

# Stochastic Games with Synchronization Objectives

LAURENT DOYEN, CNRS & LMF, ENS Paris-Saclay, France

We consider two-player stochastic games played on a finite graph for infinitely many rounds. Stochastic games generalize both Markov decision processes (MDP) by adding an adversary player, and two-player deterministic games by adding stochasticity. The outcome of the game is a sequence of distributions over the graph states, representing the evolution of a population consisting of a continuum number of identical copies of a process modeled by the game graph. We consider synchronization objectives, which require the probability mass to accumulate in a set of target states, either always, once, infinitely often, or always after some point in the outcome sequence; and the winning modes of sure winning (if the accumulated probability is equal to 1) and almost-sure winning (if the accumulated probability is arbitrarily close to 1).

We present algorithms to compute the set of winning distributions for each of these synchronization modes, showing that the corresponding decision problem is PSPACE-complete for synchronizing once and infinitely often, and PTIME-complete for synchronizing always and always after some point. These bounds are remarkably in line with the special case of MDPs, while the algorithmic solution and proof technique are considerably more involved, even for deterministic games. This is because those games have a flavour of imperfect information, in particular they are not determined and randomized strategies need to be considered, even if there is no stochastic choice in the game graph. Moreover, in combination with stochasticity in the game graph, finite-memory strategies are not sufficient in general.

CCS Concepts: • **Mathematics of computing** → **Markov processes**; • **Theory of computation** → **Algorithmic game theory**; *Logic and verification*.

Additional Key Words and Phrases: stochastic games, probability distributions, synchronization.

## ACM Reference Format:

Laurent Doyen. 2023. **Stochastic Games with Synchronization Objectives**. *J. ACM* 1, 1 (March 2023), 35 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

Stochastic games are a central model to solve synthesis problems for reactive systems [11, 17], which consist of a nonterminating finite-state program receiving input from an arbitrary, possibly stochastic, environment. The goal of synthesis is to construct a program that satisfies with the largest possible probability a given logical specification regardless of the behaviour of the environment. In this context, the qualitative analysis is to decide if the specification can be satisfied with probability 1, also called almost-surely, and the quantitative analysis is to decide if the specification can be satisfied with probability greater than a given threshold.

Synthesis naturally reduces to solving a two-player stochastic game on a graph, where the logical specification defines the objective of the game as a language of infinite words, representing the set of infinite paths through the graph that are winning for one player. A wealth of results

---

A preliminary version of this paper appeared under the title “Stochastic Games with Synchronizing Objectives” in the *Proceedings of the 37th Annual Symposium on Logic in Computer Science (LICS)*, ACM, 43:1-12, 2022.

Author’s address: Laurent Doyen, [doyen@lsv.fr](mailto:doyen@lsv.fr), CNRS & LMF, ENS Paris-Saclay, Gif-sur-Yvette, France, 91190.

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2023 Association for Computing Machinery.

0004-5411/2023/3-ART \$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

are known for stochastic games with *perfect information*, where the players are fully informed about the state of the game graph [15], such as Martin’s determinacy result and the existence of pure (non-randomized)  $\varepsilon$ -optimal strategies for Borel objectives [34], as well as decidability for  $\omega$ -regular objectives, see Chatterjee and Henzinger’s survey [15] for details and references.

The assumption of perfect information is not realistic in systems consisting of several components where each component has no access to the internal state of the other components. Models of games with *imperfect information* are notoriously more complicated to solve [39], and, combined with the probabilistic and adversarial aspects of stochastic games in general lead to undecidability, even for the simple class of reachability objectives. For instance, distributed games are undecidable, even without stochasticity [38, 40], and partial-observation games are undecidable, even without adversary, for quantitative analysis of finitary objectives [33, 37] and for qualitative analysis of infinitary objectives [4]; randomized strategies are more powerful than pure strategies [14], and determinacy no longer holds [7]. Undecidability and lack of determinacy also occur in stochastic games with objective defined by a language of infinite trees, such as branching-time objectives [10] and hyperproperties [1, 24], where the players can fully observe the current local execution but not the course of branching concurrent executions.

Recent works proposed new decidable models with a flavour of imperfect information, for the control of a large population of identical processes, modeled as a finite-state machine. The global state of the game is a distribution over the local states of the processes, and the specification describes which sequences of distributions are winning. The distributions can be discrete [2, 18] or continuous [3, 32]. The control may be applied uniformly, independently of the local state of each process, as in non-deterministic [6], and probabilistic automata [18], or it may depend on the local history of states, as in Markov decision processes (MDPs) [3, 25] and branching-time specifications [10]. In both cases imperfect information arises: either because the control is uniform (global), thus not aware of the local state of individual processes, or because the control is local, thus not aware of the global states on which the specification is defined.

In this paper, we consider the control problem for a continuous population of processes modeled as a stochastic game with local control, and objective defined by finitary and infinitary synchronization properties [6, 18, 25]. Informally, synchronization happens in a sequence of distributions when (almost) all processes are synchronously in a set of designated target states, that is when, either for  $\varepsilon = 0$ , or for all  $\varepsilon > 0$ , there is a distribution in the sequence where the probability mass in the target states is at least  $1 - \varepsilon$ . We consider finitary synchronization objectives where synchronization should happen once or forever along the sequence of distributions, called respectively *eventually* and *always* synchronizing; and infinitary objectives where synchronization should happen infinitely often or eventually forever, called respectively *weakly* and *strongly* synchronizing [25]. We distinguish the *sure* winning mode for  $\varepsilon = 0$ , and the *almost-sure* winning mode for  $\varepsilon \rightarrow 0$  (where the synchronization objective must be satisfied for all  $\varepsilon > 0$ ).

The most interesting and challenging objectives are eventually and weakly synchronizing, analogous to reachability and Büchi objectives. For those objectives, it is known that finite memory is not sufficient for almost-sure winning, already in MDPs [25], and determinacy does not hold. Therefore, both the construction of a winning strategy (to show that player 1 is almost-sure winning), and the construction of a spoiling strategy for the adversary (to show that player 1 is not almost-sure winning) are non trivial. In particular, the traditional approach of constructing a winning strategy for player 2 for the complement of the objective to obtain a spoiling strategy cannot work. The construction of a spoiling strategy must be carried out after fixing an arbitrary infinite-memory strategy for player 1, which is a substantial complication. This is the main technical challenge to prove the correctness of our algorithm. We show that the control problem for eventually

and weakly synchronizing is PSPACE-complete. For always and strongly synchronizing, a simple reduction to traditional safety and coBüchi stochastic games induces a polynomial-time solution.

*Applications and Related Works.* The main interest of this contribution lies in the combination of adversarial, stochastic, and infinitary aspects with a flavour of imperfect information in a decidable model. The works on (discrete) parameterized control considered finitary synchronization objectives (reachability of a synchronized distribution), either with an adversary [6], or with stochasticity [18], but not with both. With continuous distributions, the central model that has been studied is MDPs, thus with stochasticity but no adversary, either for finitary [3] or infinitary objectives [2, 25]. The solution of the control problem studied in this paper is known for MDPs [25].

Like in all the above previous works, the main limitation of this population model is the absence of communication between the processes. While communication plays a central role in distributed programming applications [27], self-organization and coordinated behaviour can emerge from large crowds of individuals with limited sensing ability, without signaling, and without centralized control [20, 21]. The highly developed local control necessary to achieve a complex collective behaviour may emerge naturally [28] or be engineered [35].

The line of work followed in this paper can also be viewed as an attempt to propose decidable models that are still rich enough to describe interesting natural phenomena. Many systems in natural computing exhibit several instances of the same anonymous process (without pre-defined identity or hierarchy), from particle physics to flock of birds. Examples of biological systems such as yeast [3, 6], and simple chemical systems [32] illustrate the synthesis applications of this model. The same principle underlies synthetic biology where a local control program is executed in every instance of the process [26, 36]. In more complex systems, the computational mechanisms behind local decision-making towards global behaviours have multiple origins that require more sophisticated computational models [21].

## 2 DEFINITIONS

A *probability distribution* on a finite set  $S$  is a function  $d : S \rightarrow [0, 1]$  such that  $\sum_{s \in S} d(s) = 1$ . The *support* of  $d$  is the set  $\text{Supp}(d) = \{s \in S \mid d(s) > 0\}$ . We denote by  $\mathcal{D}(S)$  the set of all probability distributions on  $S$ .

Given a set  $T \subseteq S$ , let  $d(T) = \sum_{s \in T} d(s)$ . For  $T \neq \emptyset$ , the *uniform distribution* on  $T$  assigns probability  $\frac{1}{|T|}$  to every element in  $T$ . Given  $s \in S$ , we denote by  $1_s$  the *Dirac distribution* on  $s$  that assigns probability 1 to  $s$  (which we often identify with  $s$ ).

*Stochastic games.* A *two-player stochastic game* (or simply, a game)  $\mathcal{G} = \langle Q, A, \delta \rangle$  consists of a finite set  $Q$  of states, a finite nonempty set  $A$  of actions, and a probabilistic transition function  $\delta : Q \times A \times A \rightarrow \mathcal{D}(Q)$ . We typically denote by  $n = |Q|$  the size of the state space, and by  $\eta$  the smallest positive probability in the transitions of  $\mathcal{G}$ .

From an initial state  $q_0 \in Q$ , the game is played in (infinitely many) rounds as follows. Each round starts in a state  $q_i \in Q$ , the first round starts in the initial state  $q_0$ . In each round, player 1 chooses an action  $a \in A$ , and then given  $a$ , player 2 chooses an action  $b \in A$ . Given the state  $q_i$  in which the round started, the next round starts in  $q_{i+1}$  with probability  $\delta(q_i, a, b)(q_{i+1})$ . Note that the game is turn-based as player 2 sees the action chosen by player 1 before playing.

A state  $q$  is a *player-1 state* if  $\delta(q, a, b) = \delta(q, a, b')$  for all  $a, b, b' \in A$ , and it is a *player-2 state* if  $\delta(q, a, b) = \delta(q, a', b)$  for all  $a, a', b \in A$ . We write  $\delta(q, a, -)$  or  $\delta(q, -, b)$  to emphasize and recall that  $q$  is a player-1 or player-2 state. In figures, player-1 states are shown as circles, player-2 states as boxes (except in Figure 1 where boxes emphasize the action choices of player 2 within a round). The value of the transition probabilities are not shown on figures, but diamonds represent the

probabilistic choices (the main results of this paper are independent of the exact value of transition probabilities).

Classical special cases of stochastic games include Markov decision processes (MDPs), also called one-player stochastic games, where all states are player-1 states; adversarial MDPs where all states are player-2 states; and deterministic games where  $\delta(q, a, b)$  is a Dirac distribution for all  $q \in Q$  and all  $a, b \in A$ .

Note that it is not important that the action set  $A$  is the same for both players. For example, given two nonempty action sets  $A_1$  and  $A_2$ , we can simulate a transition function  $\delta_{12} : Q \times A_1 \times A_2 \rightarrow \mathcal{D}(Q)$ , by defining  $A = A_1 \cup A_2$  and, considering fixed actions  $a_0 \in A_1$ , and  $b_0 \in A_2$ , by defining  $\delta(q, a, b) = \delta_{12}(q, a', b')$  where  $a' = a$  if  $a \in A_1$ , and  $a' = a_0$  otherwise, and  $b' = b$  if  $b \in A_2$ , and  $b' = b_0$  otherwise.

A *play* in  $\mathcal{G}$  is an infinite sequence  $\pi = q_0 a_0 b_0 q_1 a_1 b_1 q_2 \dots \in (QAA)^\omega$  such that  $\delta(q_i, a_i, b_i)(q_{i+1}) > 0$  for all  $i \geq 0$ . The prefix  $q_0 a_0 b_0 q_1 \dots q_k$  of the play  $\pi$  is denoted by  $\pi(k)$ , its length is  $|\pi(k)| = k$  and its last element is  $\text{Last}(\pi(k)) = q_k$ . The set of all plays in  $\mathcal{G}$  is denoted by  $\text{Play}(\mathcal{G})$ , and the set of corresponding finite prefixes (or histories) is denoted by  $\text{Pref}(\mathcal{G})$ .

*Strategies.* A *strategy* for player 1 in  $\mathcal{G}$  is a function  $\sigma : \text{Pref}(\mathcal{G}) \rightarrow \mathcal{D}(A)$ , and for player 2 it is a function  $\tau : \text{Pref}(\mathcal{G}) \times A \rightarrow \mathcal{D}(A)$ . We denote by  $\Sigma$ , and  $\Theta$ , the sets of all player-1 strategies, and all player-2 strategies, respectively. A strategy  $\sigma$  for player 1 is *pure* if  $\sigma(\rho)$  is a Dirac distribution for all  $\rho \in \text{Pref}(\mathcal{G})$ ; it is *counting* if  $|\rho| = |\rho'|$  and  $\text{Last}(\rho) = \text{Last}(\rho')$  implies  $\sigma(\rho) = \sigma(\rho')$  for all  $\rho, \rho' \in \text{Pref}(\mathcal{G})$ ; and it is *memoryless* if  $\text{Last}(\rho) = \text{Last}(\rho')$  implies  $\sigma(\rho) = \sigma(\rho')$  for all  $\rho, \rho' \in \text{Pref}(\mathcal{G})$ . We view deterministic strategies for player 1 as functions  $\sigma : \text{Pref}(\mathcal{G}) \rightarrow A$ , and counting strategies as functions  $\sigma : \mathbb{N} \times Q \rightarrow \mathcal{D}(A)$ .

A strategy  $\sigma$  (for player 1) uses *finite memory* if there exists a right congruence  $\approx$  of finite index (i.e., that can be generated by a finite-state transducer) over  $\text{Pref}(\mathcal{G})$  such that  $\rho \approx \rho'$  implies  $\sigma(\rho) = \sigma(\rho')$ . We omit analogous definitions of pure, counting, memoryless, and finite-memory strategies for player 2.

*State-based objectives.* The traditional view is to consider the semantics of probabilistic systems as a probability distribution over sequences (of interleaved states and actions), i.e., over plays, where the initial state is drawn from a distribution  $d_0 \in \mathcal{D}(Q)$ .

We denote by  $\text{Pr}_{d_0}^{\sigma, \tau}$  the standard probability measure on the sigma-algebra over the set of (infinite) plays, generated by the cylinder sets spanned by the (finite) prefixes of plays [5]. Given a prefix  $\rho = q_0 a_0 b_0 q_1 \dots q_k$ , the cylinder set  $\text{Cyl}(\rho) = \{\pi \in \text{Play}(\mathcal{G}) \mid \pi(k) = \rho\}$  has probability:

$$\text{Pr}_{d_0}^{\sigma, \tau}(\text{Cyl}(\rho)) = d_0(q_0) \cdot \prod_{i=0}^{k-1} \sigma(\rho(i))(a_i) \cdot \tau(\rho(i), a_i)(b_i) \cdot \delta(q_i, a_i, b_i)(q_{i+1}).$$

We say that  $\rho$  is *compatible* with  $\sigma$  (from  $d_0$ ) if  $\text{Pr}_{d_0}^{\sigma, \tau}(\text{Cyl}(\rho)) > 0$  for some player-2 strategy  $\tau$ .

State-based objectives, in this traditional semantics, are sets of plays. We consider the following state-based objectives, expressed by LTL formulas [5] where  $T \subseteq Q$  is a set of target states: the reachability and safety objectives  $\diamond T$  and  $\square T$ , their bounded variants  $\diamond^{\leq k} T$ ,  $\diamond^{< k} T$ , and  $\square^{\leq k} T$  (where  $k \in \mathbb{N}$ ), and the coBüchi objective  $\diamond \square T$ . As each of the above objectives  $\varphi$  is a measurable set, the probability  $\text{Pr}_{d_0}^{\sigma, \tau}(\varphi)$  that  $\varphi$  is satisfied along a play with initial distribution  $d_0$  and strategies  $\sigma$  for player 1 and  $\tau$  for player 2 is well defined [44]. In particular, we say that player 1 is *almost-sure* winning from an initial distribution  $d_0$  for a state-based objective  $\varphi$  if he has a strategy to win with probability 1, that is  $\exists \sigma \in \Sigma \cdot \forall \tau \in \Theta : \text{Pr}_{d_0}^{\sigma, \tau}(\varphi) = 1$ .

*Distribution-based objectives.* An alternative view is to consider probabilistic systems as generators of sequences of probability distributions (over states) [32]. We denote by  $\mathcal{G}_{d_0}^{\sigma, \tau}$  the *outcome sequence*

$d_0, d_1, \dots$  where  $d_i \in \mathcal{D}(Q)$  is, intuitively, the probability distribution over states after  $i$  rounds defined, for all  $q \in Q$ , by:

$$d_i(q) = \Pr_{d_0}^{\sigma, \tau}(\diamond^{\leq i}\{q\}) = \sum_{\substack{\rho \in \text{Pref}(\mathcal{G}) \\ |\rho|=i \\ \text{Last}(\rho)=q}} \Pr_{d_0}^{\sigma, \tau}(\text{Cyl}(\rho)).$$

For a Dirac distribution  $d_0 = 1_q$ , we often write  $\mathcal{G}_q^{\sigma, \tau}$  instead of  $\mathcal{G}_{1_q}^{\sigma, \tau}$ . We also sometimes omit the subscript  $d_0$  when the initial distribution is clear from the context. Given  $T \subseteq Q$ , we denote by  $\mathcal{G}_{d_0}^{\sigma, \tau}(T)$  the sequence of numbers  $d_0(T), d_1(T), \dots$ .

Distribution-based objectives, in this alternative semantics, are sets of infinite sequences of distributions over states. In particular, given a set  $T \subseteq Q$  of target states, *synchronization objectives* informally require that the probability mass in  $T$  tends to 1 (or is equal to 1) in a sequence  $(d_k)_{k \in \mathbb{N}}$ , in either all, some, infinitely many, or all but finitely many positions [25]. For  $0 \leq \varepsilon \leq 1$ , we say that a sequence  $\vec{d} = d_0 d_1 \dots$  of probability distributions is, always, eventually, weakly, or strongly  $(1 - \varepsilon)$ -synchronizing in  $T$  if  $d_i(T) \geq 1 - \varepsilon$ , respectively, for all  $i \geq 0$ , for some  $i \geq 0$ , for infinitely many  $i$ 's, or for all but finitely many  $i$ 's.

