NONCOVERING CONES. Given m polyhedral cones $C_a \subset \mathbb{R}^k$, $a \in \mathcal{A}$, each specified by n linear inequalities, is their union a proper subset of \mathbb{R}^n ?

An instance of NONCOVERING CONES can be reduced to S-EXCLUDABILITY, as follows. Since we are given the linear inequalities that define C_a , we can define vectors $M(a, b)$ such that $C_a = \{d \in \mathbb{R}^k \mid d^\top M(a, b) \leq 0, b = 1, \dots, n\}$. We then observe that we have a “yes” instance of NONCOVERING CONES if and only if there is some d that violates one of the constraints for each C_a , that is, if and only if for every a there exists some $f(a)$ such that $0 < d^\top M(a, f(a))$, which is equivalent to having a “yes” instance of S-EXCLUDABILITY.

To complete the proof, we reduce 3SAT to NONCOVERING CONES, thus establishing NP-completeness. Given an instance of 3SAT with n variables and m literals, we let $\varepsilon = 1/4$, and introduce the following cones in \mathbb{R}^{n+1} :

$$\{d \mid d_0 \leq 0\}, \quad (12)$$

$$\{d \mid d_i \geq d_0(1 + \varepsilon)\}, \quad i = 1, \dots, n, \quad (13)$$

$$\{d \mid d_i \leq -d_0\varepsilon\}, \quad i = 1, \dots, n, \quad (14)$$

$$\{d \mid \varepsilon d_0 \leq d_i \leq d_0(1 - \varepsilon)\}, \quad i = 1, \dots, n. \quad (15)$$

For any clause of the form $x_i \vee x_j \vee x_k$, we introduce the cone

$$\{d \mid d_i + d_j + d_k \leq 3\varepsilon d_0\}. \quad (16)$$

For any clause of the form $\bar{x}_i \vee x_j \vee x_k$, we introduce the cone

$$\{d \mid -d_i + d_j + d_k \leq (-1 + 3\varepsilon)d_0\}. \quad (17)$$

For any clause of the form $\bar{x}_i \vee \bar{x}_j \vee x_k$, we introduce the cone

$$\{d \mid -d_i - d_j + d_k \leq (-2 + 3\varepsilon)d_0\}. \quad (18)$$

For any clause of the form $\bar{x}_i \vee \bar{x}_j \vee \bar{x}_k$, we introduce the cone

$$\{d \mid -d_i - d_j - d_k \leq (-3 + 3\varepsilon)d_0\}. \quad (19)$$

Suppose that we have a “yes” instance of 3SAT and a corresponding truth assignment. Let $d_0 = 1$ and for each i , let $d_i = 1$ (respectively, $d_i = 0$) if the variable x_i is set to “true” (respectively, false). We claim that d does not belong to any of the above defined cones. This is obvious for the first four types of cones (cf. (12)–(15)). Consider a clause of the form $x_i \vee x_j \vee x_k$. Since the clause is satisfied, at least one of the variables d_i, d_j, d_k is set to 1, which is more than $3\varepsilon d_0$, and d does not belong to the cone associated with that clause. Consider then a clause of the form $\bar{x}_i \vee x_j \vee x_k$. Since the clause is satisfied, it is seen that $-d_i + d_j + d_k$ is at least 0, which is more than $(-1 + 3\varepsilon)d_0$, and d does not belong to the cone associated with that clause either. By a similar argument, d does not belong to the last two types of cones, which shows that we have a “yes” instance of NONCOVERING CONES.

For the converse, suppose that we have a “yes” instance of NONCOVERING CONES. Fix some d that does not lie in any of the above constructed cones. Since d does not belong to the cone specified by (12), we have $d_0 > 0$; without loss of generality, we can and will assume that $d_0 = 1$. Since d does not belong to the cones of the form (13), (14), and (15), we conclude that every $d_i, i \neq 0$, satisfies either $-\varepsilon < d_i < \varepsilon$, in which case we set x_i to “false,” or $1 - \varepsilon < d_i < 1 + \varepsilon$, in which case we set x_i to true. We now verify that this truth assignment satisfies all clauses. Indeed, for any clause of the form $x_i \vee x_j \vee x_k$, since d does not lie in the corresponding cone, we have $d_i + d_j + d_k > 3\varepsilon$, which implies that at least one of the variables d_i, d_j, d_k exceeds ε , implying that at least one of the variables x_i, x_j, x_k is set to “true.” Similarly, for any clause of the form $\bar{x}_i \vee x_j \vee x_k$, since d does not lie in the corresponding cone (cf. (17)), we have $-d_i + d_j + d_k > -1 + 3\varepsilon$. This implies that not all of the conditions $d_i > 1 - \varepsilon, d_j < \varepsilon, d_k < \varepsilon$ can hold. Therefore, either x_i is set to “false” or one of the variables x_j, x_k is set to true, and the clause is satisfied. The argument for the last two types of clauses is similar. \square

We make a few brief remarks. (i) The problem is clearly NP-complete for the more general case where the set D is a general polyhedron. (ii) A minor modification of the proof establishes NP-completeness for the case where D is the positive orthant. (iii) The proof actually establishes that the problem is NP-complete even in the special case where the adversary has only two actions.

3.3. Low-dimensional rewards

In this subsection, we fix the dimension k of the reward vector, and show that approachability of a polyhedral target set D can be decided in polynomial time.

