Deciding Maxmin Reachability in Half-Blind Stochastic Games

Edon Kelmendi[1(✉)] and Hugo Gimbert[2]

[1] LaBRI, Bordeaux, France
edon.kelmendi@labri.fr
[2] LaBRI & CNRS, Bordeaux, France
hugo.gimbert@labri.fr

Abstract. Two-player, turn-based, stochastic games with reachability conditions are considered, where the maximizer has no information (he is blind) and is restricted to deterministic strategies whereas the minimizer is perfectly informed. We ask the question of whether the game has maxmin value of 1 in other words we ask whether for all $\epsilon > 0$ there exists a deterministic strategy for the (blind) maximizer such that against all the strategies of the minimizer, it is possible to reach the set of final states with probability larger than $1-\epsilon$. This problem is undecidable in general, but we define a class of games, called leaktight half-blind games where the problem becomes decidable. We also show that mixed strategies in general are stronger for both players and that optimal strategies for the minimizer might require infinite-memory.

1 Introduction

Two-player stochastic games are a natural framework for modeling and verification in the presence of uncertainty, where the problem of control is reduced to the problem of optimal strategy synthesis [9]. There is a variety of two-player stochastic games that have been studied, depending on the information available to the players (perfect information or partial information), the winning objective (safety, reachability, etc.), the winning condition (surely, almost-surely, or limit-surely winning; probability higher than some quantity), whether the players choose actions concurrently or whether they take turns. Stochastic games with partial observation are particularly well suited for modeling many scenarios occurring in practice; normally we do not know the exact state of the system we are trying to model, e.g. we are aided by noisy sensors or by a software interface that provides only a partial picture. Unfortunately, compared to perfect information games, algorithmic problems on partial information games are substantially harder and often undecidable [3,13,16]. Assuming one player to be perfectly informed while the other player is partially informed (semiperfect-information games [4,5]) brings some relief to the computational hardness as opposed to general partial information games.

This work was partially supported by the French ANR project “Stoch-MC” and “LaBEX CPU” of Université de Bordeaux.

M. Gairing and R. Savani (Eds.): SAGT 2016, LNCS 9928, pp. 52–63, 2016.
DOI: 10.1007/978-3-662-53354-3_5

In the present paper we consider half-blind stochastic games: one player has no information (he is blind) and plays deterministically while the other player is perfectly informed. We study half-blind games for the reachability objective and maxmin winning condition: we want to decide if for every $\epsilon > 0$ there exists a deterministic strategy for the maximizer such that against all strategies of the minimizer, the final states are reached with probability at least $1 - \epsilon$.

The maxmin condition for half-blind games is a generalization of the value 1 problem for probabilistic finite automata [17]. Most decision problems on probabilistic finite automata are undecidable, notably language emptiness [1,13,16], and the value 1 problem [13]. Consequently, stochastic games with partial information and quantitative winning conditions (the probability of fulfilling the winning objective is larger than some quantity) are undecidable. Nevertheless recently there has been some effort on characterizing decidable classes of probabilistic automata [2,6,10,12,13], with the leaktight class [12] subsuming the others [11].

The interest of this model is twofold. First, it can be considered as a probabilistic finite automaton where the transition probabilities are not fixed but controlled by an adversary with some constraints. In this sense, it is a more robust notion of a probabilistic automaton. Second interest is the study how much more difficult a problem becomes when another player is added, in this case the problem of limit-sure reachability.

Our Results. In the present paper we show that a subclass of half-blind games called leaktight games have a decidable maxmin reachability problem. The game is abstracted through a finite algebraic structure called the belief monoid. This is an extension to the Markov monoid used in [12]. Indeed the elements of the belief monoid are sets of elements of the Markov monoid, and they contain information on the outcome of the game when one strategy choice is fixed. The algorithm builds the belief monoid and searches for particular elements which are witnesses that the set of final states is maxmin reachable. The proof of the correctness of the algorithm uses k-decomposition tree, a data structure used in [8] that is related to Simon's factorization forests. The k-decomposition trees are used to prove lower and upper bounds on certain outcomes of the game and show that it behaves as predicted by the belief monoid.

Comparison with Previous Work. The proof methods extends those developed in [12] in three aspects. First, we define a new monoid structure on top of the Markov monoid structure introduced in [12]. Second, we rely on the extension of Simon's factorization forest theorem [18] to k-factorization trees instead of 2-factorization trees in [12] in order to derive upper and lower bound on the actual probabilities abstracted by the belief monoid. Third, we rely on the leaktight hypothesis to prove both completeness and soundness, while in the case of probabilistic automata the soundness of the abstraction by the Markov monoid was for free.