For each synchronization mode  $\lambda \in \{\text{always}, \text{event}, \text{weakly}, \text{strongly}\}$ , we consider *winning modes* that require either that  $\varepsilon$  equals 0 (sure winning mode), or that  $\varepsilon$  tends to 0 (almost-sure winning mode).

We say that player 1 is:

- *sure* winning for a synchronization mode  $\lambda$  in  $T$  from an initial distribution  $d_0$  if he has a strategy to ensure 1-synchronizing in  $T$ , or  $\exists \sigma \in \Sigma \cdot \forall \tau \in \Theta : \mathcal{G}_{d_0}^{\sigma, \tau}$  is 1-synchronizing in  $T$  in mode  $\lambda$ .
- *almost-sure* winning for a synchronization mode  $\lambda$  in  $T$  from an initial distribution  $d_0$  if he has a strategy to ensure  $(1 - \varepsilon)$ -synchronizing in  $T$  for all  $\varepsilon > 0$ , or  $\exists \sigma \in \Sigma \cdot \forall \tau \in \Theta \cdot \forall \varepsilon > 0 : \mathcal{G}_{d_0}^{\sigma, \tau}$  is  $(1 - \varepsilon)$ -synchronizing in  $T$  in mode  $\lambda$ .

We denote by  $\langle\langle 1 \rangle\rangle_{\text{sure}}^\lambda(\mathcal{G}, T)$  (or simply  $\langle\langle 1 \rangle\rangle_{\text{sure}}^\lambda(T)$  when the game  $\mathcal{G}$  is clear from the context) the set of distributions  $d$  from which player 1 is sure winning for synchronization mode  $\lambda$  in  $T$ ; we define analogously the set  $\langle\langle 1 \rangle\rangle_{\text{almost}}^\lambda(\mathcal{G}, T)$ , and we say that player 1 is (sure or almost-sure) winning from  $d$ , or that  $d$  is (sure or almost-sure) winning. If  $d \notin \langle\langle 1 \rangle\rangle_{\text{almost}}^\lambda(\mathcal{G}, T)$ , we say that player 2 can *spoil* player 1 from  $d$  for almost-sure synchronizing in mode  $\lambda$ .

It immediately follows from the definitions that for all  $\lambda \in \{\text{always}, \text{event}, \text{weakly}, \text{strongly}\}$ , and for all  $\mu \in \{\text{sure}, \text{almost}\}$ :

- $\langle\langle 1 \rangle\rangle_\mu^{\text{always}}(T) \subseteq \langle\langle 1 \rangle\rangle_\mu^{\text{strongly}}(T) \subseteq \langle\langle 1 \rangle\rangle_\mu^{\text{weakly}}(T) \subseteq \langle\langle 1 \rangle\rangle_\mu^{\text{event}}(T)$ , and
- $\langle\langle 1 \rangle\rangle_{\text{sure}}^\lambda(T) \subseteq \langle\langle 1 \rangle\rangle_{\text{almost}}^\lambda(T)$ .

In general, these inclusions cannot be strengthened to equality even for MDPs [25], except for always synchronizing where we show that  $\langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{always}}(T) = \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{always}}(T)$  holds in stochastic games (Lemma 11 in Section 4.3).

We are interested in computing the sets  $\langle\langle 1 \rangle\rangle_{\text{sure}}^\lambda(\mathcal{G}, T)$  and  $\langle\langle 1 \rangle\rangle_{\text{almost}}^\lambda(\mathcal{G}, T)$  for the four synchronization modes  $\lambda \in \{\text{always}, \text{event}, \text{weakly}, \text{strongly}\}$ , which we generically call *winning regions*. It is sufficient to have an algorithm that computes the set of states  $q$  such that  $1_q$  is winning: to know if a distribution  $d$  is winning, consider a new state  $q_d$  with stochastic transitions  $\delta(q_d, a, b)(q) = d(q)$  for all  $q \in Q$ , and all  $a, b \in A$ . We consider the *membership problem*, which is to decide, given a game  $\mathcal{G}$ , a set  $T$ , and a state  $q$ , whether  $1_q \in \langle\langle 1 \rangle\rangle_{\text{sure}}^\lambda(\mathcal{G}, T)$  (resp., whether  $1_q \in \langle\langle 1 \rangle\rangle_{\text{almost}}^\lambda(\mathcal{G}, T)$ ).

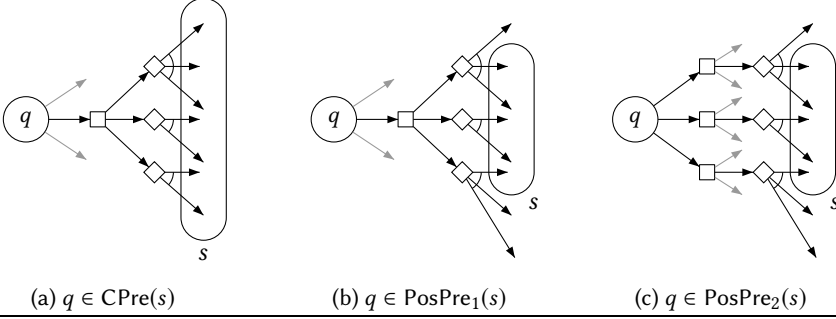


Fig. 1. The predecessor operators.

*Superposition of strategies.* We use the following property in MDPs: given two strategies  $\sigma_1, \sigma_2$  in an MDP  $\mathcal{M}$ , for all  $0 \leq \alpha \leq 1$ , there exists a strategy  $\sigma$  such that  $\mathcal{M}^\sigma = \alpha \cdot \mathcal{M}^{\sigma_1} + (1 - \alpha) \cdot \mathcal{M}^{\sigma_2}$  (for MDPs, we omit the strategy of the second player in all notations). We say that  $\sigma$  is obtained by superposition of  $\sigma_1$  and  $\sigma_2$ , and we denote it by  $\alpha \cdot \sigma_1 + (1 - \alpha) \cdot \sigma_2$ . This notation is slightly misleading, in the sense that in general  $\sigma(\rho)(a) \neq \alpha \cdot \sigma_1(\rho)(a) + (1 - \alpha) \cdot \sigma_2(\rho)(a)$ . In fact,  $\sigma(\rho)(a) = \frac{\alpha \cdot \text{Pr}^{\sigma_1}(\text{Cyl}(\rho)) \cdot \sigma_1(\rho)(a) + (1 - \alpha) \cdot \text{Pr}^{\sigma_2}(\text{Cyl}(\rho)) \cdot \sigma_2(\rho)(a)}{\alpha \cdot \text{Pr}^{\sigma_1}(\text{Cyl}(\rho)) + (1 - \alpha) \cdot \text{Pr}^{\sigma_2}(\text{Cyl}(\rho))}$ . This property will be useful in the (infinite-state) MDPs obtained from games after fixing a strategy of player 1.

*Attractors and subgames.* Let  $\text{CPre} : 2^Q \rightarrow 2^Q$  be the *controllable predecessor operator* defined for all  $s \subseteq Q$  by  $\text{CPre}(s) = \{q \in Q \mid \exists a \in A \cdot \forall b \in A : \text{Supp}(\delta(q, a, b)) \subseteq s\}$ . Intuitively,  $\text{CPre}(s)$  is the set of states from which player 1 can ensure to be in  $s$  after one round, regardless of the action chosen by player 2 and of the outcome of the probabilistic transition (Figure 1a).

For a set  $T \subseteq Q$ , the *attractor*  $\text{Attr}(T, \mathcal{G})$  is the least fixed point of the operator  $x \mapsto \text{CPre}(x) \cup T$ , that is  $\text{Attr}(T, \mathcal{G}) = \bigcup_{i \geq 0} \text{CPre}^i(T)$  (where  $\text{CPre}^0(T) = T$ ). It is the set of states in  $\mathcal{G}$  from which player 1 has a (pure memoryless) strategy to ensure eventually reaching  $T$  [43]. We refer to such a memoryless strategy as an *attractor strategy*.

Let  $\text{PosPre}_1 : 2^Q \rightarrow 2^Q$  be the *positive predecessor operator* for player 1 defined for all  $s \subseteq Q$  by  $\text{PosPre}_1(s) = \{q \in Q \mid \exists a \in A \cdot \forall b \in A : \text{Supp}(\delta(q, a, b)) \cap s \neq \emptyset\}$ , and let  $\text{PosPre}_2 : 2^Q \rightarrow 2^Q$  be defined symmetrically by  $\text{PosPre}_2(s) = \{q \in Q \mid \forall a \in A \cdot \exists b \in A : \text{Supp}(\delta(q, a, b)) \cap s \neq \emptyset\}$  (Figure 1b and Figure 1c).

For a set  $T \subseteq Q$ , the *positive attractor*  $\text{PosAttr}_i(T, \mathcal{G})$  for player  $i$  ( $i = 1, 2$ ) is the least fixed point of the operator  $x \mapsto \text{PosPre}_i(x) \cup T$ . There exists a pure memoryless strategy for player  $i$  (referred to as *positive-attractor strategy*) to ensure, regardless of the strategy for player  $3-i$  that from all states in  $\text{PosAttr}_i(T, \mathcal{G})$ , the set  $T$  is reached within  $n = |Q|$  steps with positive probability (in fact, bounded probability, at least  $\eta^n$  where  $\eta$  is the smallest positive probability in the transitions of  $\mathcal{G}$ ).

A set  $S \subseteq Q$  induces a *subgame* of  $\mathcal{G}$  if for all  $q \in S$ , there exist  $a_q, b_q \in A$  such that  $\delta(q, a_q, b_q)(S) = 1$ , that is, the players can cooperate to keep the play forever in  $S$ . In the subgame, we disallow actions that would leave  $S$  from a state  $q$  by overriding them with  $a_q$  or  $b_q$  (to keep the same alphabet of actions in every state). We denote by  $\mathcal{G} \upharpoonright [S] = \langle S, A, \delta_S \rangle$  the subgame induced by  $S$ , where for all  $q \in S$  and  $a \in A$ , if  $\delta(q, a, b_a)(S) = 1$  for some  $b_a \in A$  (note that this condition holds for  $a = a_q$ ), then for all  $b \in A$  we define  $\delta_S(q, a, b) = \delta(q, a, b)$  if  $\delta(q, a, b)(S) = 1$ , and  $\delta_S(q, a, b) = \delta(q, a, b_a)$  if  $\delta(q, a, b)(S) < 1$ ; otherwise  $\delta(q, a, b)(S) < 1$  for all  $b \in A$ , and then

we define  $\delta_S(q, a, b) = \delta_S(q, a_q, b)$  for all  $b \in A$ . For instance, the set  $S = Q \setminus \text{PosAttr}_i(T, \mathcal{G})$  (for  $i = 1, 2$ ) induces a subgame of  $\mathcal{G}$ .

A set  $S \subseteq Q$  is a *trap* for player 1 in  $\mathcal{G}$  if for all states  $q \in S$  and all actions  $a \in A$ , there exists  $b \in A$  such that  $\delta(q, a, b)(S) = 1$ . Intuitively, player 2 has a strategy to keep player 1 trapped in  $S$ . Dually, the set  $S$  is a *trap* for player 2 in  $\mathcal{G}$  if for all states  $q \in S$ , there exists  $a \in A$  such that for all  $b \in A$  we have  $\delta(q, a, b)(S) = 1$ . Note that  $S$  is a trap for player 2 if and only if  $S \subseteq \text{CPre}(S)$ . A key property of traps is that if, in the subgame  $\mathcal{G} \upharpoonright [S]$  where  $S$  is a trap for player  $i$ , player  $i$  has no winning strategy for some state-based objective from a state  $q$ , then, in  $\mathcal{G}$ , player  $i$  also has no winning strategy for the same objective from  $q$ .

In deterministic games, the operators  $\text{CPre}$  and  $\text{PosPre}_1$  coincide, as well as the attractor  $\text{Attr}(T, \mathcal{G})$  and positive attractor  $\text{PosAttr}_1(T, \mathcal{G})$ , and therefore the set  $U = Q \setminus \text{Attr}(T, \mathcal{G})$  induces a subgame of  $\mathcal{G}$ .

We recall basic properties derived from the definitions of the positive attractor, and from the analysis of stochastic games with almost-sure reachability objective [13, 23]. As before, we denote by  $n$  the size of the state space and by  $\eta$  the smallest positive probability in the transitions.

**LEMMA 1.** *If a distribution  $d_0$  is almost-sure winning for a reachability objective  $\diamond T$  in a game  $\mathcal{G}$ , then there exists a memoryless player-1 strategy  $\sigma$  such that for all  $\varepsilon > 0$ , there exists an integer  $h_\varepsilon$  such that for all player-2 strategies  $\sigma, \tau$ ,  $\Pr_{d_0}^{\sigma, \tau}(\diamond^{\leq h_\varepsilon} T) \geq 1 - \varepsilon$ .*

**PROOF.** Consider the memoryless almost-sure winning strategy for player 1 that plays according to a positive-attractor strategy to reach  $T$  (while staying in the almost-sure winning region). Then within  $n$  rounds, the probability to reach  $T$  is at least  $\eta^n$ . Therefore the probability of not having reached  $T$  is at most  $1 - x$  where  $x = \eta^n > 0$ , and within  $k \cdot n$  rounds it is at most  $(1 - x)^k$ . Since  $1 - x < 1$ , we get  $(1 - x)^{k_\varepsilon} \leq \varepsilon$  by taking  $k_\varepsilon$  sufficiently large. Taking  $h_\varepsilon = k_\varepsilon \cdot n$  concludes the proof.  $\square$

**LEMMA 2.** *If a distribution  $d_0$  is not almost-sure winning for a reachability objective  $\diamond T$  in a game  $\mathcal{G}$ , then there exists a memoryless player-2 strategy  $\tau$  such that for all player-1 strategies  $\sigma$ , for all  $i \geq 0$  we have  $\Pr_{d_0}^{\sigma, \tau}(\diamond^i T) \leq 1 - \eta_0 \cdot \eta^n$  where  $\eta_0 = \min\{d_0(q) \mid q \in \text{Supp}(d_0)\}$  is the smallest positive probability in the initial distribution  $d_0$ .*

In Lemma 2 it is crucial to notice that the bound  $\eta_0 \cdot \eta^n$  is independent of the number  $i$  of steps.

**PROOF.** The set of almost-sure winning states for the reachability objective  $\diamond T$  can be computed by graph-theoretic algorithms [23], and can be expressed succinctly by the  $\mu$ -calculus fixpoint formula  $\varphi_{\text{AS}} = \nu Y. \mu X. T \cup \text{APre}(Y, X)$  [22] where, given  $X, Y \subseteq Q$ :

$$\text{APre}(Y, X) = \{q \in Q \mid \exists a \in A \cdot \forall b \in A : \text{Supp}(\delta(q, a, b)) \subseteq Y \wedge \text{Supp}(\delta(q, a, b)) \cap X \neq \emptyset\}$$

is the set of states from which there is an action to ensure that all successor states are in  $Y$  and that with positive probability the successor state is in  $X$ . Note that  $\text{CPre}(Y) = \text{APre}(Y, Q)$ .

We briefly recall the interpretation of  $\mu$ -calculus formulas [8, 9] based on the Knaster-Tarski theorem. Given a monotonic function  $\psi : 2^Q \rightarrow 2^Q$  (i.e., such that  $X \subseteq Y$  implies  $\psi(X) \subseteq \psi(Y)$ ), the expression  $\nu Y. \psi(Y)$  is the (unique) greatest fixpoint of  $\psi$ , which can be computed as the limit of the sequence  $(Y_i)_{i \in \mathbb{N}}$  defined by  $Y_0 = Q$ , and  $Y_i = \psi(Y_{i-1})$  for all  $i \geq 1$ . Dually, the expression  $\mu X. \psi(X)$  is the (unique) least fixpoint of  $\psi$ , and the limit of the sequence  $(X_i)_{i \in \mathbb{N}}$  defined by  $X_0 = \emptyset$ , and  $X_i = \psi(X_{i-1})$  for all  $i \geq 1$ . If  $|Q| = n$ , then it is not difficult to see that the limit of those sequences is reached after at most  $n$  iterations,  $X_n = X_{n+1}$  and  $Y_n = Y_{n+1}$ .

Intuitively, the formula  $\varphi_{\text{AS}}$  computes the largest set  $S$  of states such that every state  $q \in S$  has a strategy to ensure reaching  $T$  with positive probability, while at the same time staying in  $S$  with

probability 1. It follows that  $S$  is the set of all states from which there exists a strategy to reach  $T$  with probability 1.

Now consider the states  $q_0 \notin \varphi_{AS}$  that are not almost-sure winning, and let the rank of  $q_0$  be the least integer  $i$  such that  $q_0 \in Y_i$  and  $q_0 \notin Y_{i+1}$ . Let  $\Omega_i = Y_i \setminus Y_{i+1}$  for all  $i \geq 0$  be the set of states of rank  $i$ , and for  $\sim \in \{<, \leq\}$  let  $\Omega_{\sim i} = \bigcup_{j \sim i} \Omega_j$ . It is easy to show by induction that  $Q \setminus Y_i = \Omega_{<i}$  since  $\Omega_{<i} = (Y_0 \setminus Y_1) \cup (Y_1 \setminus Y_2) \cup \dots \cup (Y_{i-1} \setminus Y_i)$  and  $Y_0 = Q$ . Moreover since  $T \subseteq Y_i$  we have  $\Omega_i \cap T = \emptyset$ .

For  $q \in \Omega_i$ , we show that there exists a player-2 strategy  $\tau$  such that for all player-1 strategies  $\sigma$  we have  $\Pr_q^{\sigma, \tau}(\Box \Omega_i \cup \Diamond \Omega_{<i}) \geq \eta$ . Since  $q \notin Y_{i+1} = \mu X. T \cup \text{APre}(Y_i, X)$ , we have  $q \notin \text{APre}(Y_i, Y_{i+1})$ . Then for all actions  $a \in A$ , there exists  $b \in A$  such that:

$$\text{either } \text{Supp}(\delta(q, a, b)) \not\subseteq Y_i, \text{ or } \text{Supp}(\delta(q, a, b)) \cap Y_{i+1} = \emptyset,$$

that is

$$\text{either } \text{Supp}(\delta(q, a, b)) \cap \Omega_{<i} \neq \emptyset, \text{ or } \text{Supp}(\delta(q, a, b)) \subseteq \Omega_{\leq i}.$$

Consider the memoryless strategy  $\tau$  for player-2 that, given a state  $q \in \Omega_i$  and action  $a$  of player 1, plays such an action  $b$ .