A naive algorithm, based on Theorem 3.1 would check condition (11) for every $f \in \mathcal{B}^A$, and there are exponentially many such f to be considered. To illustrate the key idea behind the algorithm given here, let us assume that $D = \{0\}$ and that for some f , the vector 0 does not belong to the convex hull of the points $M(a, f(a))$, so that there is a direction vector d that satisfies (11). It turns out that the search for such a vector d can be restricted to a smaller set of candidate directions, namely those directions obtained by projecting the zero vector on the convex hull of only $k + 1$

of the points $M(a, f(a))$. The number of possible choices of these $k + 1$ points is polynomially bounded (when k is fixed), suggesting the algorithm to be given shortly.

Some terminology first. A *partial response* g is defined as a specification

$$((a_1, b_1), (a_2, b_2), \dots, (a_{k+1}, b_{k+1}))$$

of a mapping $f : \mathcal{A} \rightarrow \mathcal{B}$ on only $k + 1$ distinct elements of \mathcal{A} . Let $P(g)$ be the convex hull of $\{M(a_i, b_i) \mid i = 1, \dots, k + 1\}$. The following algorithm assumes that $k + 1 \leq m$. If $m < k + 1$, we can always introduce $k + 1 - m$ new actions a with $M(a, b) = M(1, b)$ for all b , which does not affect the nature of the game.

Algorithm

Input:

A polyhedron D specified either in terms of linear inequality constraints, or in terms of its extreme points, and an $m \times n$ matrix M with entries $M(a, b) \in \mathbb{R}^k$, where $k + 1 \leq m$.

Steps:

- (a) For every partial response g , of the form $((a_1, b_1), (a_2, b_2), \dots, (a_{k+1}, b_{k+1}))$, do the following:
 - (a-i) Find $x \in D$ and $y \in P(g)$ for which $\rho(x, y)$ is minimized.
 - (a-ii) If $x \neq y$ check whether there exists a response $f : \mathcal{A} \rightarrow \mathcal{B}$, that extends g to all of \mathcal{A} , and such that $(y - x)^\top y \leq (y - x)^\top M(a, f(a))$ for all $a \in \mathcal{A}$. If such an f is found, return “no” (declare the set D excludable) and terminate.
- (b) If the algorithm did not terminate earlier, return “yes” (declare the set D approachable).

Theorem 3.3. *The above algorithm is correct, and runs in polynomial time, when the dimension k is held fixed.*

Proof. The algorithm is polynomial because for each partial response g (of which there are at most $(mn)^{k+1}$), it mainly needs to solve a convex quadratic programming problem (Step (a-i)), which takes polynomial time (Kozlov et al., 1979). We also note that the algorithm always terminates. If it returns “no,” the response f found in Step (a-ii) provides a certificate that D is excludable.

Suppose now that D is excludable. By Theorem 3.1, there exists some $f \in \mathcal{B}^A$ such that D does not intersect the convex hull of the points $M(a, f(a))$, $a \in \mathcal{A}$, which we denote by $Q(f)$. Let $x \in D$ and $z \in Q(f)$ be such that the distance $\rho(x, z)$ is minimal. Since D and $Q(f)$ are disjoint, we have $\rho(x, z) > 0$, and the direction $d = z - x$ defines a hyperplane that separates D from $Q(f)$, that is,

$$\max_{w \in D} d^\top w \leq d^\top x < d^\top z \leq \min_{w \in Q(f)} d^\top w. \quad (20)$$

Consider the polyhedron $Q_0(f) = \{w \in Q(f) \mid d^\top w = d^\top z\}$. By Caratheodory’s theorem z can be written as a convex combination of at most $k + 1$ extreme points of $Q_0(f)$. These extreme points are also extreme points of $Q(f)$ (see, e.g., Bertsimas and Tsitsiklis, 1999, p. 66), and therefore they are a subset of the points $M(a, f(a))$. If necessary, we augment that collection of extreme points to a total of $k + 1$, by picking some more of the points $M(a, f(a))$ in an arbitrary fashion. Let $Q(g)$ be the convex hull of these $k + 1$ points. Since $Q(g)$ is a subset of $Q(f)$ and $z \in Q(g)$, it follows that x and z also minimize the distance $\rho(x, z)$ between D and $Q(g)$. Consider the iteration of the algorithm where this particular partial response g was considered in Step (a). The point y calculated by the algorithm during that iteration is then equal to z . It follows from Eq. (20) that

$$(y - x)^\top y \leq (y - x)^\top M(a, f(a)), \quad \forall a \in \mathcal{A}.$$

The termination condition in Step (a-ii) is reached at some point, so that the algorithm returns “yes.” \square

3.4. The case where player P_2 leads

We comment briefly on the Stackelberg variant where P_2 is the leader and P_1 is the follower. In particular, at each stage, the action of P_1 can take into account the action of P_2 , but all other aspects of the repeated game remain the same as before. For this variant, it can be shown that a polyhedron D is excludable if and only if there exists some

$b \in \mathcal{B}$ such that the intersection of D with the convex hull of the points $\{M(a, b) \mid a \in \mathcal{A}\}$ is empty. Checking this condition amounts to checking the feasibility of n systems of linear inequalities, which can be done in polynomial time.

References

- Bertsimas, D., Tsitsiklis, J.N., 1999. Introduction to Linear Optimization. Athena Scientific.
- Blackwell, D., 1956. An analog of the minimax theorem for vector payoffs. *Pacific J. Math.* 6 (1), 1–8.
- Hart, S., Mas-Colell, A., 2001. A general class of adaptive strategies. *J. Econ. Theory* 98, 26–54.
- Kozlov, M., Tarasov, S., Khachian, L., 1979. Polynomial solvability of convex quadratic programming. *Soviet Math. Doklady* 20, 1108–1111.
- Murty, K.G., Kabadi, S.N., 1987. Some NP complete problems in quadratic and nonlinear programming. *Math. Program.* 39, 117–129.