Outline of the Paper. We start by fixing some notions and notation in Sect. 2 as well as providing a couple of examples. In Sect. 3 we introduce the belief monoid algorithm and the Markov and belief monoids themselves. In Sect. 4 the class of leaktight games is defined using the notion of a leak. The correctness of the algorithm is sketched in Sect. 5, and finally we discuss the power of different types of strategies in Sect. 6 and conclude. The details and proofs can be found in [15].

2 Half-Blind Games and the Maxmin Reachability Problem

Given a set X, we denote by $\Delta(X)$ the set of distributions on X, i.e. functions $f : X \to [0,1]$ such that $\sum_{x \in X} f(x) = 1$.

A *half-blind* game is a two-player, zero-sum, stochastic, turn-based game, played on a finite bipartite graph, where the maximizer has no information, whereas the minimizer has perfect information. Formally a game G is given by the tuple $G = (\mathbf{S_1}, \mathbf{S_2}, \mathbf{A_1}, \mathbf{A_2}, p, F)$. The finite set $\mathbf{S_i}$ is the states controlled by Player i, the finite set $\mathbf{A_i}$ is the actions available to Player i ($i = 1, 2$). Player 1 is the maximizer and Player 2 is the minimizer. The function p mapping $(\mathbf{S_1}, \mathbf{A_1})$ to $\Delta(\mathbf{S_2})$ and $(\mathbf{S_2}, \mathbf{A_2})$ to $\Delta(\mathbf{S_1})$ gives the dynamics of the game. The sets $\mathbf{S_1}, \mathbf{S_2}$ and $\mathbf{A_1}, \mathbf{A_2}$ are disjoint, i.e. $\mathbf{S_1} \cap \mathbf{S_2} = \emptyset$ and $\mathbf{A_1} \cap \mathbf{A_2} = \emptyset$. The set $F \subseteq \mathbf{S_1}$ is the set of final states.

A play of such a game takes place in turns. Initially the game is in some state $s_1 \in \mathbf{S_1}$, then the maximizer (a.k.a. player 1) chooses some action $a_1 \in \mathbf{A_1}$ which moves the game to some state $t_1 \in \mathbf{S_2}$ selected randomly according to the lottery $p(s_1, a_1)$. It is up to the minimizer (a.k.a. player 2) now to choose some action $b_1 \in \mathbf{A_2}$ which moves the game to some state $s_2 \in S_1$. Then again maximizer chooses some action $a_2 \in \mathbf{A_1}$ and so on, until the maximizer decides to stop, at which point, if the game is in a state that belongs to the set of final states F, the maximizer wins, otherwise it is the minimizer who wins. The maximizer is totally blind and does not know what happens, he does not know in which state the game is nor the actions played by the minimizer. Moreover the maximizer plays in a deterministic way, he is not allowed to use a random generator to select his actions. As a consequence, the decisions of maximizer only depend on the time elapsed and can be represented as words on $\mathbf{A_1}$. On the other hand, the minimizer has full information and is allowed to plays actions selected randomly.

Formally, the set of strategies for the maximizer is denoted by Σ_1 they consist of finite words, i.e. $\Sigma_1 = \mathbf{A_1}^*$. In order to emphasize that the strategies of the maximizer are words, elements of Σ_1 are usually denoted by w.

The minimizer's strategies are functions from $\mathcal{H} = (\mathbf{S_1 A_1 S_2 A_2})^* \mathbf{S_1}$ to $\Delta(\mathbf{A_2})$. Let Σ_2 be the set of such strategies. Its elements are typically denoted by τ.

Fixing strategies $w \in \Sigma_1$ of length n, $\tau \in \Sigma_2$ and an initial state $s \in \mathbf{S_1}$ gives a probability measure on the set $\mathcal{H}_n = (\mathbf{S_1 A_1 S_2 A_2})^n \mathbf{S_1}$ which is denoted by $\mathbb{P}_s^{w,\tau}$: for a history $h = s_1 a_1 t_1 b_1 \cdots s_n a_n t_n b_n s_{n+1} \in \mathcal{H}_n$,

$$\mathbb{P}_s^{w,\tau}(h) = \prod_{i=1}^{n} p(s_i, a_i)(t_i) \cdot \tau(h_i)(b_i) \cdot p(t_i, b_i)(s_{i+1})$$

if $s = s_1$ and $w = a_1 \cdots a_n$, and 0 otherwise, where $h_i = s_1 a_1 t_1 b_1 \cdots s_i a_i t_i$, $1 \leq i \leq n$.

For $t \in \mathbf{S_1}$, we will denote by $\mathbb{P}_s^{w,\tau}(t)$ the chance of ending up in state t after starting from state s and playing the respective strategies, i.e. $\mathbb{P}_s^{w,\tau}(t) = \sum_{ht \in \mathcal{H}} \mathbb{P}_s^{w,\tau}(ht)$. Whereas for a set of states $R \subseteq \mathbf{S_1}$ let $\mathbb{P}_s^{w,\tau}(R) = \sum_{t \in R} \mathbb{P}_s^{w,\tau}(t)$.