It follows in both cases that if not all successors of a state  $q_0 \in \Omega_i$  on action  $a$  are in  $\Omega_i$ , then at least one of them is in  $\Omega_{<i}$ , which entails that  $\Pr_{q_0}^{\sigma, \tau}(\Box \Omega_i \cup \Diamond \Omega_{<i}) \geq \eta$  for all player 1 strategies  $\sigma$ . For  $i = 0$ , since  $\Omega_{<0} = \emptyset$ , we have  $\Pr_{q_0}^{\sigma, \tau}(\Box \Omega_0) = 1$  and thus  $\Pr_{q_0}^{\sigma, \tau}(\Box(Q \setminus T)) = 1$  for all  $q_0 \in \Omega_0$  and all strategies  $\sigma \in \Sigma$ . Inductively, if  $\Pr_{q_0}^{\sigma, \tau}(\Box(Q \setminus T)) \geq p_i$  for all  $q_0 \in \Omega_{<i}$  and all  $\sigma \in \Sigma$ , then  $\Pr_{q_0}^{\sigma, \tau}(\Box(Q \setminus T)) \geq \eta \cdot p_i$  for all  $q_0 \in \Omega_i$  and all  $\sigma \in \Sigma$ .

It follows that if  $q_0 \notin \varphi_{AS}$  is not almost-sure winning for the reachability objective  $\Diamond T$ , then  $q_0 \in \Omega_{<n}$  and  $\Pr_{q_0}^{\sigma, \tau}(\Box(Q \setminus T)) \geq \eta^n$  for all  $\sigma \in \Sigma$ . The result of the lemma follows.  $\square$

*Strongly connected component.* In a directed graph  $\langle V, E \rangle$ , a *strongly connected component (SCC)* is a nonempty set  $s \subseteq V$  such that for all  $v, v' \in s$ , there exists a nonempty path from  $v$  to  $v'$ . Our definition excludes singletons  $\{v\}$  to be an SCC if there is no self-loop  $(v, v)$  in  $E$ . The *period* of an SCC is the greatest common divisor of the lengths of all its cycles.

*Overview of the results.* We present the solution of the membership problem for sure winning in Section 3 and for almost-sure winning in Section 4. The most challenging case is almost-sure weakly synchronizing, which we solve first for deterministic games (Section 4.1), and then in the general case of stochastic games (Section 4.2). The other synchronization modes for almost-sure winning are considered in Section 4.3. We also present tight complexity bounds and memory requirement for all winning modes.

### 3 SURE SYNCHRONIZING

In the rest of this paper, when the initial distribution  $d_0$  is irrelevant or clear from the context, we denote the  $i$ -th element  $d_i$  in the sequence  $\mathcal{G}_{d_0}^{\sigma, \tau} = d_0, d_1, \dots$  by  $\mathcal{G}_i^{\sigma, \tau}$ .

For sure winning, only the support of distributions is important (not the exact value of probabilities). Intuitively for player 1, the worst case that can happen is that player 2 uses the *uniform strategy*  $\tau_u$  that plays all actions uniformly at random, in order to scatter the probability mass in as many states as possible, where  $\tau_u(\rho)(a) = \frac{1}{|A|}$  for all  $\rho \in \text{Pref}(\mathcal{G})$  and  $a \in A$ . Formally, given a set  $T \subseteq Q$  and an arbitrary player-1 strategy  $\sigma \in \Sigma$  it is easy to show that  $\text{Supp}(\mathcal{G}_i^{\sigma, \tau}(T)) \subseteq \text{Supp}(\mathcal{G}_i^{\sigma, \tau_u}(T))$ , for all player-2 strategies  $\tau \in \Theta$  and all  $i \geq 0$ .

Therefore, in all synchronization modes, player 1 is sure winning if and only if player 1 is sure winning against the uniform strategy  $\tau_u$  for player 2, and computing the sure winning distributions



in stochastic games reduces to the same problem in MDPs (obtained by fixing  $\tau_u$  in  $\mathcal{G}$ ), which is known [25]. We immediately derive the following results.

**THEOREM 1.** *The membership problem for sure always and strongly synchronizing can be solved in polynomial time, and pure memoryless strategies are sufficient for player 1.*

*The membership problem for sure eventually and weakly synchronizing is PSPACE-complete, and pure strategies with exponential memory are sufficient (and may be necessary) for player 1.*

Note that when the uniform strategy  $\tau_u$  is fixed, the controllable predecessor operator CPre coincides with the predecessor operator used to solve the membership problem for MDPs [25].

Also note that even in the case of deterministic games, the winning regions for sure and almost-sure winning do not coincide, for eventually and weakly synchronizing, as illustrated by the game  $\mathcal{G}_{\text{win}}$  (Figure 2a). The state  $q_1$  is almost-sure winning (as we show in the beginning of Section 4.1), but not sure winning for weakly synchronizing in  $T = \{q_1, q_3\}$  (e.g., against the uniform strategy for player 2). We show in Section 4.3 that in deterministic games, the winning regions for sure and almost-sure winning do coincide for always and strongly synchronizing (even for all stochastic games in the case of always synchronizing).

## 4 ALMOST-SURE SYNCHRONIZING

We first consider almost-sure weakly synchronizing, which is the most interesting and challenging case. We present an algorithm to compute the set  $\langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(T)$  and we show that pure counting strategies are sufficient for player 1.

### 4.1 Weakly synchronizing in deterministic games

The key ideas of the algorithm are easier to present in the special case of deterministic games, and with the assumption that pure counting strategies are sufficient for player 1 (but player 2 is allowed to use an arbitrary strategy). We show in Section 4.2 how to compute the winning region for almost-sure weakly synchronizing in general stochastic games, and without any assumption on the strategies of player 1. It will follow from our results that pure counting strategies are in fact sufficient for player 1.

Given a deterministic game  $\mathcal{G} = \langle Q, A, \delta \rangle$ , a *selector* is a function  $\alpha : Q \rightarrow A$ , and for a set  $s \subseteq Q$ , let  $\delta_\alpha(s) = \{\delta(q, \alpha(q), b) \mid q \in s \wedge b \in A\} \subseteq Q$ . A pure counting strategy can be viewed as an infinite sequence of selectors. The *subset construction* for  $\mathcal{G}$  is the graph  $\mathcal{P}(\mathcal{G}) = \langle V, E \rangle$  where  $V = 2^Q \setminus \{\emptyset\}$  and  $E = \{(s, \delta_\alpha(s)) \mid s \in V \wedge \alpha \text{ is a selector}\}$ . Given a set  $T \subseteq Q$  of target states, and a set  $s \in V$ , we say that  $s$  is *accepting* if  $s \subseteq T$  (for singletons  $\{q\}$  we simply say that  $q$  is accepting).

The *central property* of the subset construction  $\mathcal{P}(\mathcal{G})$  is that for every sequence of selectors  $\alpha_1, \alpha_2, \dots, \alpha_k$ , the sequence  $s_1, s_2, \dots, s_{k+1}$  such that  $s_{i+1} = \delta_{\alpha_i}(s_i)$  for all  $1 \leq i \leq k$  is a path in  $\mathcal{P}(\mathcal{G})$ , and that for every state  $q \in s_{k+1}$ , there exists a play prefix  $q_1 a_1 b_1 q_2 \dots q_{k+1}$  in  $\mathcal{G}$  such that  $q_{k+1} = q$  and  $q_i \in s_i$  for all  $1 \leq i \leq k$ , that is compatible with the given sequence of selectors,  $a_i = \alpha_i(q_i)$ . The central property is easily proved by induction on  $k$  [14]. Using König's Lemma [31], the central property holds for infinite sequences, namely for every infinite path  $s_1, s_2, \dots$  in  $\mathcal{P}(\mathcal{G})$ , there exists an infinite play  $q_1 a_1 b_1 q_2 \dots$  in  $\mathcal{G}$  such that  $q_i \in s_i$  and  $a_i = \alpha_i(q_i)$  for all  $i \geq 1$ . We also mention a simple monotonicity property: if  $s \subseteq s'$ , then  $\delta_\alpha(s) \subseteq \delta_\alpha(s')$  for all selectors  $\alpha$ .

We illustrate the key technical insights with two examples. First consider the deterministic game  $\mathcal{G}_{\text{win}}$  in Figure 2a, where the target set is  $T = \{q_1, q_3\}$ . Only state  $q_2$  has a relevant choice for player 1, and only  $q_1$  has a relevant choice for player 2 (for the sake of clarity, we denote by  $a_1, a_2$  the actions of player 1, and by  $b_1, b_2$  the actions of player 2).

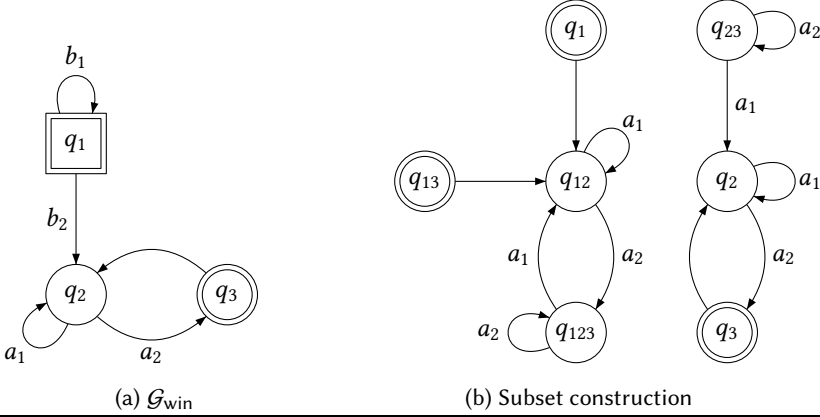


Fig. 2. The deterministic game  $\mathcal{G}_{\text{win}}$  where player 1 is almost-sure weakly synchronizing (from all states), and its subset construction.

Player 1 is almost-sure weakly synchronizing in  $T$  from every state. A winning strategy plays  $a_1$  at even rounds, and  $a_2$  at odd rounds. Note that without the self-loop on  $q_2$ , player 1 is no longer almost-sure weakly synchronizing in  $T$  from  $q_1$  (but still from  $q_2$  and from  $q_3$ ).

Consider the subset construction in Figure 2b, obtained by considering all subsets  $q_{12} = \{q_1, q_2\}$ ,  $q_{13} = \{q_1, q_3\}$ , etc. of  $Q$ , and with an edge from  $s$  to  $s'$  if there exists a selector  $\alpha : Q \rightarrow A$  such that  $s' = \delta_\alpha(s)$ . Intuitively, the selector describes the actions played by a pure counting strategy of player 1 at a given round. Figure 2b labels the edges  $(s, s')$  with the action played in  $q_2$  by the corresponding selector in  $s$  (if relevant, i.e., if  $q_2 \in s$ ). Accepting sets  $s \subseteq T$  are marked by a double line.

An accepting strongly connected component is an SCC containing an accepting set, such as  $C = \{\{q_2\}, \{q_3\}\}$  in our example. This is a witness that player 1 is almost-sure weakly synchronizing in  $T$  from all states  $q$  such that  $C$  is reachable from  $\{q\}$  in  $\mathcal{P}(\mathcal{G})$ . This sufficient condition for almost-sure winning is not necessary, as player 1 is almost-sure winning from  $q_1$  as well, but the set  $\{q_1\}$  cannot reach an accepting SCC in  $\mathcal{P}(\mathcal{G})$ . However, we will show that if there is no accepting SCC in the subset construction, then there is no state from which player 1 is almost-sure weakly synchronizing in  $T$ .

This situation is illustrated in Figure 3 where the subset construction for the game  $\mathcal{G}_{\text{lose}}$  contains no accepting SCC, and player 1 is not almost-sure weakly synchronizing in  $T$  (no matter from which initial state). This is not trivial to see, and we present the crux of the argument below. Although player 1 is not almost-sure weakly synchronizing in  $T$ , it is not true either that player 2 can fix a strategy  $\tau$  in  $\mathcal{G}_{\text{lose}}$  to prevent player 1 from almost-sure winning for weakly synchronizing in  $T$ . This means that in general spoiling strategies can be constructed only after a strategy for the other player has been fixed, which brings technical difficulty in the proofs.

*Why player 1 can spoil player 2 in  $\mathcal{G}_{\text{lose}}$ .* Given an arbitrary strategy  $\tau$  for player 2, we can construct a strategy  $\sigma$  for player 1 such that the outcome sequence  $\mathcal{G}_{\text{lose}}^{\sigma, \tau}$  (from any initial distribution) is almost-sure weakly synchronizing in  $T$ .

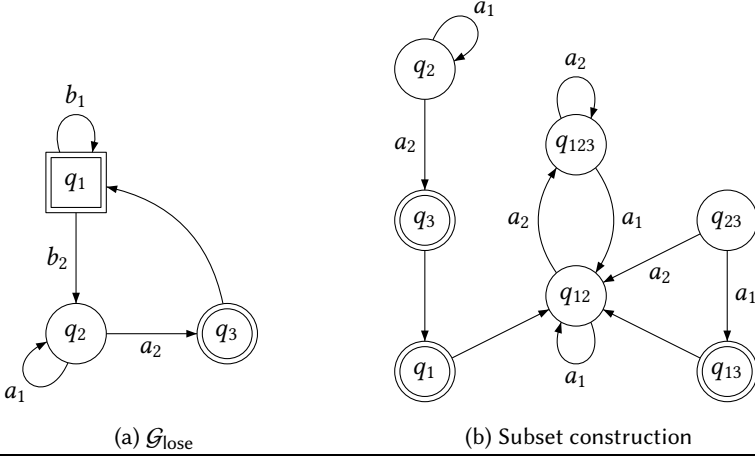


Fig. 3. The deterministic game  $\mathcal{G}_{\text{lose}}$  where player 1 is not almost-sure weakly synchronizing (no matter the initial state), and its subset construction.

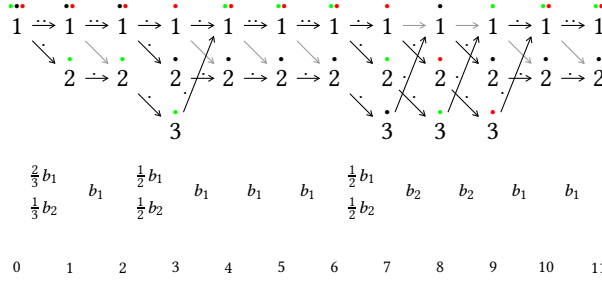


Fig. 4. Construction of a spoiling strategy for player 2 (in  $\mathcal{G}_{\text{lose}}$ ).

Consider the strategy  $\sigma_{\text{loop}}$  that always plays  $a_1$  (to loop through  $q_2$ ), and note that in the outcome  $\mathcal{G}_{\text{lose}}^{\sigma_{\text{loop}}, \tau} = d_0, d_1, \dots$  (from any initial distribution  $d_0$ ), the probability mass in  $q_2$  is non-decreasing, hence  $\lim_{k \rightarrow \infty} d_k(q_2)$  exists, which we denote by  $\alpha(d_0)$ . We construct  $\sigma$  to play as follows, starting with  $\varepsilon = \frac{1}{2}$ : (1) Given  $\varepsilon > 0$  and the current distribution  $d$ , play  $\sigma_{\text{loop}}$  for  $n_\varepsilon$  rounds, where  $n_\varepsilon$  is such that  $d_{n_\varepsilon}(q_2) \geq \alpha(d) - \varepsilon$ , then (2) play  $a_2$  in the next round, and (3) repeat from (1) with  $\varepsilon := \frac{\varepsilon}{2}$ . In the outcome  $\mathcal{G}_{\text{lose}}^{\sigma, \tau}$ , after playing  $a_2$ , the probability mass in  $q_2$  is the probability mass transferred from  $q_1$  in the previous step, which is at most  $\varepsilon$ . It follows that the probability mass in  $T = \{q_1, q_3\}$  is at least  $1 - \varepsilon$ . The repetition of this pattern for  $\varepsilon \rightarrow 0$  entails that  $\mathcal{G}_{\text{lose}}^{\sigma, \tau}$  is almost-sure weakly synchronizing in  $T$ .

*Why player 2 can spoil player 1 in  $\mathcal{G}_{\text{lose}}$ .* We sketch the crux of the argument for initial state  $q_1$ , showing that player 2 can spoil an arbitrary strategy  $\sigma : \mathbb{N} \times Q \rightarrow A$  for player 1 (that is pure and counting), which is an infinite sequence of selectors and thus corresponds to an infinite path from  $\{q_1\}$  in the subset construction (shown in Figure 3b).

Such an infinite path is shown in Figure 4 where an edge  $(s, s')$  labeled by a selector  $\alpha$  is drawn as the set of edges  $(q, q')$  such that  $q \in s$  and  $q' = \delta(q, \alpha(q), b)$  for some  $b \in A$ . Note that, from some point on, all sets in such a path contain a non-target state  $q \in Q \setminus T$ , and a spoiling strategy

for player 2 must ensure a probability mass in  $Q \setminus T$  bounded away from 0, at every round from some point on.

In the example, player 2 can ensure a probability mass of  $\frac{1}{3}$  in state  $q_2 \in Q \setminus T$  from the second round on. In Figure 4, we put three tokens, each carrying a probability mass of  $\frac{1}{3}$ , in the initial state  $q_1$  and we show how the three tokens can move along the edges to always maintain one token in  $q_2$  (after the first round). The choice of which edge from  $q_1$  is taken by a token is made by player 2 with the corresponding action  $b_1$  or  $b_2$  (the corresponding randomized selector is shown below the figure for each round – edges are drawn in gray if no token flows through it). It is easy to show that the pattern suggested in Figure 4 can be prolonged ad infinitum. A key insight is that player 2 should not move a token from  $q_1$  to  $q_2$  at every round (which may cause a depletion of tokens), but only in the rounds where player 1 sends the probability mass from  $q_2$  to  $q_3$ .

Each token follows a play that is compatible with the strategy of player 1. The set of three plays that are followed by the three tokens has the property that in every round after round 1 at least one of the plays is in  $q_2$ . We say that the plays (or the tokens) *cover* the state  $q_2$  from round 2 on. Note that it would be easy to cover  $q_2$  from some round  $n_0$  on by using plays of the form  $(q_1)^n q_2 Q^\omega$  ( $n = n_0, n_0 + 1, \dots$ ), but this is an infinite set of plays, which would require infinitely many tokens and would not allow a positive lower bound on the probability mass of each token. The key to cover  $q_2$  with a finite number of plays is to reuse the tokens when possible. It is however not obvious in general how to construct finitely many such plays, given an arbitrary strategy of player 1.

We show in Lemma 3 that in deterministic games, a fixed number  $K$  of tokens (each representing a probability mass  $\frac{1}{K}$ ) is sufficient for player 2 to spoil any pure counting strategy of player 1, where  $K = 2^{|Q|}$ .

**LEMMA 3.** *The following equivalence holds in deterministic games: there exists a state  $q$  from which player 1 has a pure counting strategy that is almost-sure winning for weakly synchronizing in  $T$  if and only if there exist a strongly connected component  $C$  in  $\mathcal{P}(\mathcal{G})$  and a set  $s \in C$  that is accepting.*

**PROOF.** First, if there exists a strongly connected component  $C$  in  $\mathcal{P}(\mathcal{G})$  and an accepting set  $s \in C$ , then there exists an infinite path  $s_0 s_1 \dots$  in  $\mathcal{P}(\mathcal{G})$  from  $s_0 = s$  that visits  $s$  infinitely often. Consider the corresponding sequence of selectors  $\alpha_0 \alpha_1 \dots$  (such that  $s_{i+1} = \delta_{\alpha_i}(s_i)$ ). It is easy to see that from all states  $q \in s$ , the pure counting strategy  $\sigma$  defined by  $\sigma(i, q) = \alpha_i(q)$  is sure (thus also almost-sure) winning for weakly synchronizing in  $s \subseteq T$ .

For the second direction, we prove the contrapositive. Assume that no SCC in  $\mathcal{P}(\mathcal{G})$  contains an accepting set, and show that from all states  $q_0$  no pure counting strategy for player 1 is almost-sure winning for weakly synchronizing in  $T$ .

Let  $\sigma : \mathbb{N} \rightarrow (Q \rightarrow A)$  be a pure counting strategy for player 1, and we construct a spoiling strategy for player 2 from  $q_0$ . First, consider the sequence  $s_0, s_1, \dots$  defined by  $s_0 = \{q_0\}$  and  $s_{i+1} = \delta_{\sigma(i)}(s_i)$ , and as this sequence is a path in the subset construction  $\mathcal{P}(\mathcal{G})$ , by our assumption there exists an index  $i_0$  such that  $s_i$  is non-accepting for all  $i \geq i_0$ . Let  $\text{Reject} = Q \setminus T$ . We have:

$$s_i \cap \text{Reject} \neq \emptyset \text{ for all } i \geq i_0.$$

By the central property of the subset construction, for every set  $s_i$  and every state  $q_i \in s_i$  (in particular for  $q_i \in s_i \cap \text{Reject}$  if  $i \geq i_0$ ), there exists a play of length  $i$  from  $q_0$  to  $q_i$  that is compatible with the strategy  $\sigma$ . Player 2 can use such plays to inject positive probability into  $q_i$  at every position  $i$ . However, if all those plays form an infinite set (as illustrated by the plays  $(q_1)^n q_2 Q^\omega$  in the example of Figure 3a), then player 2 may not be able to guarantee a probability mass bounded away from 0 in  $q_i$  at every round  $i$  from some point on.

Now we construct a data structure that will help player 2 to determine their strategy and to construct a *finite* set  $\Pi$  of plays compatible with  $\sigma$  in  $\mathcal{G}$  such that, for all  $i \geq i_0$ , there exists a play  $\pi \in \Pi$  with  $\text{Last}(\pi(i)) \in \text{Reject}$ .

The data structure consists, for each position  $i \geq i_0$ , of a tuple  $u_i = \langle r_1, \dots, r_k \rangle$  of registers, each storing a nonempty subset of  $Q$ . The number of registers is not fixed (but will never decrease along the sequence  $u_{i_0}, u_{i_0+1}, \dots$ ), and for  $i = i_0$  let  $u_{i_0} = \langle r_1 \rangle$ , thus  $u_{i_0}$  consists of one register, and let  $r_1 = \{q_{i_0}\}$  where  $q_{i_0} \in s_{i_0} \cap \text{Reject}$ . Each register  $r$  in  $u_i$  corresponds to one token, and stores the possible states in which the token can be after  $i$  steps by following a play compatible with  $\sigma$ .

Given  $u_i = \langle r_1, \dots, r_k \rangle$ , define  $u_{i+1}$  as follows: first, let  $r'_j = \sigma(i)(r_j)$  be the set obtained from  $r_j$  by following the selector  $\sigma(i)$  at position  $i$  (which is the same selector used to define  $s_{i+1}$  from  $s_i$ ). We consider two cases:

- (1) if all registers are accepting, that is  $r'_j \cap \text{Reject} = \emptyset$  for all  $1 \leq j \leq k$ : let  $q_{i+1} \in s_{i+1} \cap \text{Reject}$  and create a new register  $r'_{k+1} = \{q_{i+1}\}$ , define  $u_{i+1} = \underbrace{\langle r'_1, \dots, r'_k, r'_{k+1} \rangle}_{\text{accepting}}$ ;
- (2) otherwise, some register is non-accepting, and let  $j$  be the largest index such that  $r'_j \cap \text{Reject} \neq \emptyset$ . Let  $q_{i+1} \in r'_j \cap \text{Reject}$  and we remove the register at position  $j$ , and replace it by a new register  $r'_{k+1} = \{q_{i+1}\}$  at the end of the tuple. Define  $u_{i+1} = \langle r'_1, \dots, r'_{j-1}, \underbrace{r'_{j+1}, \dots, r'_k, r'_{k+1}}_{\text{accepting}} \rangle$ .

In both cases, we say that the parent of a register  $r'_l$  (for  $l \leq k$ ) that occurs in  $u_{i+1}$  is the register  $r_l$  in  $u_i$ , and that  $r'_{k+1}$  has no parent (in the second case above, we say that  $r'_{k+1}$  is a clone child of  $r_j$ ). An ancestor of a register  $r$  in  $u_i$  is either  $r$  or an ancestor of the parent of  $r$  (but not of the clone parent of  $r$ ). In the sequel, we assume that the registers in a tuple  $u_i$  are called  $r_1, r_2, \dots$  with consecutive indices in the order they appear in  $u_i$ .

We now state key invariant properties of the sequence  $u_{i_0}, u_{i_0+1}, \dots$ , for all  $i \geq i_0$ :

- (*consistency property*) in every  $u_i$ , for every register  $r$  in  $u_i$  we have  $r \subseteq s_i$ ;
- (*singleton property*) in every  $u_i$ , the rightmost register is a singleton containing a non-accepting state;
- (*chain property*) for all  $j \geq i$ , every chain of registers (with order defined by the parent relation) from an ancestor register  $r$  in  $u_i$  to a register  $r'$  in  $u_j$  is a path in the subset construction labeled by the selectors  $\sigma(i), \dots, \sigma(j)$ ;
- (*key property*) in every  $u_i$ , if there are  $k$  registers at the right of a register  $r$ , then  $r$  has at least  $k$  ancestors that are accepting.

It is easy to verify that these properties hold by construction of  $u_{i_0}$ , and of  $u_{i+1}$  from  $u_i$  (by induction). In particular the key property holds because whenever a register is appended (or moved) to the right of a register  $r$ , then  $r$  is accepting.

By our initial assumption (no SCC in  $\mathcal{P}(\mathcal{G})$  contains an accepting set), a register  $r$  cannot have more than  $2^{|Q|}$  ancestors that are accepting (using the chain property). Then it follows from the key property that a tuple  $u_i$  cannot have more than  $2^{|Q|}$  registers, and since the number of registers is not decreasing, there is an index  $i_0^*$  such that all  $u_i$ , for  $i \geq i_0^*$ , contain the same number  $K \leq 2^{|Q|}$  of registers. We are now ready to construct the set  $\Pi$ , containing  $K$  plays.

For each  $i > i_0^*$ , consider the permutation  $f_i$  on  $\{1, \dots, K\}$  that maps index  $j$  to  $k = f_i(j)$  such that  $r_k$  in  $u_{i-1}$  is the parent of  $r_j$  in  $u_i$  (or clone parent if  $r_j$  has no parent).

Following the permutation  $f_i$  at position  $i$  (for  $i > i_0^*$ ), we can define  $K$  equivalence classes  $C_1, \dots, C_K$  of registers that contain exactly one register from each  $u_i$ : the register  $r_j$  in the tuple  $u_i$  belongs to the class  $C_k$  with  $k = (f_{i_0^*+1} \circ f_{i_0^*+2} \circ \dots \circ f_i)(j)$ , see the illustration in Figure 6. Note in

$$\begin{array}{c}
 u_{i-1} = \langle r_1, r_2, r_3, r_4, r_5 \rangle \\
 \uparrow \quad \swarrow \quad \nearrow \quad \nearrow \\
 u_i = \langle r'_1, r'_2, r'_3, r'_4, r'_5 \rangle
 \end{array}$$

Fig. 5. A permutation on registers.

particular that every rightmost register  $r_K$  (highlighted in Figure 6), which is a singleton, belongs to some class.

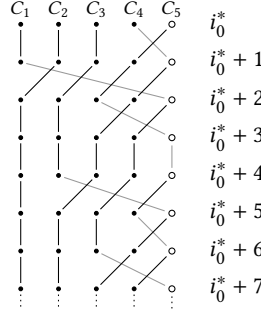


Fig. 6. A sequence of permutations on registers.

Using the central property of the subset construction, from those  $K$  classes we construct  $K$  infinite plays in  $\mathcal{G}$ , all compatible with  $\sigma(i_0^*), \sigma(i_0^* + 1), \dots$ , and such that for all  $i \geq i_0^*$ , for all registers  $r_j$  in  $u_i$ , one of the plays is in a state of  $r_j$  at position  $i$ . In particular, since  $r_K$  is always a non-accepting singleton, the constructed plays cover a non-accepting state at every position  $i \geq i_0^*$ .

Given an equivalence class  $C$ , consider the register  $r$  of  $u_i$  in  $C$ , and the register  $r'$  of  $u_{i+1}$  in  $C$ . Then, by the central property of the subset construction, for all states  $q' \in r'$ , there exists a state  $q \in r$  such that  $q' = \delta(q, \alpha_i(q), b)$  for some action  $b \in A$  of player 2 (no matter whether  $r$  is the parent or clone parent of  $r'$ ). Now consider all *singular* positions  $i$  where  $r_K$  (the rightmost, singleton, register) is the register of  $u_i$  in  $C$ . Between any such two positions  $i_1 < i_2$ , there exists a (segment of) play in  $\mathcal{G}$  from the state in the register  $r_K$  at position  $i_1$  to the state in the register  $r_K$  at position  $i_2$ , that is compatible with the strategy  $\sigma$  (at the corresponding positions  $i_1, i_1 + 1, \dots, i_2$ ) using the chain property. Analogously, from some state in  $s_{i_0^*}$  (using the consistency property) there is a segment of play to the state in register  $r_K$  at the first singular position  $i_0$ . The constructed segments can be concatenated to form a single play, and if this play is finite, we can prolong it to an infinite play compatible with  $\sigma$ .

Given the  $K$  plays in  $\mathcal{G}$  constructed in this way from  $C_1, \dots, C_K$ , it is easy to construct a strategy for player 2 that ensures a probability mass of  $\frac{1}{K}$  (a token) will move along each of the  $K$  plays (possibly using randomization), and therefore a probability mass of at least  $\frac{1}{K}$  in a non-accepting state at every round  $i \geq i_0^*$ , showing that the strategy  $\sigma$  of player 1 is not almost-sure winning for weakly synchronizing in  $T$ , which concludes the proof.  $\square$

In the deterministic game  $\mathcal{G}_{\text{win}}$  of Figure 2a, the strongly connected component  $C = \{\{q_2\}, \{q_3\}\}$  in the subset construction  $\mathcal{P}(\mathcal{G}_{\text{win}})$ , which contains the accepting set  $U = \{q_2\}$ , shows that player 1

is almost-sure winning from some state (Lemma 3), namely from  $q_2$  and from  $q_3$ . This holds even if there is no self-loop on  $q_2$ . However, whether player 1 is almost-sure winning from  $q_1$  depends on the presence of that self-loop: with the self-loop on  $q_2$ , the period of the SCC  $C$  is  $p = 1$  (see Figure 2b) and player 1 is almost-sure winning from  $q_1$ , whereas without the self-loop, the period of  $C$  is  $p = 2$  and player 1 is not almost-sure winning from  $q_1$  (player 2 can inject an equal mass of probability from  $q_1$  to  $q_2$  in two successive rounds, and as those masses can never merge, the probability mass in  $q_2$  is always bounded away from 1). In fact, player 1 needs to ensure that any probability mass injected in  $C$  is always injected at the same round (modulo  $p$ ), where  $p$  is the period of  $C$ .

To track the number of rounds modulo  $p$ , define the game  $\mathcal{G} \times [p]$  that follows the transitions of  $\mathcal{G}$  and decrements a tracking counter (modulo  $p$ ) along each transition (Figure 7). Formally, let  $\mathcal{G} \times [p] = \langle Q', A, \delta' \rangle$  where  $Q' = Q \times \{p-1, \dots, 1, 0\}$  and  $\delta'$  is defined as follows, for all  $\langle q, i \rangle, \langle q', j \rangle \in Q'$  and  $a \in A$ :

$$\delta'(\langle q, i \rangle, a)(\langle q', j \rangle) = \begin{cases} \delta(q, a)(q') & \text{if } j = i - 1 \pmod{p}, \\ 0 & \text{otherwise.} \end{cases}$$

Given  $0 \leq t_0 < p$ , there is a bijection  $\mu_{t_0}$  between the histories in  $\mathcal{G}$  and in  $\mathcal{G} \times [p]$  that maps the history  $q_0 a_0 b_0 q_1 \dots q_k$  in  $\mathcal{G}$  to the history  $(q_0, t_0) a_0 b_0 (q_1, t_1) \dots (q_k, t_k)$  in  $\mathcal{G} \times [p]$  where  $t_i = t_{i-1} - 1 \pmod{p}$  for all  $0 < i \leq k$ . Therefore, when  $t_0$  is fixed, strategies in  $\mathcal{G}$  can be transformed into strategies in  $\mathcal{G} \times [p]$ , via this bijection. In the sequel, we take the freedom to omit mentioning that this bijection needs to be applied, and we consider that strategies can be played both in  $\mathcal{G}$  and in  $\mathcal{G} \times [p]$ . We say that a distribution  $d$  on  $Q \times \{p-1, \dots, 1, 0\}$  is *proper* if there exists an index  $0 \leq j < p$  such that  $\text{Supp}(d) \subseteq Q \times \{j\}$ . We also omit the bijection  $\mu_j$  between proper distributions and distributions on  $Q$  (when the index  $j$  is not relevant, or clear from the context). For a distribution  $d$  in  $Q$ , we denote by  $d \times \{j\}$  the corresponding proper distribution such that  $d \times \{j\}(q, i)$  is equal to  $d(q)$  if  $i = j$  (and 0 otherwise).

A simple property relating the game  $\mathcal{G}$  with the games  $\mathcal{G} \times [p]$  is that player 1 can fix (in advance, regardless of the strategy of player 2) the value of the counter when synchronization occurs in  $T$ .

**LEMMA 4.** *Player 1 is almost-sure weakly synchronizing in  $T$  from  $q$  in the game  $\mathcal{G}$  if and only if for all  $p \geq 0$ , there exists  $0 \leq i \leq p-1$  such that player 1 is almost-sure weakly synchronizing in  $T \times \{0\}$  from  $\langle q, i \rangle$  in  $\mathcal{G} \times [p]$ .*

**PROOF.** It is immediate that almost-sure weakly synchronizing in  $T \times \{0\}$  from  $\langle q, i \rangle$  in the game  $\mathcal{G} \times [p]$  implies almost-sure weakly synchronizing in  $T$  from  $q$  in  $\mathcal{G}$ .

For the converse implication, we prove the contrapositive. If for all  $0 \leq i \leq p-1$ , player 1 is not almost-sure weakly synchronizing in  $T \times \{0\}$  from  $\langle q, i \rangle$  in  $\mathcal{G} \times [p]$ , then for all player-1 strategies  $\sigma$ , there exist player-2 strategies  $\tau_0, \tau_1, \dots, \tau_{p-1}$  such that  $\mathcal{G}_{\langle q, i \rangle}^{\sigma, \tau_i}$  is not almost-sure weakly synchronizing in  $T \times \{0\}$ , that is there exist rounds  $j_0, j_1, \dots, j_{p-1}$  and positive bounds  $\varepsilon_0, \varepsilon_1, \dots, \varepsilon_{p-1}$  such that for all  $0 \leq i \leq p-1$ , for all  $j \geq j_i$  with  $j = i \pmod{p}$ , under strategies  $\sigma, \tau_i$  in  $\mathcal{G}$  from  $q$  the probability mass in  $T$  at position  $j$  is at most  $1 - \varepsilon_i$ . Consider the strategy  $\tau$  playing the superposition  $\sum_i \frac{1}{p} \tau_i$ . For  $\varepsilon = \frac{1}{p} \min_i \varepsilon_i$  and  $j^* = \max_i j_i$ , under strategies  $\sigma, \tau$  in  $\mathcal{G}$  from  $q$  the probability mass in  $T$  at all positions after  $j^*$  is at most  $1 - \varepsilon$ , which shows that  $\sigma$  is not almost-sure weakly synchronizing in  $T$ .  $\square$

Given an accepting set  $U \subseteq T$  that belongs to an SCC with period  $p$  in  $\mathcal{P}(\mathcal{G})$ , we solve the game  $\mathcal{G}$  by removing from  $\mathcal{G} \times [p]$  the attractor  $W$  of  $U \times \{0\}$ , and by recursively solving the subgame of  $\mathcal{G} \times [p]$  induced by  $Q \setminus W$ , with target set  $T \times \{0\}$ .