2.1 The Maxmin Reachability Problem

Now we can introduce the maxmin reachability and for half-blind games, using the notation and notions just defined. Given a game with initial state $s \in \mathbf{S_1}$ and final states $F \subseteq \mathbf{S_1}$, the maxmin value $\underline{val}(s)$ is defined by

$$\underline{val}(s) = \sup_{w \in \Sigma_1} \inf_{\tau \in \Sigma_2} \mathbb{P}_s^{w,\tau}(F).$$

In case $\underline{val}(s) = 1$, we say that F is maxmin reachable from s.

Problem 1 (Maxmin reachability). Given a game, is the set of final states F maxmin reachable from the initial state s?

There is no hope to decide this problem in general. The reason is that in the special case where the minimizer has no choice in any of the states that she controls, then Problem 1 is equivalent to the value one problem for *probabilistic finite automata* which is already known to be undecidable [13]. However, in the present paper, we establish that Problem 1 is decidable for a subclass of half-blind games called leaktight games.

2.2 Deterministic Strategies for the Minimizer

In general, strategies of the minimizer are functions from $\mathcal{H} = (\mathbf{S_1 A_1 S_2 A_2})^* \mathbf{S_1}$ to $\Delta(\mathbf{A_2})$. However, because in the present paper we focus on the maxmin reachability problem, we can assume that strategies of the minimizer have a much simpler form: the choice of action by the minimizer is deterministic and only depends on the current state and on how much time has elapsed since the beginning of the play. Formally, we assume that minimizer strategies are functions $\mathbb{N} \to (\mathbf{S_2} \to \mathbf{A_2})$. Denote Σ_2^p the set of all such strategies. This restriction of the set of minimizer strategies does not change the answer to the maxmin reachability problem (Theorem 1, [15]).

2.3 Two Examples

The graph on which a half-blind game is played is visualized as in Figs. 1 and 2. The circle states are controlled by the maximizer, and the square states are controlled by the minimizer, so for the example in Fig. 1, $\mathbf{S_1} = \{i, f\}$ and $\mathbf{S_2} = \{1, 2\}$. We represent only edges (s,t) such that $p(s,a)(t) > 0$ for some action a and we label the edge (s,t) by a if $p(s,a)(t) = 1$ and by $(a, p(s,a,t))$ otherwise.

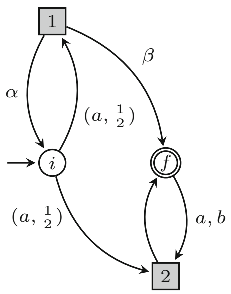

Fig. 1. A half-blind game with $\underline{val}(i) = 1$.

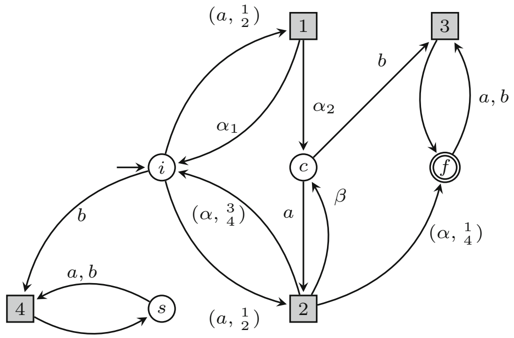

Fig. 2. A half-blind game with $\underline{val}(i) < 1$.

For the game in Fig. 1 it is easy to see that $\underline{val}(i) = 1$, since if the maximizer plays the strategy a^n, no matter what strategy the minimizer chooses the probability to be on the final state is at least $1 - \frac{1}{2^n}$. On the other hand in the game depicted on Fig. 2, $\{f\}$ is not maxmin reachable from i. If the maximizer plays a strategy of only a's then the minimizer always plays the action β and α_1 for example and the probability to be in the final state will be 0. Therefore the maximizer has to play a b at some point. But then the strategy of the minimizer will be to play β except against the action just before b, against that action the minimizer plays α letting at most 1/4 of the chance to go to the final state, but making sure that the rest of the probability distribution is stuck in the sink state s. Consequently $\underline{val}(s) = 1/4$. We reuse the examples above to illustrate the belief monoid algorithm in the next section.

3 The Belief Monoid Algorithm

We abstract the game using two (finite) monoid structures that are constructed, one on top of the other. Given that the game belongs to the class of leaktight games, the monoids will contain enough information to decide maxmin reachability.

3.1 The Markov Monoid

The Markov monoid is a finite algebraic object that is in fact richer than a monoid; it is a *stabilization* monoid (see [7]). The Markov monoid was used in [12] to decide the value 1 problem for leaktight probabilistic automata on finite words.

Elements of the Markov monoid are $\mathbf{S_1} \times \mathbf{S_1}$ binary matrices. They are typically denoted by capital letters such as U, V, W. The entry that corresponds to the states $s, t \in \mathbf{S_1}$ is denoted by $U(s,t)$. We will make use of the notation $s \xrightarrow{U} t$ in place of $U(s,t) = 1$, when it is helpful.

We define two operations on these matrices: the product and the iteration.

Definition 1. *Given two* $\mathbf{S_1} \times \mathbf{S_1}$ *binary matrices* U, V*, their* product *(denoted* UV*) is defined for all* $s, t \in \mathbf{S_1}$ *as*

$$UV(s,t) = 1 \iff \exists s' \in \mathbf{S_1},\ s \xrightarrow{U} s' \wedge s' \xrightarrow{V} t.$$

Given a $\mathbf{S_1} \times \mathbf{S_1}$ *binary matrix* U *that is idempotent, i.e.* $U^2 = U$*, its* iteration *(denoted* $U^{\#}$*) is defined for all* $s, t \in \mathbf{S_1}$ *as*

$$U^{\#}(s,t) = 1 \iff s \xrightarrow{U} t \text{ and } t \text{ is } U\text{-recurrent.}$$

We say that some state $t \in \mathbf{S_1}$ *is* U*-recurrent, if for all* $t' \in \mathbf{S_1}$*,* $t \xrightarrow{U} t' \implies t' \xrightarrow{U} t$*. Otherwise we say that* t *is* U*-transient.*

For a set X of binary matrices, we denote $\langle X \rangle$ the smallest set of binary matrices containing X and closed under product and iteration. Let $B^{a,\tau}$, $a \in \mathbf{A_1}$, $\tau \in \Sigma_2^p$ be a matrix defined by $s \xrightarrow{B^{a,\tau}} t \iff \mathbb{P}_s^{a,\tau}(t) > 0$, $s, t \in \mathbf{S_1}$. Now the definition of the Markov monoid can be given.

Definition 2 (Markov monoid). *The Markov monoid denoted* $\mathcal{M}$ *is*

$$\mathcal{M} = \langle \{B^{a,\tau} \ \mid \ a \in \mathbf{A_1}, \tau \in \Sigma_2^p\} \cup \{\mathbf{1}\} \rangle\,,$$

where $\mathbf{1}$ *is the unit matrix.*

3.2 The Belief Monoid

Roughly speaking, while the elements of the Markov monoid try to abstract the outcome of the game when both strategies are fixed, the belief monoid tries to abstract the *possible outcomes* of the game when only the strategy of the maximizer is fixed. Hence the elements of the belief monoid are subsets of $\mathcal{M}$, and they are typically denoted by boldfaced lowercase letters such as $\mathbf{u}, \mathbf{v}, \mathbf{w}$.

Given two elements of the belief monoid $\mathbf{u}$ and $\mathbf{v}$, their product is the product of their elements, while the iteration of some idempotent $\mathbf{u}$ is the sub-Markov monoid that is generated by $\mathbf{u}$ minus the elements in $\mathbf{u}$ that are not iterated.

Definition 3. *Given* $\mathbf{u}, \mathbf{v} \subseteq \mathcal{M}$*, their product (denoted* $\mathbf{uv}$*) is defined as*

$$\mathbf{uv} = \{UV \ \mid \ U \in \mathbf{u}, V \in \mathbf{v}\}.$$

Given $\mathbf{u} \subseteq \mathcal{M}$ *that is idempotent, i.e.* $\mathbf{u}^2 = \mathbf{u}$*, its iteration (denoted* $\mathbf{u}^{\#}$*) is defined as*

$$\mathbf{u}^{\#} = \left\langle \left\{ UE^{\#}V \mid U, E, V \in \mathbf{u}, EE = E \right\} \right\rangle \ .$$

Given $a \in \mathbf{A_1}$, let $\mathbf{a} = \{B^{a,\tau} \mid \tau \in \Sigma_2^p\}$, we give a definition of the belief monoid.

Definition 4 (Belief Monoid). *The belief monoid, denoted* $\mathcal{B}$*, is the smallest subset of* $2^{\mathcal{M}}$ *that is closed under product and iteration and contains* $\{\mathbf{a} \mid a \in \mathbf{A_1}\} \cup \{\{\mathbf{1}\}\}$*, where* $\mathbf{1}$ *is the unit matrix.*

For the proofs we use *extended* versions of the monoids denoted $\widetilde{\mathcal{M}}, \widetilde{\mathcal{B}}$, where the edges that are deleted by the iteration operation are saved for book-keeping, the definitions can be found in the long version of the paper [15].