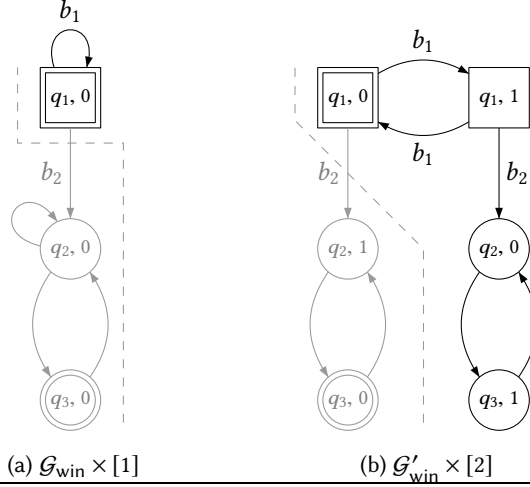


Fig. 7. Removal of positive attractor in  $\mathcal{G}_{\text{win}} \times [1]$  and  $\mathcal{G}'_{\text{win}} \times [2]$ , where  $\mathcal{G}'_{\text{win}}$  is the variant of  $\mathcal{G}_{\text{win}}$  without a self-loop on  $q_2$ .

In  $\mathcal{G}_{\text{win}}$ , the set  $U = \{q_3\}$  belongs to an SCC of period 1, and its attractor is  $W = \{q_2, q_3\}$ . After removal of the attractor, the subgame is winning for player 1 (Figure 7a). In the variant  $\mathcal{G}'_{\text{win}}$  of  $\mathcal{G}_{\text{win}}$  without a self-loop on  $q_2$ , the set  $U = \{q_3\}$  is also in an SCC, but with period 2. After removal of the attractor to  $U \times \{0\}$  in  $\mathcal{G}'_{\text{win}} \times [2]$ , the subgame is not winning for player 1 (Figure 7b).

By Lemma 4, solving  $\mathcal{G}$  with target set  $T$  is equivalent to solving  $\mathcal{G} \times [p]$  with target set  $T \times \{0\}$ . However, in general combining two almost-sure winning strategies constructed in two different subgames may not give an almost-sure winning strategy (if, for instance, one strategy ensures the probability mass in  $T \times \{0\}$  tends to 1 at even rounds, and the other strategy at odd rounds). To establish the correctness of our solution, note that all states in  $U \times \{0\}$  belong to the (controllable predecessor of the) attractor of  $U \times \{0\}$  in  $\mathcal{G} \times [p]$ , and since the period of the SCC containing  $U$  is  $p$ , in  $\mathcal{P}(\mathcal{G} \times [p])$  there is a path from  $U \times \{0\}$  to itself of length  $\ell = k \cdot p$  for all sufficiently large  $k$ . See the Frobenius problem [30] for questions related to computing the largest  $k_0$  such that there is no such path of length  $\ell = k_0 \cdot p$ . It follows that, for  $W = \text{Attr}(U \times \{0\}, \mathcal{G} \times [p])$ , given a state  $\langle q, i \rangle \in W$  and an arbitrary length  $\ell_0 = i + k \cdot p$  for  $k > k_0$ , player 1 has a strategy to get eventually synchronized in  $U \times \{0\} \subseteq T \times \{0\}$  at round  $\ell_0$  (where the tracking counter is 0), and by the same argument at any round  $\ell_1 = \ell_0 + k' \cdot p$  for  $k' > k_0$ , and so on. That is, for any sequence  $i_0, i_1, \dots$  such that  $i_j - i_{j-1} > k_0 \cdot p$  (where  $i_{-1} = 0$ ) and  $i_j = i \pmod p$  for all  $j \geq 0$ , player 1 has a strategy from  $\langle q, i \rangle$  such that for all outcomes  $d_0, d_1, \dots$  of a strategy of player 2 in  $\mathcal{G} \times [p]$ , we have  $\liminf_{k \rightarrow \infty} d_{i_k}(T \times \{0\}) = 1$  (thus also  $\limsup_{k \rightarrow \infty} d_{i_k}(T \times \{0\}) = 1$ ).

Intuitively, we can choose the sequence  $i_0, i_1, \dots$  in order to synchronize the probability mass in  $W$  with the outcome of the strategy constructed in the subgame of  $\mathcal{G} \times [p]$  induced by the complement of  $W$ .

LEMMA 5. Given a set  $U$  in a strongly connected component of period  $p$  in the subset construction  $\mathcal{P}(\mathcal{G})$  that is accepting ( $U \subseteq T$ ), let  $W = \text{Attr}(U \times \{0\}, \mathcal{G} \times [p])$  and  $\mathcal{H} = \mathcal{G} \times [p] \upharpoonright [Q \times [p] \setminus W]$ . We have:

$$\langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{G}, T) \cap \{1_q \mid q \in Q\} = \{1_q \mid \exists i : \langle q, i \rangle \in W \cup \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{H}, T \times \{0\})\}.$$



PROOF. We establish the claim by showing an inclusion in both ways. We recall that we generally identify  $1_s$  with  $s$ . First, we show that if  $q \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{G}, T)$ , then there exists  $0 \leq i \leq p-1$  such that either  $\langle q, i \rangle \in W$ , or  $\langle q, i \rangle \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{H}, T \times \{0\})$ . By Lemma 4, there exists  $0 \leq i \leq p-1$  such that  $\langle q, i \rangle \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{G} \times [p], T \times \{0\})$ , and thus if  $\langle q, i \rangle$  is in the state space of  $\mathcal{H}$ , then since all choices of player 2 in  $\mathcal{H}$  are also possible in  $\mathcal{G} \times [p]$ , we have  $\langle q, i \rangle \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{H}, T \times \{0\})$ , and otherwise  $\langle q, i \rangle \in W$ .

Second, we show that if there exists  $0 \leq i \leq p-1$  such that either  $\langle q, i \rangle \in W$ , or  $\langle q, i \rangle \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{H}, T \times \{0\})$ , then  $q \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{G}, T)$ . To show this, we construct a strategy  $\sigma$  that is almost-sure weakly synchronizing in  $T \times \{0\}$  from  $\langle q, i \rangle$  (in the game  $\mathcal{G} \times [p]$ ), which entails that player 1 is almost-sure weakly synchronizing in  $T$  from  $q$  (in the game  $\mathcal{G}$ ).

The construction proceeds by induction on the subgames. Assume that we have constructed an almost-sure winning strategy  $\sigma_{\mathcal{H}}$  in  $\mathcal{H}$  as well as a sequence of indices  $i_0, i_1, \dots$  such that for all outcomes  $d_0, d_1, \dots$  of a strategy  $\tau$  of player 2 in  $\mathcal{H}$ , we have  $\lim_{k \rightarrow \infty} d_{i_k}(T \times \{0\}) = 1$  (we say that under  $\sigma_{\mathcal{H}}$ , synchronization is guaranteed to occur at indices  $i_0, i_1, \dots$ ). For the induction step, we construct an almost-sure winning strategy  $\sigma_{\mathcal{G}}$  in  $\mathcal{G} \times [p]$  together with a sequence of indices  $j_0, j_1, \dots$  where the synchronization is guaranteed to occur. Note that  $i_k - i_{k-1}$  is a multiple of  $p$  (for all  $k \geq 1$ ) and that the counter  $c$  is 0 in any state  $\langle q', c \rangle$  carrying positive probability at round  $i_k$ . The sequence  $j_0, j_1, \dots$  is a(ny) sub-sequence of  $i_0, i_1, \dots$  such that  $j_k - j_{k-1} \geq k_0 \cdot p$  (where  $j_{-1} = 0$ ). There is a strategy  $\sigma_W$  for player 1 (essentially following paths of the appropriate lengths in  $\mathcal{P}(\mathcal{G} \times [p])$ ) such that given a state  $\langle q', c \rangle$  carrying probability  $\eta$  at round  $j$  with  $j_{k-1} \leq j \leq j_k$ , for all outcomes  $d'_0, d'_1, \dots$  of a strategy  $\tau$  of player 2, we have  $\lim_{m \rightarrow \infty} d'_{j_{k+m}-j}(T \times \{0\}) = 1$ , that is synchronization occurs at rounds  $j_{k+1}, j_{k+2}, \dots$  from  $\langle q', c \rangle$ .

The strategy  $\sigma_{\mathcal{G}}$  plays according to  $\sigma_W$  whenever the state is in  $W$ , and according to the almost-sure winning strategy  $\sigma_{\mathcal{H}}$  otherwise. Note that  $\sigma$  is well-defined because once a play leaves  $\mathcal{H}$  (and enters  $W$ ) it never leaves  $W$ . It is easy to see that  $\sigma$  is almost-sure weakly synchronizing in  $T \times \{0\}$  since, for all strategies  $\tau \in \Theta$  of player 2, the outcome sequence from  $\langle q, i \rangle$  is the sum of a sequence  $d_0, d_1, \dots$  of probability distributions with support in  $W$ , and a sequence  $d'_0, d'_1, \dots$  of probability distributions with support in  $Q \times [p] \setminus W$  and such that, for some  $\eta \in [0, 1]$ :

$$\lim_{k \rightarrow \infty} d_{j_k}(T \times \{0\}) = \eta, \text{ and}$$

$$\lim_{k \rightarrow \infty} d'_{j_k}(T \times \{0\}) = 1 - \eta,$$

from which it follows that  $\limsup_{i \rightarrow \infty} d_i(T \times \{0\}) + d'_i(T \times \{0\}) = 1$ , thus player 1 is almost-sure weakly synchronizing in  $T \times \{0\}$  from  $\langle q, i \rangle$ .  $\square$

Lemma 5 suggests a recursive procedure to compute the almost-sure winning set for a weakly synchronizing objective, shown as Algorithm 1. We illustrate the execution on the example of Figure 8. First the set  $U = \{y\}$  is accepting and belongs to an SCC of period  $p = 2$  (line 2) in the subset construction (line 1). The game  $H = \mathcal{G} \times [p]$  with tracking counter modulo  $p = 2$  is shown in Figure 9, with the attractor  $W$  to  $U \times \{0\} = \{\langle y, 0 \rangle\}$  shaded (lines 3-4). In the induced subgame (line 5), all states except  $\langle x, 0 \rangle$  and  $\langle y, 1 \rangle$  are winning, which is found in the recursive call (line 6). It follows that all states (all Dirac distributions) are winning for player 1, and in fact all distributions that do not contain both  $x$  and  $y$  in their support are winning.

We establish the correctness and termination of Algorithm 1 as follows. The correctness straightforwardly follows from Lemma 5, and we show that the depth of the recursive calls in  $\text{Solve}(\mathcal{G}_0, T)$  is bounded by the size  $|Q_{\mathcal{G}_0}|$  of the state space of  $\mathcal{G}_0$ . This is not immediately obvious, since the size of the first argument  $\mathcal{G}$  in a recursive call may increase (the game  $\mathcal{H}_{\text{sub}}$  is a subgame of  $\mathcal{H}$ , which

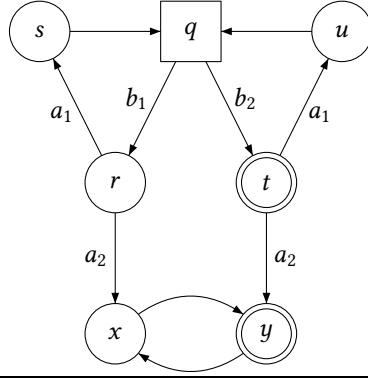
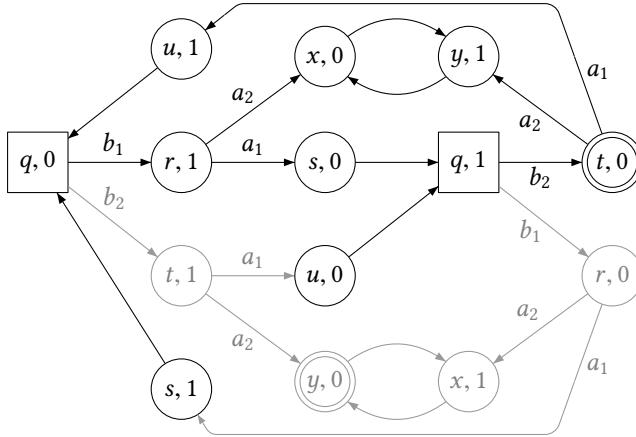


Fig. 8. A deterministic game.

Fig. 9. The game  $\mathcal{H}_{sub}$  computed by Algorithm 1 (line 5) for the game of Fig. 8, where the shaded region is the set  $W$  (line 4), and the set  $U = \{y, 0\}$  is self-recurrent.

is  $p$  times bigger than  $\mathcal{G}$ ). However, we claim that an invariant of the execution of  $Solve(\mathcal{G}_0, T_0)$  is that, in all recursive calls  $Solve(\mathcal{G}, T)$ , the algorithm only needs to consider states of the first argument  $\mathcal{G}$  that form a subgame (isomorphic to a subgame) of  $\mathcal{G}_0 \times [k]$  for some  $k$ . This holds in the initial call (take  $k = 1$ ), and if  $\mathcal{G}$  is a subgame of  $\mathcal{G}_0 \times [k]$ , then the period  $p$  computed at line 2 is a multiple of  $k$ , and therefore in all states  $\langle \langle q, i \rangle, j \rangle$  in  $(\mathcal{G}_0 \times [k]) \times [p]$  the value  $j - i \bmod k$  is constant along the transitions. Given the target states  $(T \times \{0\}) \times \{0\}$  we only need to consider states  $\langle \langle q, i \rangle, j \rangle$  with  $j - i = 0 \bmod k$ , and we can project  $\langle \langle q, i \rangle, j \rangle$  to  $\langle q, j \rangle$  without loss. It follows that  $\mathcal{H}$  (and also  $\mathcal{H}_{sub}$  used in the recursive call) can be viewed as a subgame of  $\mathcal{G}_0 \times [p]$ . Moreover, the attractor  $W$  contains at least one state for every value of the tracking counter, and therefore the size of the game  $\mathcal{G}$  measured as  $\max_i |\{q \in Q_{\mathcal{G}_0} \mid \langle q, i \rangle \in Q_{\mathcal{G}}\}|$  is strictly decreasing. It follows that there are at most  $|Q_{\mathcal{G}_0}|$  recursive calls in  $Solve(\mathcal{G}_0, T)$ .

Given  $0 \leq i \leq p - 1$ , the *slice* at  $i$  of a set  $W \subseteq Q \times [p]$  is the set  $\{q \in Q \mid \langle q, i \rangle \in W\}$ .

**Algorithm 1:** *Solve*( $\mathcal{G}, T$ )

**Input** :  $\mathcal{G} = \langle Q, A, \delta \rangle$  is a deterministic game,  $T \subseteq Q$  is a target set.

**Output** : The set  $\{q \in Q \mid 1_q \in \langle\langle 1 \rangle\rangle_{almost}^{weakly}(\mathcal{G}, T)\}$ .

**begin**

```

1   if there is an SCC  $C$  of  $\mathcal{P}(\mathcal{G})$  containing a set  $U \in C$  with  $U \subseteq T$  then
2      $p \leftarrow$  period of  $C$  ;
3      $\mathcal{H} \leftarrow \mathcal{G} \times [p]$  ;
4      $W \leftarrow \text{Attr}(U \times \{0\}, \mathcal{H})$  ;
5      $\mathcal{H}_{sub} \leftarrow \mathcal{H} \upharpoonright [Q \times [p] \setminus W]$  ;
6     return  $\{q \in Q \mid \exists i : \langle q, i \rangle \in W \cup \text{Solve}(\mathcal{H}_{sub}, T \times \{0\} \setminus W)\}$  ;
  else
7   return  $\emptyset$  ;

```

LEMMA 6. *Algorithm 1 computes the almost-sure winning Dirac distributions for weakly synchronizing in deterministic games. It can be implemented in PSPACE.*

PROOF. The correctness and termination of Algorithm 1 have been established above. To show the PSPACE upper bound, note that if  $\mathcal{G}$  is a subgame of  $\mathcal{G}_0 \times [k]$ , then the the period  $p$  computed at line 2 is at most  $2^n \cdot k$ , because slices are subsets of  $Q_{\mathcal{G}_0}$ . Since the depth of recursive calls is at most  $n$ , the game  $\mathcal{H}$  constructed at line 3 is a subgame of  $\mathcal{G}_0 \times [p]$  where  $p \leq (2^n)^n = 2^{n^2}$ .

In a PSPACE implementation of Algorithm 1, we can store the game  $\mathcal{G}_0$  but not the subgames  $\mathcal{H}_{sub}$  of  $\mathcal{G}_0 \times [k]$  (constructed at line 5). However, we will show that there is a PSPACE procedure to determine the transitions of  $\mathcal{H}_{sub}$ , namely, given  $\langle q, i \rangle, \langle q', j \rangle \in Q \times [k]$  and  $a, b \in A$  to decide whether  $(\langle q, i \rangle, a, b, 1_{\langle q', j \rangle})$  is a transition in  $\mathcal{H}_{sub}$ .

We describe a (N)PSPACE implementation of Algorithm 1 as follows. In the first call to the algorithm, at line 1 we guess the set  $U \subseteq T$ . We can check in PSPACE that  $U$  belongs to an SCC of  $\mathcal{P}(\mathcal{G})$  (by guessing the selectors along a path from  $U$  to itself) and we can compute its period  $p$ . We can construct a PSPACE procedure to check if a state  $\langle q, i \rangle$  belongs to the attractor  $W$  of  $U \times 0$  in  $\mathcal{G} \times [p]$  (analogously, by guessing the selectors along a path from the singleton  $\{\langle q, i \rangle\}$  to  $W$  in  $\mathcal{P}(\mathcal{G})$ ). A transition  $(\langle q, i \rangle, a, b, 1_{\langle q', j \rangle})$  is in  $\mathcal{H}_{sub}$  if  $j = i - 1$  and  $(q, a, b, 1_{q'})$  is a transition in  $\mathcal{G}$  and  $\langle q, i \rangle, \langle q', j \rangle$  are not in the attractor  $W$ . All these conditions can be checked in PSPACE. In the recursive calls (where  $\mathcal{G}$  is  $\mathcal{H}_{sub}$  from caller), we guess a slice of the set  $U$  at line 1, and we check that  $U$  belongs to an SCC of  $\mathcal{P}(\mathcal{G})$  using our PSPACE procedure to decide the transitions in  $\mathcal{H}_{sub}$ . We can do this for all recursive calls up to depth  $n$  by storing the slice of the set  $U$  for each recursive call.  $\square$

THEOREM 2. *The membership problem for almost-sure weakly synchronizing in deterministic games is PSPACE-complete.*

PROOF. The PSPACE upper bound is given by Lemma 6. To establish the PSPACE lower bound, we present a reduction from the membership problem<sup>1</sup> for sure eventually synchronizing in MDPs, which is PSPACE-complete [25, Theorem 2].