We are interested in a particular kind of elements in the belief monoid, called *reachability witnesses.*

Definition 5 (Reachability Witness). *An element* $\mathbf{u} \in \mathcal{B}$ *is called a reachability witness if for all* $U \in \mathbf{u}$, $s \xrightarrow{U} t \implies t \in F$, *where* s *is the initial state of the game and* F *is the set of final states.*

We give an informal description of the way that the belief monoid abstracts the outcomes of the game. Roughly speaking the strategy choice of the maximizer corresponds to choosing an element $\mathbf{u} \in \mathcal{B}$ while the strategy choice of the minimizer corresponds to picking some $U \in \mathbf{u}$. Consequently under those strategy choices, U will tell us the outcome of the game, that is to say if for some $s, t \in \mathbf{S_1}$, if we have $s \xrightarrow{U} t$ then there is some positive probability (larger than a uniform bound) of going from the state s to the state t. In case of $s \overset{U}{\not\to} t$ we will be ensured that the probability of reaching the state t from s can be made arbitrarily small. Therefore if a reachability witness is found then we will know that for any strategy that the minimizer picks the probability of going to some non-final state from the initial state can be made to be arbitrarily small.

3.3 The Belief Monoid Algorithm

The belief monoid associated with a given game is computed by the belief monoid algorithm, see Algorithm 1. We will see later that under some conditions, the belief monoid algorithm decides the maxmin reachability problem.

Algorithm 1. The belief monoid algorithm.

Data: A leaktight half-blind game.
Result: Answer to the Maxmin reachability problem.
$\mathcal{B} \leftarrow \{\mathbf{a} \mid a \in \mathbf{A_1}\}$.
Close $\mathcal{B}$ by product and iteration
Return true iff there is a reachability witness in $\mathcal{B}$

We illustrate the computation of the belief monoid with an example. Consider the game represented on Fig. 2. The minimizer has four pure stationary strategies $\tau_{\alpha_1\alpha}$, mapping 1 to α_1 and 2 to α, and similarly the strategies $\tau_{\alpha_1\beta}, \tau_{\alpha_2\alpha}, \tau_{\alpha_2\beta}$. Now we compute $B^{a,\tau}$ where τ is one of the strategies above. Assume that we have the following order on the states: $i < c < s < f$, then $B^{a,\tau_{\alpha_1\alpha}} = \begin{bmatrix} 1&0&0&1 \\ 1&0&0&1 \\ 0&0&1&0 \\ 0&0&0&1 \end{bmatrix}$, $B^{a,\tau_{\alpha_1\beta}} = \begin{bmatrix} 1&1&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&0&0&1 \end{bmatrix}$, $B^{a,\tau_{\alpha_2\alpha}} = \begin{bmatrix} 1&1&0&1 \\ 1&0&0&1 \\ 0&0&1&0 \\ 0&0&0&1 \end{bmatrix}$, and $B^{a,\tau_{\alpha_2\beta}} = \begin{bmatrix} 0&1&0&0 \\ 0&1&0&1 \\ 0&0&1&0 \\ 0&0&0&1 \end{bmatrix}$. The set that contains these matrices is the set $\mathbf{a}$. We can verify that $\mathbf{a}$ is not idempotent but the set $\mathbf{a}^2$ on the other hand is closed under taking products, i.e. $\mathbf{a}^4 = \mathbf{a}^2$. Therefore we can take it's iteration and compute the element $(\mathbf{a}^2)^\#$. The reader can verify that $(\mathbf{a}^2)^\#$ contains $(B^{a,\tau_{\alpha_1\alpha}})^\# = \begin{bmatrix} 0&0&0&1\\ 0&0&0&1\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}$, $(B^{a,\tau_{\alpha_1\beta}})^\# = \begin{bmatrix} 0&1&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}$, and $B^{a,\tau_{\alpha_2\beta}}$. But it also contains $(B^{a,\tau_{\alpha_1\beta}})^\# B^{a,\tau_{\alpha_1\alpha}} = B^{a,\tau_{\alpha_1\alpha}}$. Therefore $(\mathbf{a}^2)^\#\mathbf{b}$ is not a reachability witness because if we pick $A = B^{a,\tau_{\alpha_1\alpha}}$ in $(\mathbf{a}^2)^\#$ and some $B \in \mathbf{b}$, we will have $i \xrightarrow{AB} s$, and s is a sink state.

This roughly tells us that maximizer cannot win with the strategies $\big((a^{2n}b)\big)_n$, because against $a^{2n}b$ the minimizer plays the strategy $\tau_{\alpha_1\beta}$ for the first $2n-1$ turns and then plays the strategy $\tau_{\alpha_1\alpha}$ against the last a, making sure that after the b is played the we end up in the sink state s with at least $3/4$ probability. Continuing the computation we can verify that the belief monoid of the game in Fig. 2 does not contain a reachability witness.