Given an MDP  $\mathcal{M}$  and target set  $T$ , construct a deterministic game  $\mathcal{G}$  as a copy of  $\mathcal{M}$  (illustrated in Figure 10) where each probabilistic choice in  $\mathcal{M}$  becomes a choice for player 2 in  $\mathcal{G}$ : for each  $q' \in \delta_{\mathcal{M}}(q, a, -)$ , there is an action  $b \in A$  such that  $\delta_{\mathcal{G}}(q, a, b) = 1_{q'}$  (assuming w.l.o.g. that there

<sup>1</sup>As a side note, we recall that this problem is the same as emptiness of one-letter alternating automata [29].

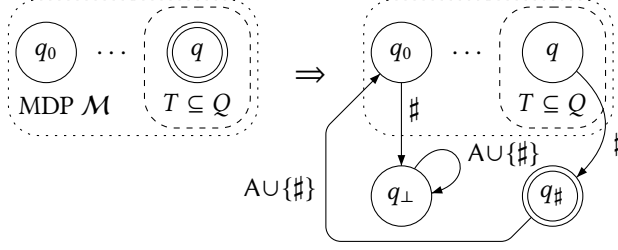


Fig. 10. Sketch of the reduction to show PSPACE-hardness of the membership problem for almost-sure weakly synchronizing.

are sufficiently many actions in  $A$ ). The alphabet of  $\mathcal{G}$  is  $A \cup \{\#\}$  where the action  $\#$  can be used by player 1 from the states in  $T$  to visit the new (target) state  $q_{\#}$ . From the non-target states, playing  $\#$  leads to a sink state.

We claim that player 1 is sure winning for eventually synchronizing in  $T$  from  $q_0$  (in  $\mathcal{M}$ ) if and only if player 1 is almost-sure winning for weakly synchronizing in  $\{q_{\#}\}$  from  $q_0$  (in  $\mathcal{G}$ ).

First, if player 1 has a sure-winning strategy for eventually synchronizing in  $T$  from  $q_0$  (in  $\mathcal{M}$ ), then player 1 can play the same strategy in  $\mathcal{G}$  followed by playing two times  $\#$  to visit  $q_{\#}$  and restart from  $q_0$  repeating the same strategy. Hence player 1 is almost-sure winning for weakly synchronizing in  $\{q_{\#}\}$  from  $q_0$  (in  $\mathcal{G}$ ).

Conversely, if player 1 is almost-sure winning for weakly synchronizing in  $\{q_{\#}\}$  from  $q_0$  (in  $\mathcal{G}$ ), then by Lemma 3 there exists an SCC  $C$  in  $\mathcal{P}(\mathcal{G})$  containing  $\{q_{\#}\}$ . Given a path  $s_0, s_1, \dots, s_k$  (induced by a sequence of selectors  $\alpha_1, \alpha_2, \dots, \alpha_k$ ) from  $\{q_{\#}\}$  to itself in  $\mathcal{P}(\mathcal{G})$  ( $s_0 = s_k = \{q_{\#}\}$ ), consider the largest index  $i < k$  such that  $q_{\#} \in s_i$ . It is easy to see that the selectors  $\alpha_{i+2}, \alpha_{i+3}, \dots, \alpha_k$  play only actions in  $A$ , and that they define a sure-winning strategy for eventually synchronizing in  $T$  from  $q_0$  (in  $\mathcal{M}$ ), which concludes the proof.  $\square$

## 4.2 Weakly synchronizing in stochastic games

We present an algorithm to compute the almost-sure winning region for weakly synchronizing objectives in stochastic games, which generalizes the result of Section 4.1. This algorithm has the flavor of the algorithm for deterministic games, with additional complications due to the probabilistic transitions in the game. The proof is also more technical because we no longer assume that pure counting strategies are sufficient for player 1 (but we show that such strategies are indeed always sufficient for almost-sure winning).

Recall that throughout this section we consider a stochastic game  $\mathcal{G} = \langle Q, A, \delta \rangle$  and we denote by  $n = |Q|$  the size of the state space, and by  $\eta$  the smallest positive probability in the transitions of  $\mathcal{G}$ . We consider the almost-sure weakly synchronizing objective defined by a set  $T \subseteq Q$  of accepting states.

Given a set  $U \subseteq Q$ , consider the sequence  $U_i = \text{CPre}^i(U)$  for  $i \geq 1$  (and  $U_0 = U$ ). Since  $U_i \subseteq Q$ , this sequence is ultimately periodic. Consider the least  $k \geq 0$  for which there exists  $r > 1$  such that  $U_k = U_{k+r}$ , and consider the least such  $r$ , called the *period*. It is easy to see that  $k, r \leq 2^n$ . For  $R = U_k$  we call  $\langle R, r, k \rangle$  the *periodic scheme* of  $U$  and we refer to its elements as  $R(U) = U_k$ ,  $r(U) = r$ , and  $k(U) = k$ . The set  $U$  is *self-recurrent* if  $U \neq \emptyset$  and there exists an index  $0 \leq t < r$  such that all states in  $U \times \{t\}$  are almost-sure winning for the (state-based) reachability objective  $\diamond(R \times \{0\})$  in  $\mathcal{G} \times [r]$ . The intuitive meaning of being self-recurrent appears in Lemma 7 below.

Note that in deterministic games  $\mathcal{G}$ , an accepting set  $U \subseteq T$  contained in a strongly connected component  $C$  of  $\mathcal{P}(\mathcal{G})$  is self-recurrent. Self-recurrent sets are the key to generalize the result of Lemma 3 to stochastic games. The argument of the proof is more involved, and presented in the following three lemmas.

LEMMA 7. *If there exists a self-recurrent set  $U \subseteq T$ , then there exists a state from which player 1 is almost-sure winning for weakly synchronizing in  $T$ , using a pure and counting strategy.*

PROOF. Let  $\langle R, r, k \rangle$  be the periodic scheme of  $U$ . We show that if all states in  $U \times \{t\}$  are almost-sure winning for the (state-based) reachability objective  $\diamond(R \times \{0\})$  in  $\mathcal{G} \times [r]$ , then we can construct a strategy  $\sigma_{\text{as}}$  for player 1 in  $\mathcal{G}$  that is almost-sure winning for weakly synchronizing in  $U$  (and thus also in  $T$ ) from all distributions with support in  $U$ .

The key argument is to show that for all distributions  $d$  on  $U$ , for all  $\varepsilon > 0$ , there exists a strategy  $\sigma_\varepsilon$  for player 1 that ensures from  $d \times \{t\}$ , against all strategies of player 2, that after finitely many rounds, a distribution  $d' \times \{t'\}$  is reached such that (i)  $d'(U) > 1 - \varepsilon$ , and (ii)  $d' \times \{t'\}$  is almost-sure winning for the reachability objective  $\diamond(R \times \{0\})$  in  $\mathcal{G} \times [r]$ . From this key argument, we can construct a pure and counting strategy  $\sigma_{\text{as}}$  for player 1 that successively plays according to the strategies  $\sigma_{\frac{1}{2}}, \sigma_{\frac{1}{4}}, \sigma_{\frac{1}{8}}, \dots$  and show that it is almost-sure weakly synchronizing in  $U$  (and thus also in  $T$ ) from the distribution  $d$ , which concludes the proof.

To prove the key argument, we construct  $\sigma_\varepsilon$  as follows (see also [25, Theorem 7]). Since  $d \times \{t\}$  is almost-sure winning for the reachability objective  $\diamond(R \times \{0\})$  in  $\mathcal{G} \times [r]$ , there exists a (pure memoryless) strategy  $\sigma$  and an integer  $h_\varepsilon$  such that for all strategies  $\tau$  of player 2, we have  $\Pr_{q_0}^{\sigma, \tau}(\diamond^{\leq h_\varepsilon} R \times \{0\}) \geq 1 - \varepsilon$  (Lemma 1).

We construct the strategy  $\sigma_\varepsilon$  to play according to  $\sigma$  as long as no state in  $R \times \{0\}$  is reached. Whenever a state in  $R \times \{0\}$  is reached, if it happens within the first  $h_\varepsilon$  rounds, then the strategy  $\sigma_\varepsilon$  plays to reach again  $R \times \{0\}$  after  $r$  more rounds (which is possible since  $R = \text{CPre}^r(R)$ ); if it happens after  $h_\varepsilon$  rounds, then it plays according to a sure-winning strategy for eventually synchronizing in  $U$  from  $R$  (thus for  $k$  more steps, where  $k$  is such that  $R = \text{CPre}^k(U)$ ). It immediately follows that for all strategies  $\tau$  of player 2, after finitely many rounds (at most  $h_\varepsilon + r + k$  rounds) a distribution  $d' \times \{t'\}$  is reached such that (i)  $d'(U) \geq 1 - \varepsilon$ , and (ii)  $d' \times \{t'\}$  is almost-sure winning for the reachability objective  $\diamond(R \times \{0\})$  in  $\mathcal{G} \times [r]$  (recall that for a reachability objective, a distribution is almost-sure winning if all states in its support are almost-sure winning).  $\square$

For the converse of Lemma 7, the structure of the argument is similar to the proof of Lemma 3 for deterministic games and pure strategies. However, the technical details are more involved due to stochasticity (in the game graph, and in the strategy of player 1).

For convenience, we separate the structure of the argument and the technical details.

*Structure of the argument (substitution game).* We present the *substitution game*, loosely inspired by player substitutions in ice hockey: for simplicity there is a single player on ice and a coach who manages (i) a main team of players (those who have been on ice at least once during the game) initially empty, and (ii) a reserve team of  $K$  players (fresh players who have never been on ice). We consider the following protocol for substitutions of players. A player stays on ice for a one-minute period, then needs to rest and to be substituted: he returns to the main team, and a player from the main team (possibly the same player) is selected by the coach to go on ice. At each one-minute period, the players in the main team have the possibility to *pass* (e.g., if they are tired), which they can do whenever they want, not necessarily on consecutive periods but at most a fixed number  $N$  of times. When a player returns from ice to the main team, he gets recharged with  $N$  passes. The next player to go on ice is chosen, at the discretion of the coach, among the players of the main team who did not pass. If at some period all players in the main team pass, a player from

the reserve team is called to go on ice instead (reserve players cannot pass), and will join the main team after a one-minute period, thereby getting  $N$  possibilities to pass. Initially, the main team is empty. The question is how many players should there be initially in the reserve team, in order for the coach to be able to play the substitution game forever regardless of when the players decide to pass, and what is an optimal strategy for the coach (assuming there is no player on ice initially).

Formally, a configuration of the substitution game consists of a set  $M$ , initially  $M = \emptyset$ , and a function  $f : M \rightarrow \mathbb{N}$  that maps each member of the main team to their number of passes. Given a configuration  $\langle M, f \rangle$ , the players choose a set  $P \subseteq \{p \in M \mid f(p) \geq 1\}$  of players who will pass, and then the coach chooses a player  $p$  such that  $p \in M \setminus P$  if  $P \neq M$ , and  $p = |M|$  if  $P = M$ . The next configuration is  $\langle M', f' \rangle$  where  $M' = M \cup \{p\}$  and  $f'$  is defined by  $f'(p) = N$ , and for all  $m \in M \setminus \{p\}$ :

$$f'(m) = \begin{cases} f(m) - 1 & \text{if } m \in P, \\ f(m) & \text{if } m \notin P. \end{cases}$$

The game is won by the coach if there exists  $K$  (the number of reserve players) such that the game continues forever, with  $|M| \leq K$  in all configurations.

It is easy to see that  $K \geq N + 1$  is necessary for the coach to win (e.g., if all players always pass, then  $K = N$  players is not sufficient). To show that  $K = N + 1$  is sufficient, consider the strategy of the coach that, given the current configuration  $\langle M, f \rangle$ , chooses a player  $p$  in  $M \setminus P$  with largest number of passes (if  $P \neq M$ , that is  $p \in \operatorname{argmax}_{x \in M \setminus P} f(x)$ ). Under this strategy, the following invariant holds in all configurations  $\langle M, f \rangle$  during the game: for  $k = |M|$ , let  $p_0, p_1, \dots, p_{k-1}$  be the players in  $M$  in ascending order according to  $f$  (i.e., if  $i \leq j$ , then  $f(p_i) \leq f(p_j)$ ), then  $f(p_i) \leq N + 1 - k + i$ . The proof is by induction, where the initial configuration satisfies the invariant trivially, and given the invariant holds in a configuration  $\langle M, f \rangle$  with  $k = |M|$ , we consider two cases: (1) if  $P = M$ , then all players in  $M$  are passing, thus  $|M'| = k + 1$  in the next configuration  $\langle M', f' \rangle$  and  $f'(p_i) = f(p_i) - 1 \leq (N + 1 - k + i) - 1 = N + 1 - (k + 1) + i$  for  $p_i \in M$  (in ascending order), and for the new player  $p_k$  in position  $k$ , we verify that  $f'(p_k) = N \leq N + 1 - (k + 1) + k$ ; (2) if  $P \neq M$ , then  $|M'| = |M| = k$  in the next configuration  $\langle M', f' \rangle$ , and for the players  $p_i$  on the left of the player chosen by the coach, their position in  $M'$  is the same as in  $M$  (assuming that among the players with the same number of passes, the leftmost are passing if any, which is no loss of generality since the positions of two players with the same number of passes can be swapped while maintaining the invariant) and their value of  $f$  does not increase, and for the players  $p_i$  on the right of the player chosen by the coach, their position in  $M'$  and their value of  $f$  decrease by 1, thus in both cases  $f'(p_i) \leq N + 1 - k + i$  holds, while for the player chosen by the coach, its new position is  $k - 1$  and we verify that  $f'(p_{k-1}) = N \leq N + 1 - k + (k - 1)$ .

It follows from this invariant that  $f(p_0) \leq N + 1 - k$  for the first player in ascending order, and  $k \leq N + 1$  since  $f(p_0) \geq 0$ , that is each configuration has at most  $N + 1$  players in  $M$ , showing that  $N + 1$  reserve players are sufficient.

In the proof of Lemma 3, the players of the substitution game were the registers, and the number  $N$  of allowed passes was  $N = 2^n - 1$  (which is an upper bound on the number of positions that player 2 could not cover). We concluded that  $K = 2^n$  registers were sufficient, which allowed us to define a spoiling strategy for player 2 using  $K$  tokens, each representing a probability mass of  $\frac{1}{K}$ .

*Technical aspects.* For stochastic games, we use a structure of argument based on the substitution game to show the converse implication of Lemma 7. In Lemma 8 we construct a number  $\varepsilon_w > 0$  and a bound  $N_w$  on the number of positions that player 2 may not be able to cover against an arbitrary strategy of player 1, where covering a position informally means that a probability mass  $\varepsilon_w > 0$  is in  $Q \setminus T$  at that position.

LEMMA 8. Let  $\mathcal{G}$  be a stochastic game. There exists  $\varepsilon_w > 0$  and  $N_w \in \mathbb{N}$  such that the following holds: if there exists no set  $U \subseteq T$  that is self-recurrent, then in  $\mathcal{G}$  for all player-1 strategies  $\sigma$ , for all but at most  $N_w$  rounds  $i$ , there exists a player-2 strategy  $\tau$  such that  $\mathcal{G}_i^{\sigma, \tau}(T) \leq 1 - \varepsilon_w$ .

PROOF. Given a set  $U \subseteq T$  with periodic scheme  $\langle R, r, k \rangle$  and given  $0 \leq t < r$ , since  $U$  is not self-recurrent (by the assumption of the lemma) and using Lemma 2, there exists a strategy  $\tau_t^U$  for player 2 in  $\mathcal{H} = \mathcal{G} \times [r]$  (with initial distribution  $d_0$  such that  $\text{Supp}(d_0) = U \times \{t\}$ ), such that for all player-1 strategies  $\sigma$ , we have  $\mathcal{H}_i^{\sigma, \tau_t^U}(R \times \{0\}) \leq 1 - \eta_0 \cdot \eta^{n \cdot 2^n}$  where  $\eta_0 = \min\{d_0((q, t)) \mid q \in U\}$  is the smallest positive probability in  $d_0$ .

*Construction of the strategy  $\tau^U$ .* Each strategy  $\tau_t^U$  maps to a strategy  $\tau$  in  $\mathcal{G}$  via the bijection  $\mu_t$  (namely,  $\tau(\rho) = \tau_t^U(\mu_t(\rho))$ , see p.15), and by an abuse of notation we also denote by  $\tau_t^U$  the corresponding strategy  $\tau$  in  $\mathcal{G}$ . Thus  $\tau_t^U$  plays in  $\mathcal{G}$  as if the initial index (in  $\mathcal{G} \times [r]$ ) was  $t$ . Fix an arbitrary player-1 strategy  $\sigma$  in  $\mathcal{G}$ , and define the strategy  $\tau^U = \sum_{t=0}^{r(U)-1} \frac{1}{r(U)} \cdot \tau_t^U$  as the uniform superposition of the strategies  $\tau_t^U$  for  $t = 0, \dots, r(U) - 1$  in the (infinite-state) MDP obtained by fixing  $\sigma$  in  $\mathcal{G}$ . Notice that if  $d_0(q) \geq \eta_0$  for all  $q \in U$ , then from  $d_0$  the strategy  $\tau^U$  ensures, against player-1 strategy  $\sigma$ , that  $\mathcal{G}_i^{\sigma, \tau^U}(Q \setminus R) \geq \frac{\eta_0 \cdot \eta^{n \cdot 2^n}}{2^n}$  for all  $i \geq 0$ , since  $r(U) \leq 2^n$ .

*Construction of the set  $I_w$ .* We define the numbers  $\varepsilon_w > 0$  and  $N_w \in \mathbb{N}$  required in the lemma. Let  $N_w = 4^n$  and  $\varepsilon_w = \frac{1}{2^n} \cdot \left(\frac{\eta^{(n+1) \cdot 2^n}}{n \cdot 4^n}\right)^{2^n}$ .