4 Leaks

Leaks were first introduced in [12] to define a decidable class of instances for the value 1 problem for probabilistic automata on finite words. The decidable class of *leaktight automata* is general enough to encompass all known decidable classes for the value 1 problem [11] and is optimal in some sense [10]. We extend the notion of leak from probabilistic automata to half-blind games and prove that when a game does not contain any leak then the belief monoid algorithm decides the maxmin reachability problem.

Intuitively a leak happens when there is some communication between two recurrence classes with transitions that have a small probability of occurring. Whether this small probability builds up to render one of the recurrence classes transient is a computationally hard question to answer — and in fact impossible in general. Examples of leaks can be found in [11] and the link between leaks and convergence rates are discussed further in [10].

Definition 6 (Leaks). *An element of the extended Markov monoid* $(U, \widetilde{U}) \in \widetilde{\mathcal{M}}$ *is a leak if it is idempotent and there exist* $r, r' \in \mathbf{S_1}$*, such that: (1)* r, r' *are* U*-recurrent, (2)* $r \overset{U}{\not\to} r'$ *and (3)* $r \xrightarrow{\widetilde{U}} r'$*.*

An element of the extended belief monoid $\mathbf{u} \in \widetilde{\mathcal{B}}$ *is a leak if it contains* $(U, \widetilde{U}) \in \mathbf{u}$ *such that* $(U, \widetilde{U})$ *is a leak.*

A game is leaktight if its extended belief monoid does not contain any leaks.

Note also that the question of whether a game is leaktight is decidable, since this information can be found in the belief monoid itself.

5 Correctness of the Belief Monoid Algorithm

This section is dedicated to proving that when the game is leaktight the belief monoid algorithm is both sound (a reachability witness is found implies $\underline{val}(s) = 1$) and complete (no reachability witness is found implies $\underline{val}(s) < 1$).

Theorem 1. *The belief monoid algorithm solves the maxmin reachability problem for half-blind leaktight games.*

Theorem 1 is a direct consequence of Theorems 2 and 3 which are given in the next two sections.

5.1 Soundness

In this section we give the main ideas to prove soundness of the belied monoid algorithm.

Theorem 2 (Soundness). *Assume that the game is leaktight and that its extended belief monoid contains a reachability witness. Then the set of final states is maxmin reachable from the initial state.*

Theorem 2 is justifying the *yes* instances of the belief monoid algorithm, i.e. if the algorithm replies yes, then indeed $\underline{val}(s) = 1$. It is interesting to note that the equivalent soundness theorem for probabilistic automata in [12] does not make use of the leaktight hypothesis. Theorem 2 follows as a corollary of:

Lemma 1. *Given a game whose extended belief monoid is leaktight, with every element $\mathbf{u} \in \mathcal{B}$ of its belief monoid we can associate a sequence $(u_n)_n$, $u_n \in \Sigma_1$ such that for all $(\tau_n)_n$, $\tau_n \in \Sigma_2^p$ there exists $U \in \mathbf{u}$ and a subsequence $((u'_n, \tau'_n))_n \subset ((u_n, \tau_n))_n$ for which*

$$U(s,t) = 0 \implies \lim_n \mathbb{P}_s^{u'_n, \tau'_n}(t) = 0,$$

for all $s, t \in \mathbf{S_1}$.

We can prove Theorem 2 as follows. We are given a game that is leaktight and has a reachability witness $\mathbf{u} \in \mathcal{B}$, to whom we can associate a sequence of words $(u_n)_n$ according to Lemma 1. If on the contrary there exists $\epsilon > 0$ such that $\underline{val}(s) \leq 1 - \epsilon$ then there exists a sequence of strategies $(\tau_n)_n$ such that for all $n \in \mathbb{N}$, $\mathbb{P}_s^{u_n, \tau_n}(F) \leq 1 - \epsilon'$, for some $\epsilon' > 0$. This contradicts Lemma 1 because for the reachability witness we have by definition that for all $U \in \mathbf{u}$, $U(s,t) = 1$ implies $t \in F$.

We give a short sketch of the main ideas utilized into proving Lemma 1.

To $\mathbf{a} \in \mathcal{B}$, $a \in \mathbf{A_1}$ we associate the constant sequence of words $(a)_n$. To the product of two elements in $\mathcal{B}$ we associate the concatenation of their respective sequences, and to $\mathbf{u}^{\#} \in \mathcal{B}$ the sequence $(u_n^n)_n$ is associated, given that $(u_n)_n$ is coupled with $\mathbf{u}$. Then we consider words whose letters are pairs (a, τ), where $a \in \mathbf{A_1}$ and τ is a strategy that maps $\mathbf{S_2}$ to $\mathbf{A_2}$, i.e. a pure and stationary strategy, and give a morphism from these words to the extended Markov monoid $\widetilde{\mathcal{M}}$. This allows us to construct k-decomposition trees of such words with respect to $\widetilde{\mathcal{M}}$ (see [15]) . Then the k-decomposition trees are used to prove lower and upper bounds on the outcomes of the game under the strategy choices given by the word of pairs. The main idea is that we can construct for longer and longer words, k-decomposition trees for larger and larger k, thereby making sure that the iteration nodes have a large enough number of children which enables us to show that the probability of being in transient states is bounded above by a quantity that vanishes in the limit.