To prove the lemma, given the arbitrary player-1 strategy  $\sigma$  in  $\mathcal{G}$ , we construct a set  $I_w$  of positions such that  $|I_w| \leq N_w$ , and for all  $i \in \mathbb{N} \setminus I_w$ , there exists a player-2 strategy  $\tau$  such that  $\mathcal{G}_i^{\sigma, \tau}(T) \leq 1 - \varepsilon_w$ .

We construct the set  $I_w$  iteratively as follows. We construct a sequence  $I = I_0, I_1, \dots$  of (finite) sets  $I_i \subseteq \mathbb{N}$  of indices (initially,  $I_0 = \emptyset$ ) and a sequence  $\mathcal{U} = \mathcal{U}_0, \mathcal{U}_1, \dots$  of sets  $\mathcal{U}_i \subseteq 2^T$  of nonempty subsets of accepting states (initially,  $\mathcal{U}_0 = \emptyset$ ). The sequence  $\mathcal{U}$  will be strictly increasing  $\mathcal{U}_0 \subsetneq \mathcal{U}_1 \subsetneq \dots$ , and therefore the construction has (at most)  $2^n$  iterations. Each iteration  $k$  corresponds to a round, numbered  $\hat{i}_k$ , and defines the sets  $I_k$  and  $\mathcal{U}_k$ .

At iteration  $k$  ( $k = 1, 2, \dots$ ), we consider the round  $\hat{i}_{k-1}$  defined from the previous iteration (initially,  $\hat{i}_0 = 0$ ), and either the construction terminates at iteration  $k$ , or we define a new index  $\hat{i}_k > \hat{i}_{k-1} + 2^n$ . We associate with index  $\hat{i}_k$  a set of sub-distributions<sup>2</sup> that sum up to the distribution  $d_{\hat{i}_k}$  in the outcome of the game  $\mathcal{G}$  (from initial distribution  $d_0$ ) under the player-1 strategy  $\sigma$  and some player-2 strategy  $\tau$ . The sub-distributions associated with index  $\hat{i}_k$  are described below (where  $K = 2^n$ ), see also Figure 11:

- for each  $U \in \mathcal{U}_k$ , we have  $(K - k)$  identical sub-distributions  $f^U$  (which we call a  $U$ -token);
- we have a sub-distribution  $g_k$  storing the remaining probability mass.

All tokens for a given set  $U$  were created at the iteration, say  $j_U$ , where  $U$  was added to  $\mathcal{U}$ . At the end of iteration  $j_U$ , each token carries probability  $\varepsilon_{j_U}$  in each state of  $U$  and the tokens are updated according to the strategy  $\tau^U$ , which ensures that at every step after  $j_U$ , a probability mass bounded away from 0 (namely,  $\varepsilon_{j_U} \cdot \frac{\eta^{n \cdot 2^n}}{2^n}$ ) lies in  $Q \setminus R(U)$  from where, by definition of  $R(U)$ , player 2 has the possibility, using the spoiling strategy against sure eventually synchronizing in  $U$ , to inject a fraction  $\eta^{k(U)}$  of the probability mass into  $Q \setminus U$  after  $k(U)$  steps. Since  $k(U) \leq 2^n$ , for all  $j \geq \hat{i}_k + 2^n$ , there is a player-2 strategy to ensure probability bounded away from 0 in  $Q \setminus U$  at step  $j$  (namely,  $\varepsilon_{j_U} \cdot \frac{\eta^{(n+1) \cdot 2^n}}{2^n}$ ). Then the state in  $Q \setminus U$  with largest probability mass, carries at least  $\frac{\varepsilon_{j_U}}{n} \cdot \frac{\eta^{(n+1) \cdot 2^n}}{2^n}$ . If that state is not in  $T$ , then the index  $j$  is covered. However, if that state is in  $T$  (and

<sup>2</sup>A sub-distribution on  $S$  is a function  $d : S \rightarrow [0, 1]$  such that  $\sum_{s \in S} d(s) \leq 1$ .

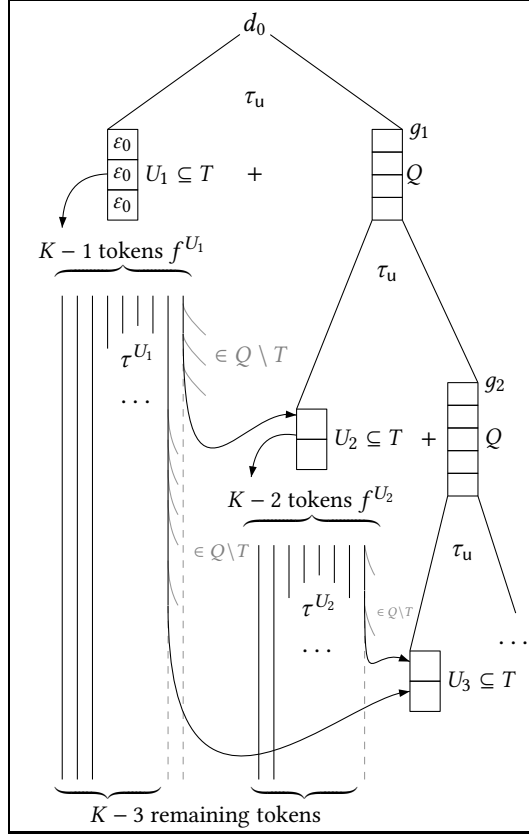


Fig. 11. The construction of the strategy  $\tau_w$  as the limit of the sequence  $\tau_0, \tau_1, \dots$  (proof of Lemma 8).

if this is the case for all  $U \in \mathcal{U}_k$ , then the index  $j$  is not covered, and we insert  $j$  into  $I_w$ . On the other hand, if player 2 does inject probability in  $Q \setminus U$  in this way, we can show that a new set  $U$  (i.e.,  $U \notin \mathcal{U}_k$ ) carries a probability bounded away from 0. Therefore, the situation where an index  $j$  is not covered cannot occur more than  $2^n$  times (which is an upper bound on the cardinality of the sets in the sequence  $\mathcal{U}$ ). Note that when player 2 does inject probability in  $Q \setminus U$ , one token is “consumed”, and therefore the number of tokens for each  $U$  will decrease (by 1) at each iteration. We create sufficiently many tokens for each  $U$  (namely  $K = 2^n$  tokens) to avoid depletion.

Initially we have  $\mathcal{U}_0 = \emptyset$  and  $g_0 = d_0$  is the (full) initial distribution. At iteration  $k$ , given index  $\hat{i}_{k-1}$  we first construct the index  $\hat{i}_k$  and then the sub-distributions associated with  $\hat{i}_k$ . The construction is illustrated in Figure 11.

The first iteration ( $k = 1$ ) of the construction is slightly different from the other iterations because  $\mathcal{U}_0$  is empty. Consider the outcome  $\mathcal{G}^{\sigma, \tau_u} = d_0, d_1, \dots$  from the initial distribution  $g_0 = d_0$  and the uniform strategy  $\tau_u$ . Let  $\hat{i}_1$  be the smallest index  $j \geq 0$  such that the set  $T \cap \{q \mid d_j(q) \geq \varepsilon_0\}$  is nonempty, where  $\varepsilon_0 = \frac{1}{2n}$ . We consider two cases: (1) if no such index  $j$  exists, then  $d_j(T) \leq n \cdot \varepsilon_0 = \frac{1}{2} \leq \varepsilon_w$ , for all  $j \geq 0$ , hence we can take  $I_w = \emptyset$  and the construction terminates; (2) otherwise, let  $U_1 = T \cap \{q \mid d_{\hat{i}_1}(q) \geq \varepsilon_0\} \neq \emptyset$  and decompose the sub-distribution  $d_{\hat{i}_1}$  as  $d_{\hat{i}_1} = (K-1) \cdot f^{U_1} + g_1$  where  $f^{U_1}(q) = \frac{\varepsilon_0}{K-1}$  if  $q \in U_1$ , and  $f^{U_1}(q) = 0$  otherwise (and let  $g_1 = d_{\hat{i}_1} - (K-1) \cdot f^{U_1}$ ), thus



$f^{U_1}$  and  $g_1$  are sub-distributions. We update  $I$  and  $\mathcal{U}$  as follows: let  $I_1 := I_0 \cup [\hat{i}_1, \hat{i}_1 + 2^n]$  and  $\mathcal{U}_1 := \mathcal{U}_0 \cup \{U_1\}$ . Let  $\varepsilon_1 = \frac{\varepsilon_0}{K-1}$  be the probability mass in each state of  $U_1$  according to  $f^{U_1}$ .

Now we present iteration  $k$  for  $k \geq 2$ . For each  $U \in \mathcal{U}_{k-1}$ , pick one of the  $K - k + 1$  tokens of iteration  $k - 1$  (i.e., a sub-distribution  $f^U$  associated with index  $\hat{i}_{k-1}$  that carried probability mass at least  $\varepsilon_{k-1}$  in each state of  $U$  when  $U$  was added to  $\mathcal{U}$ ) and for each  $j \geq \hat{i}_{k-1} + 2^n$ , consider the strategy  $\tau^U$  played until step  $j - k(U)$ , then the spoiling strategy for sure eventually synchronizing in  $U$ . The strategy  $\tau^U$  would ensure probability at least  $\frac{\varepsilon_{k-1} \cdot \eta^{n \cdot 2^n}}{2^n}$  in  $Q \setminus R(U)$  at every step, thus in particular at step  $j - k(U)$ , and the spoiling strategy can inject probability at least  $\beta \cdot \varepsilon_{k-1}$  in  $Q \setminus U$  (at step  $j$ ) where  $\beta = \eta^{2^n} \cdot \frac{\eta^{n \cdot 2^n}}{2^n} \leq \eta^{k(U)} \cdot \frac{\eta^{n \cdot 2^n}}{2^n}$ .

Hence under that strategy, there is a state  $q_j^U \in Q \setminus U$  that contains probability at least  $\frac{\beta}{n} \cdot \varepsilon_{k-1}$ . We consider two cases: (1) if for all  $j \geq \hat{i}_{k-1} + 2^n$  there exists  $U \in \mathcal{U}_{k-1}$  such that  $q_j^U \notin T$ , then the construction is over and we define  $I_w = I_{k-1}$ ; (2) otherwise, let  $\hat{i}_k$  be the smallest index  $j \geq \hat{i}_{k-1} + 2^n$  such that  $q_j^U \in T$  for all  $U \in \mathcal{U}_{k-1}$ , and consider the sub-distribution  $d$  at step  $\hat{i}_k$  that originates from the superposition of playing  $\tau_u$  from  $g_{k-1}$  and playing, from each  $U$ -token for  $U \in \mathcal{U}_{k-1}$  the strategy  $\tau^U$  followed (at step  $j - k(U)$  for one of the  $U$ -tokens) by the spoiling strategy for sure eventually synchronizing in  $U$ . Let  $U_k = T \cap \{q \mid d(q) \geq \frac{\beta}{n} \cdot \varepsilon_{k-1}\} \neq \emptyset$  be the set of accepting states that carry a sufficiently significant probability mass according to  $d$ . Note that  $q_j^U \in U_k$  for all  $U \in \mathcal{U}_{k-1}$ , and since  $q_j^U \notin U$ , it follows that  $U_k \neq U$  for all  $U \in \mathcal{U}_{k-1}$ , that is  $U_k \notin \mathcal{U}_{k-1}$ . Decompose the sub-distribution  $d$  as  $d = (K - k) \cdot f^{U_k} + g_k$  where  $f^{U_k}(q) = \frac{\beta}{n} \cdot \frac{\varepsilon_{k-1}}{K-k}$  if  $q \in U_k$ , and  $f^{U_k}(q) = 0$  otherwise (and let  $g_k = d - (K - k) \cdot f^{U_k}$ ), thus  $f^{U_k}$  and  $g_k$  are sub-distributions.

With step  $\hat{i}_k$ , we associate the sub-distribution  $g_k$ , the  $K - k$  copies of the sub-distribution  $f^{U_k}$ , and for each  $U \in \mathcal{U}_{k-1}$  the  $K - k$  remaining  $U$ -tokens updated according to  $\tau^U$  (see Figure 11). We update  $I$  and  $\mathcal{U}$  as follows: let  $I_k := I_{k-1} \cup [\hat{i}_k, \hat{i}_k + 2^n]$  and  $\mathcal{U}_k := \mathcal{U}_{k-1} \cup \{U_k\}$ . Let  $\varepsilon_k = \frac{\beta}{n} \cdot \frac{\varepsilon_{k-1}}{K-k}$ .

The construction must terminate after at most  $2^n$  iterations because the cardinality of the sets  $\mathcal{U}_i$  is increasing by 1 at each iteration (in fact  $|\mathcal{U}_i| = i$ ), and  $\mathcal{U}_i \subseteq 2^T$  thus  $|\mathcal{U}_i|$  is bounded by  $2^n$ . We note that  $I_k$  is the union of  $k$  intervals of size  $2^n$ , thus  $|I_k| \leq k \cdot 2^n \leq 4^n$ . If the construction terminates before the first iteration is complete, then we have already shown that the result of the lemma holds. When the construction terminates, say during iteration  $k_{\max} < K = 2^n$ , then for all  $k \leq k_{\max}$ , considering the two cases, in all rounds  $j$  between  $\hat{i}_{k-1} + 2^n$  and  $\hat{i}_k$  and in all rounds  $j$  after round  $\hat{i}_{k_{\max}} + 2^n$ , player 2 can inject probability  $\frac{\beta}{n} \cdot \frac{\varepsilon_{k-1}}{K} \geq \varepsilon_K$  in  $q_j^U$  for  $U = U_k$ , thus outside  $T$ , which concludes the proof with  $N_w = 4^n$  and  $\varepsilon_w = \varepsilon_K > \varepsilon_0 \cdot \left(\frac{\beta}{n \cdot K}\right)^{2^n} = \frac{1}{2n} \cdot \left(\frac{\eta^{(n+1) \cdot 2^n}}{n \cdot 4^n}\right)^{2^n}$ .  $\square$

In the proof of Lemma 8, the set  $I_w$  of cardinality  $N_w$  corresponds to the positions that player 2 does not cover from an initial distribution  $d_0$  (where he would pass in the substitution game). Note the order of the quantifiers in the statement of Lemma 8: player 2 may use different strategies to cover different positions. We use the structure of argument of the substitution game to show that a single strategy of player 2 can cover all but finitely many positions, and we obtain the generalization of Lemma 3 to stochastic games.

**LEMMA 9.** *Let  $\mathcal{G}$  be a stochastic game. The following equivalence holds: there exists a self-recurrent set  $U \subseteq T$ , if and only if, there exists a state from which player 1 is almost-sure winning for weakly synchronizing in  $T$ .*

**PROOF.** One direction of the lemma is given by Lemma 7. For the converse direction, the proof uses Lemma 8 and is similar to the proof of Lemma 3.

**Algorithm 2:** *Solve*( $\mathcal{H}, p, T$ )**Input** :  $\mathcal{G} = \langle Q, A, \delta \rangle$  is a stochastic game,  $T \subseteq Q$  is a target set.**Output** : The set  $\{\text{Supp}(d) \mid d \in \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(\mathcal{G}, T)\}$ .**begin**

```

1    $r \leftarrow 1$ 
2    $\mathcal{H} \leftarrow \mathcal{G} \times [r]$ 
3    $S \leftarrow Q \times [r]$ 
4    $K \leftarrow S$ 
5   repeat
6     if there is a self-recurrent set  $U \subseteq T$  in  $\mathcal{H} \upharpoonright [K]$  then
7       Let  $\langle R, r, k \rangle$  be the periodic scheme of  $U$ 
8        $\mathcal{H} \leftarrow \mathcal{G} \times [r]$ 
9        $K, S \leftarrow \text{expand}(r, K, S)$ 
10      Let  $W$  be the almost-sure winning
11      region for the (state-based) reachability
12      objective  $\diamond(R \times \{k \bmod r\})$  in  $\mathcal{H} \upharpoonright [K]$ 
13       $X \leftarrow \text{PosAttr}_1(W, \mathcal{H} \upharpoonright [K])$ 
14       $K \leftarrow K \setminus X$ 
15    else
16       $L \leftarrow \text{PosAttr}_2(K, \mathcal{H} \upharpoonright [S])$ 
17       $S \leftarrow S \setminus L$ 
18       $K \leftarrow S$ 
19    until  $K = \emptyset$ 
20  return  $\{s \subseteq Q \mid \exists i : s \times \{i\} \subseteq S\}$ 

```

Given an arbitrary strategy  $\sigma$  for player 1, by Lemma 8 there exists an index  $i_0$  (namely,  $i_0 = \max I_w$  the largest index in the finite set  $I_w$  constructed in the proof of Lemma 8) such that for all  $i \geq i_0$ , there exists a player-2 strategy  $\tau^i$  such that  $d_i = \mathcal{G}_i^{\sigma, \tau^i}(T) \leq 1 - \varepsilon_w$ . The strategies  $\tau^i$  correspond to the reserve team of tokens (in the substitution game).

Now for each  $d_i$  (as an initial distribution), considering the strategy  $\sigma$  of player 1 played from  $d_i$ , we also get from Lemma 8 a finite set  $I_{d_i}$  of at most  $N_w$  indices, which are the times associated with  $d_i$  where player 2 will pass in the substitution game. As we know that at most  $N_w + 1$  tokens from the reserve team may be needed in the substitution game, there is a time  $i_0^*$  after which no token will ever be transferred from the reserve team and let  $K$  be the number of token in the main team of tokens at time  $i_0^*$ . We assign probability mass  $\frac{1}{K}$  to each token. At every step  $i \geq i_0^*$ , there is a token that does not pass, and therefore there exists a strategy  $\tau$  transforming that token into a sub-distribution that carries probability at least  $\frac{\varepsilon_w}{K}$  in  $Q \setminus T$  at time  $i$ . The resulting sub-distribution (at time  $i$ ) has again from Lemma 8 a finite set of indices of size  $N_w$  where it can pass, as in the substitution game. Hence the process can continue forever, and at every step  $i \geq i_0^*$ , there is a probability mass at least  $\frac{\varepsilon_w}{K}$  in  $Q \setminus T$ , showing that  $\sigma$  is not almost-sure winning for weakly synchronizing in  $T$ .  $\square$

We present Algorithm 2 to compute the almost-sure winning set for weakly synchronizing objectives. We use the game of Figure 12 for illustration. The game contains a self-recurrent set

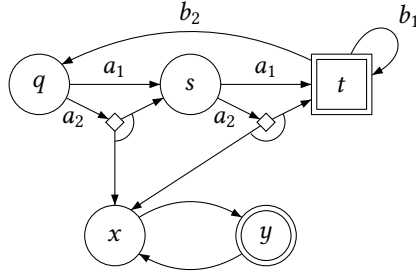


Fig. 12. A stochastic game  $\mathcal{G}$  to illustrate the execution of Algorithm 2 (see also Fig. 13).