5.2 Completeness

Before introducing the main theorem of this section let us give a definition.

Definition 7 (μ-faithful abstraction). *Let $u \in \Sigma_1$ be a word, and $\mu > 0$ a strictly positive real number. We say that $\mathbf{u} \in \widetilde{\mathcal{B}}$ is a μ-faithful abstraction of the word u if for all $(U, \widetilde{U}) \in \mathbf{u}$ there exists $\tau \in \Sigma_2^p$ such that for all $s, t \in \mathbf{S_1}$,*

$$\widetilde{U}(s,t) = 1 \iff \mathbb{P}_s^{u,\tau}(t) > 0 \tag{1}$$

$$U(s,t) = 1 \implies \mathbb{P}_s^{u,\tau}(t) \geq \mu. \tag{2}$$

This section is devoted to giving the main ideas behind the proof of the following theorem.

Theorem 3. *Assume that the game is leaktight. Then there exists $\mu > 0$ such that for all words $u \in \Sigma_1$ there is some element $\mathbf{u} \in \widetilde{\mathcal{B}}$ that is a μ-faithful abstraction of u.*

The notion of μ-faithful abstraction is compatible with product in the following sense.

Lemma 2. *Let $\mathbf{u}, \mathbf{v} \in \widetilde{\mathcal{B}}$ be μ-faithful abstractions of $u \in \Sigma_1$ and $v \in \Sigma_1$ respectively. Then $\mathbf{uv}$ is a μ^2-faithful abstraction of $uv \in \Sigma_1$.*

A naïve use of Lemma 2 shows that any word w has a μ_w-faithful abstraction in $\widetilde{\mathcal{B}}$, where μ_w converges to 0 as the length of w increases. However we need μ_w to depend only on $\widetilde{\mathcal{B}}$, independently of $|w|$. For that we make use of k-decomposition trees. More precisely we build N-decomposition trees for words in Σ_1 where $N = 2^{3 \cdot |\widetilde{\mathcal{M}}|}$. We can construct N-decomposition trees for any word $u \in \Sigma_1$ whose height is at most $3 \cdot |\widetilde{\mathcal{B}}|^2$ and since N is fixed we will be able to propagate the constant μ, it only remains to take care that the constant does not shrink as a function of the number of children in iteration nodes, hence the following lemma.

Lemma 3. *Let $u \in \Sigma_1$ be a word factorized as $u = u_1 \cdots u_n$ where $n > 2^{3 \cdot |\widetilde{\mathcal{M}}|} = N$, and $\mathbf{u} \in \widetilde{\mathcal{B}}$ an idempotent element such that $\mathbf{u}$ is a μ-faithful abstraction of u_i, $1 \leq i \leq n$, for some $\mu > 0$. If $\mathbf{u}$ is not a leak then $\mathbf{u}^{\#}$ is a μ'-faithful abstraction of u, where $\mu' = \mu^{N+1}$.*

Then Theorem 3 is an easy consequence from the lemmata above, which can be shown as follows. We construct a N-decomposition tree for the word $u \in \Sigma_1$, and propagate the lower bound from the leaf nodes, for which we have the bound $\nu > 0$ (where ν is the smallest transition probability appearing in the game) up to the root node. If we know that a bound $\mu > 0$ holds for the children, for the parents we have the following lower bounds as a function of the kind of the node: (1) *product node*: μ^2; (2) *idempotent node* μ^N; (3) *iteration node* μ^{N+1}. Since the length of the tree is at most $h = 3 \cdot |\widetilde{\mathcal{B}}|^2$ we have the lower bound $\mu = \nu^{h(N+1)}$ that holds for all $u \in \Sigma_1$.

6 Complexity of Optimal Strategies

The maxmin reachability problem solved by the belief monoid algorithm concerns games where the maximizer is restricted to pure strategies, and decides whether $\underline{val}(s) = \sup_{w \in \Sigma_1} \inf_{\tau \in \Sigma_2} \mathbb{P}_s^{w,\tau}(F) = 1$, where $\Sigma_1 = \mathbf{A_1}^*$. If we extend further the set Σ_1 of strategies of the maximizer and allow him to have mixed strategies too, then half-blind games have a value [14]. Let $\Sigma_1^m = \Delta(\mathbf{A_1}^*)$ be the set of mixed words then

$$val(s) = \sup_{w \in \Sigma_1^m} \inf_{\tau \in \Sigma_2} \mathbb{P}_s^{w,\tau}(F) = \inf_{\tau \in \Sigma_2} \sup_{w \in \Sigma_1^m} \mathbb{P}_s^{w,\tau}(F).$$

Define Σ_2^f to be the set of finite-memory strategies for the minimizer. These are strategies that are stochastic finite-state probabilistic transducers reading histories and outputting elements of $\Delta(\mathbf{A_2})$, mixed actions.

Let $val^f(s) = \inf_{\tau \in \Sigma_2^f} \sup_{w \in \Sigma_1} \mathbb{P}_s^{w,\tau}(F)$. In general $\underline{val}(s) \leq val(s) \leq val^f(s)$.

A natural question is whether the inequalities above are strict in general, i.e. whether mixed strategies are strictly more powerful for the maximizer and whether infinite-memory strategies are strictly more powerful for the minimizer. The answer to both questions is positive; the relevant examples and further details can be found in [15].

Conclusion

We have defined a class of stochastic games with partial observation where the maxmin-reachability problem is decidable. This holds under the assumption that maximizer is restricted to deterministic strategies. The extension of this result to the value 1 problem where maximizer is allowed to use mixed strategies seems rather challenging.

References

1. Bertoni, A.: The solution of problems relative to probabilistic automata in the frame of the formal languages theory. In: Siefkes, D. (ed.) Gl-4.Jahrestagung: LNCS, vol. 26, pp. 107–112. Springer, Heidelberg (1975)
2. Chadha, R., Sistla, A.P., Viswanathan, M.: Power of randomization in automata on infinite strings. In: Bravetti, M., Zavattaro, G. (eds.) CONCUR 2009. LNCS, vol. 5710, pp. 229–243. Springer, Heidelberg (2009)

3. Chatterjee, K., Doyen, L.: Partial-observation stochastic games: how to win when belief fails. ACM Trans. Comput. Logic (TOCL) **15**(2), 16 (2014)
4. Chatterjee, K., Doyen, L., Nain, S., Vardi, M.Y.: The complexity of partial-observation stochastic parity games with finite-memory strategies. In: Muscholl, A. (ed.) FOSSACS 2014 (ETAPS). LNCS, vol. 8412, pp. 242–257. Springer, Heidelberg (2014)
5. Chatterjee, K., Henzinger, T.A.: Semiperfect-information games. In: Sarukkai, S., Sen, S. (eds.) FSTTCS 2005. LNCS, vol. 3821, pp. 1–18. Springer, Heidelberg (2005)
6. Chatterjee, K., Tracol, M.: Decidable problems for probabilistic automata on infinite words. In: 2012 27th Annual IEEE Symposium on Logic in Computer Science (LICS), vol. 2, pp. 185–194. IEEE (2012)
7. Colcombet, T.: The theory of stabilisation monoids and regular cost functions. In: Albers, S., Marchetti-Spaccamela, A., Matias, Y., Nikoletseas, S., Thomas, W. (eds.) ICALP 2009, Part II. LNCS, vol. 5556, pp. 139–150. Springer, Heidelberg (2009)
8. Colcombet, T.: Regular cost functions, part I: logic and algebra overwords. Logical Methods Comput. Sci. **9**(3), 47 (2013)
9. de Alfaro, L., Henzinger, T.A., Kupferman, O.: Concurrent reachability games. Theor. Comput. Sci. **386**(3), 188–217 (2007)
10. Fijalkow, N.: Profinite techniques for probabilistic automata and the optimality of the markov monoid algorithm. CoRR, abs/1501.02997 (2015)
11. Fijalkow, N., Gimbert, H., Kelmendi, E., Oualhadj, Y: Deciding the value 1 problem for probabilistic leaktight automata. Logical Methods Comput. Sci. 11(2) (2015)
12. Fijalkow, N., Gimbert, H., Oualhadj, Y.: Deciding the value 1 problem for probabilistic leaktight automata. In: LICS, pp. 295–304. IEEE Computer Society (2012)
13. Gimbert, H., Oualhadj, Y.: Probabilistic automata on finite words: decidable and undecidable problems. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6199, pp. 527–538. Springer, Heidelberg (2010)
14. Gimbert, H., Renault, J., Sorin, S., Venel, X., Zielonka, W.: On the values of repeated games with signals. CoRR, abs/1406.4248 (2014)
15. Kelmendi, E., Gimbert, H.: Deciding maxmin reachability in half-blind stochastic games. CoRR, abs/1605.07753 (2016). http://arxiv.org/abs/1605.07753
16. Paz, A.: Some aspects of probabilistic automata. Inf. Control **9**(1), 26–60 (1966)
17. Rabin, M.O.: Probabilistic automata. Inf. Control **6**(3), 230–245 (1963)
18. Simon, I.: On semigroups of matrices over the tropical semiring. ITA **28**(3–4), 277–294 (1994)