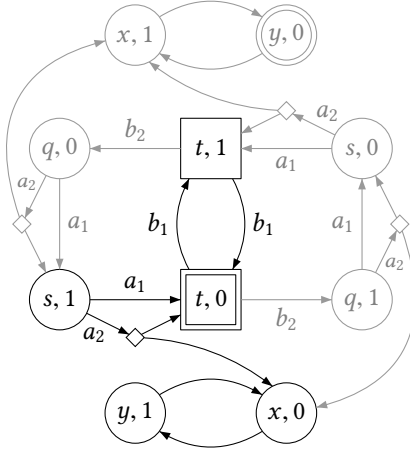


Fig. 13. The game  $\mathcal{H} = \mathcal{G} \times [2]$  for the game  $\mathcal{G}$  of Fig. 12 with the shaded region  $X$  computed by Algorithm 2 (line 11).

$U = \{y\}$  (with period 2), thus player 1 is winning from  $x$  and from  $y$ . Player 1 is also winning from the other states: the mass of probability that eventually stays in  $t$  is winning, and the remaining mass of probability can be injected in  $U$  by player 1 at the correct times to be synchronized modulo the period 2, thanks to the consecutive transitions on  $a_2$  from  $q$  and from  $s$ .

Given the game  $\mathcal{G}$  as input, the algorithm considers subgames of  $\mathcal{H} = \mathcal{G} \times [r]$  where  $r = 1$  initially (lines 1-2), with state space  $S$ , initially  $S = Q \times [1]$  (line 3). The auxiliary variable  $K$  (line 4) is used to compute losing states for player 1.

The algorithm proceeds iteratively to construct  $K$ , by removing states from  $S$ . In the loop of line 5, as long as there is a self-recurrent set  $U$  in the subgame  $\mathcal{H} \upharpoonright [K]$ , we expand the state space to track the number of rounds modulo the period of  $U$  (which must be a multiple of the period of  $\mathcal{H}$ ). The map between the state spaces of the expanded game and of the original game is defined as follows. Given a set  $S \subseteq Q \times [p]$ , and a period  $r$  that is a multiple of  $p$ , the  $r$ -expansion of  $S$  is the set  $\{\langle q, i \rangle \mid 0 \leq i \leq r-1 \wedge \langle q, i \bmod p \rangle \in S\}$ . The expand function computes the  $r$ -expansion of  $S$  and  $K$  at line 9 (where  $p = |\{i \mid \langle q, i \rangle \in S\}|$  can be derived from the set  $S$ ). In the expanded game  $\mathcal{H}$  (line 8), we compute the almost-sure winning region  $W$  for the (state-based) reachability

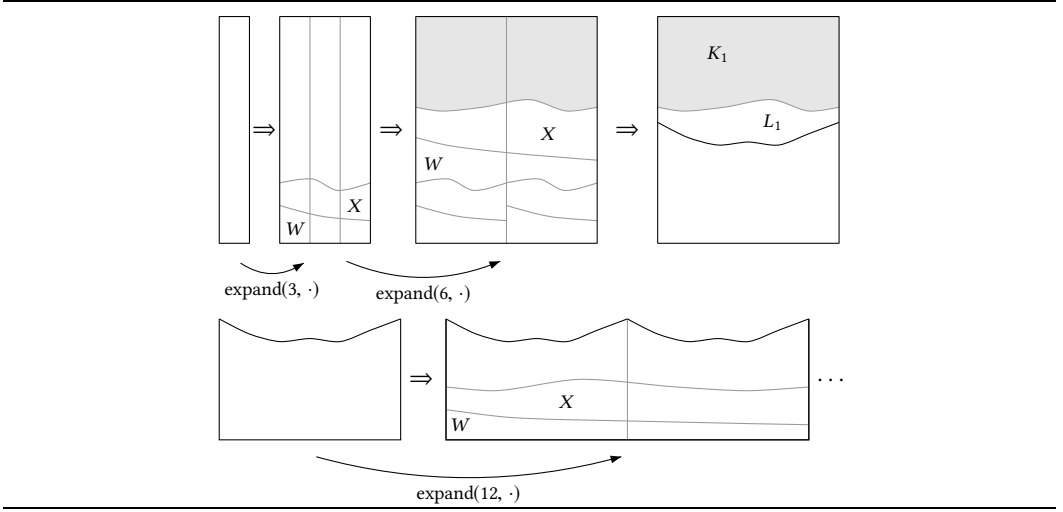


Fig. 14. View of the iterations of Algorithm 2.

objective  $\diamond(R \times \{k\})$ , which we call a *core winning region*, where  $k = k(U) \bmod r$ . From the states in  $W$  player 1 is almost-sure weakly synchronizing in  $T \times \{0\}$  (see the proof of Lemma 7). Figure 13 shows the 2-expansion of the game of Figure 12. The value  $k$  is such that from  $R \times \{k\}$  player 1 can inject all the probability mass into  $T \times \{0\}$  (in fact, in  $U \times \{0\}$ ).

The next iteration starts after removing from the state space  $K$  the positive attractor for player 1 to  $W$  (lines 11-12), thus ensuring  $\mathcal{H} \upharpoonright [K]$  is again a subgame (the dark part of Figure 13). Whenever there is no set  $U$  in  $\mathcal{H} \upharpoonright [K]$  satisfying the conditions of Lemma 9, the whole state space  $K$  is losing for player 1 and we remove its positive attractor for player 2 (lines 13-15). This part of the algorithm is illustrated in the game of Figure 15, where the self-recurrent set  $U = \{y\}$  (with period 1) induces a core winning region  $W = \{x, y\}$ , and in the subgame obtained by removing the positive attractor for player 1 to  $W$ , the set  $U = \{q\}$  is self-recurrent. The remaining subgame with state space  $\{s\}$  has no self-recurrent set, thus we remove  $s$  and its positive attractor  $\{q, s\}$  for player 2, showing that  $q$  and  $s$  are losing for player 1.

The loop (line 5) terminates when  $K = \emptyset$ , which can happen if either the state space  $S$  can be partitioned by positive attractors to core winning regions, and then the whole state space  $S$  is winning for player 1, or if all states in  $S$  are losing ( $L = S$ ), and then the winning region for player 1 is empty. The algorithm then returns the slices of the winning region, which correspond to the support of the winning distributions. Figure 14 illustrates the first iterations of the algorithm.

*Termination.* The termination of Algorithm 2 is established by showing that  $K \subseteq S$  is an invariant of the repeat-loop and that at every iteration, either (i) the size of  $S$  is strictly decreasing, or (ii) the size of  $S$  is unchanged, and the size of  $K$  is strictly decreasing, where the *size* of  $S$  (and  $K$ ) is defined by  $\max_i |\{q \in Q \mid \langle q, i \rangle \in S\}|$  (and similarly for  $K$ ). Note that the expand function may increase the cardinality of  $S$  and  $K$ , but not their size (line 9). To show (i) and (ii), if the condition of the test (line 6) holds, then  $W$  is nonempty and contains at least one state for each value of the tracking counter, thus the size of  $K$  decreases (and  $S$  remains unchanged); otherwise, the size of  $S$  decreases (line 14) since  $K$  is nonempty at the beginning of each iteration (the loop terminates if  $K = \emptyset$ ). It follows that the number of executions of the main loop (line 5) is at most  $n^2$ .

*Correctness.* Let  $\mathcal{G}$  be a game with state space  $Q$ , let  $T \subseteq Q$ .

In the main loop of Algorithm 2 (line 5), let  $L_1, L_2, \dots, L_{i_\ell}$  be the sequence of values of the variable  $L$  successively computed in the else-clause (line 13) and let  $X_1, X_2, \dots, X_{i_x}$  be the values of the variable  $X$  successively computed in the then-clause *after* the last execution of the else-clause (line 11). Each of these two sequences may be empty, but not both. The sets  $L_i$  and  $X_i$  are possibly computed with different values of  $r$ . In the sequel, we consider their  $p$ -expansion for the value  $p$  of  $r$  when the algorithm terminates, which leads to the property that in the game  $\mathcal{H} = \mathcal{G} \times [p]$ , the sets  $\text{expand}(p, L_i)$  ( $i = 1, \dots, i_\ell$ ) and  $\text{expand}(p, X_j)$  ( $j = 1, \dots, i_x$ ) form a partition of the state space  $Q \times [r]$  (Figure 16). We often omit the  $\text{expand}$  function and write  $L_i$  instead of  $\text{expand}(p, L_i)$  (and analogously for  $X_j$ ).

We refine this partition with a decomposition of the losing states  $L_i$ , and of the winning states  $X_i$  (Figure 17). There exist sets  $K_i, F_i$  for  $i = 1, \dots, i_\ell$ , and  $F_{i_\ell+1}$  such that  $F_1 = Q \times [r]$  and for all  $1 \leq i \leq i_\ell$ :

- $K_i$  is a trap for player 1 in  $\mathcal{H} \upharpoonright [F_i]$ ,
- $L_i = \text{PosAttr}_2(K_i, \mathcal{H} \upharpoonright [F_i])$ ,
- $F_{i+1} = F_i \setminus L_i$ .

The set  $K_i$  is the value of variable  $K$  (up to  $p$ -expansion) when  $L_i$  is computed (at line 13), and the value  $F_i$  is the value of variable  $S$  computed in the next line (line 14). On the other hand, there exist nonempty sets  $U_i, R_i, S_i, W_i$  for  $i = 1, \dots, i_x$ , and  $U_{i_x+1}$  such that  $S_1 = F_{i_\ell+1}$  and for all  $1 \leq i \leq i_x$ :

- $U_i$  is a self-recurrent set in  $\mathcal{H} \upharpoonright [S_i]$ , with periodic scheme  $\langle R_i, r_i, k_i \rangle$  where  $r_i = p$  (since  $\mathcal{H}$  is the  $p$ -expansion of  $\mathcal{G}$ , thus the least possible period in a subgame of  $\mathcal{H}$  is  $p$ ).
- $X_i = \text{PosAttr}_1(W_i, \mathcal{H} \upharpoonright [S_i])$  where  $W_i$  is the almost-sure winning region for reachability to  $R_i \times \{k_i\}$  in  $\mathcal{H} \upharpoonright [S_i]$ ,
- $S_{i+1} = S_i \setminus X_i$ .

Note that  $S_{i_x+1} = \emptyset$  since the sets  $\{L_i\}_{1 \leq i \leq i_\ell}$  and  $\{X_j\}_{1 \leq j \leq i_x}$  form a partition of the state space  $Q \times [p]$ . The values  $U_i, R_i, W_i$  correspond to the variables  $U, R, W$  (up to  $p$ -expansion) at the iteration where  $X_i$  is computed (lines 6-12). The set  $S_i$  is the value of variable  $K$  at the beginning of that iteration. Note that in the subgame  $\mathcal{H} \upharpoonright [S_i]$  player 2 cannot play the actions for which positive probability leaves  $S_i$  (Figure 17).

We now show that in the game  $\mathcal{H} = \mathcal{G} \times [p]$ , every state in  $\bigcup_i L_i$  is losing, and every state in  $\bigcup_j X_j$  is winning, for almost-sure weakly synchronizing in  $T \times \{0\}$ .

For states in  $\bigcup_i L_i$ , the claim follows from Lemma 9 and the fact that  $K_i$  is a trap for player 1 in  $\mathcal{H} \upharpoonright [F_i]$ . In  $L_i \setminus K_i$ , player 2 uses the attractor strategy, and in  $K_i$ , for every strategy of player 1 there

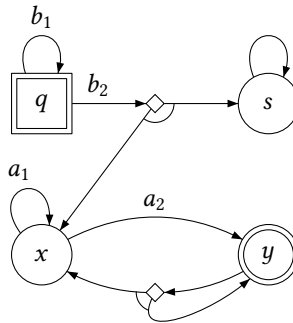


Fig. 15. A stochastic game to illustrate the execution of Algorithm 2 (lines 13-15).

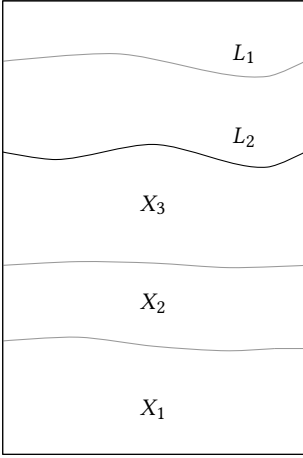


Fig. 16. The sets  $L_i$  of losing states, and  $X_j$  of winning states in  $\mathcal{G} \times [r]$  computed by Algorithm 2 (see *Correctness*).

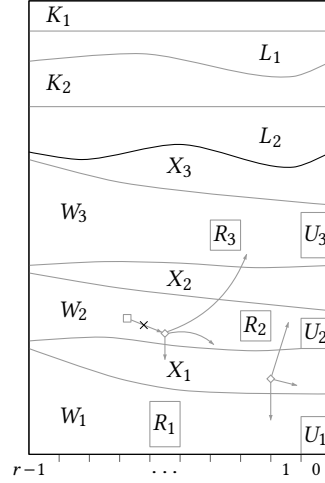


Fig. 17. Refined view of the sets  $L_i$  of losing states, and  $X_j$  of winning states in  $\mathcal{G} \times [r]$  (from Fig. 16).

is a strategy of player 2 to avoid almost-sure weakly synchronizing in  $T \times \{0\}$ . By the attractor strategy, the probability mass that reaches  $K_i$  is bounded (at least  $\eta^n$ ) regardless of the strategy of player 1, and thus there is also a lower bound on the probability mass outside  $T \times \{0\}$  from some point on.

For states in  $X := S_1 = \bigcup_j X_j$ , we present an almost-sure winning strategy  $\sigma_{as}$  for player 1 that successively plays according to the strategies  $\sigma_1, \sigma_2, \dots$  where  $\sigma_N$  is defined as follows, for all  $N \geq 1$ . The strategy  $\sigma_N$  plays in two phases: in the first phase, whenever a state in  $R = \bigcup_j R_j \times \{k_j \bmod p\}$  is reached, it plays to reach again  $R$  after  $p$  rounds (which is possible since  $p = r_j$ ); if no state in  $R$  was reached, it plays according to the (memoryless) attractor strategy in  $X_j \setminus W_j$ , and according to the (memoryless) almost-sure winning strategy to reach  $R_j \times \{k_j \bmod p\}$  in  $W_j$ . This first phase is played for  $N$  rounds. Let  $c$  be the value of the tracking counter at the end of the first phase. The second phase is played for  $N'$  rounds such that  $N' > 2^n$  and  $c + N' = 0 \bmod p$ , thus the tracking counter will be 0 at the end of the second phase. In the second phase, the strategy  $\sigma_N$  plays like in the first phase, except if a state in  $R_j \times \{k_j \bmod p\}$  is reached at step  $N + N' - k_j$  (note that  $N' > 2^n \geq k_j$ ), where the strategy then plays according to the sure-winning strategy for eventually synchronizing in  $U_j \times \{0\}$ . Note that  $\sigma_{as}$  is pure and counting.

For  $j = 1, \dots, i_x$ , consider the event

$$A_j = \{q_0 a_0 b_0 q_1 \dots \in (QAA)^\omega \mid \exists I \geq 0 \cdot \forall i \geq I : q_i \in W_j\}$$

where from some point on the play remains in the set  $W_j$ . We show that from every state in  $X$ , the strategy  $\sigma_{as}$  is almost-sure winning for weakly synchronizing in  $T \times \{0\}$ . The argument has two parts:

- (*Correctness under event  $A_j$* ). For an arbitrary  $\varepsilon > 0$  let  $N_\varepsilon$  given by Lemma 1. Under the event  $A_j$ , since the strategy  $\sigma_{N_\varepsilon}$  plays in  $W_j$  according to an almost-sure winning strategy for the reachability objective  $\diamond(R_j \times \{k_j \bmod p\})$  for  $N_\varepsilon$  steps, it follows that for all states  $\langle q, t \rangle \in X$  and strategies  $\tau$  of player 2 in  $\mathcal{H}$ ,

$$\Pr_{\langle q, t \rangle}^{\sigma_{N_\varepsilon}, \tau} (\diamond^{\leq N_\varepsilon} (R_j \times \{k_j \bmod p\}) \mid A_j) \geq 1 - \varepsilon,$$

and since  $U_j \times \{0\} \subseteq W_j$  we can repeat the argument after the second phase of the strategy  $\sigma_{N_\varepsilon}$ , and get  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\diamond(R_j \times \{k_j \bmod p\}) \mid A_j) = 1$  and  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\square \diamond R_j \times \{k_j \bmod p\} \mid A_j) = 1$ , which implies that  $\sigma_{as}$  is almost-sure winning for weakly synchronizing in  $T \times \{0\}$  under event  $A_j$ .

- (The event  $\bigcup_j A_j$  has probability 1). Now we show that the strategy  $\sigma_{as}$  is almost-sure winning for the event  $\bigcup_{1 \leq j \leq i_x} A_j$ , that is with probability 1 the play will remain forever in some  $W_j$ . Intuitively, this is because if a play visits infinitely often the positive attractor of  $W_1$  (namely,  $X_1$ ), then the set  $W_1$  is reached with probability 1 and never left since it is a trap for player 2 in the subgame  $\mathcal{H}[S_1]$  (where  $S_1$  is also a trap for player 2); on the other hand, if a play eventually remains outside  $X_1$ , then from some point on the play remains always in the subgame  $\mathcal{H}[S_2]$  (recall that  $S_2 = S_1 \setminus X_1$ ) and then visiting  $X_2$  infinitely often implies reaching and staying forever in  $W_2$  with probability 1. Repeating this argument  $i_x$  times shows that in all cases, the play has to remain forever in some  $W_j$  with probability 1. Formally, fix an arbitrary state  $\langle q, t \rangle \in S_1$ , and a strategy  $\tau$  of player 2 in  $\mathcal{H}$ , and we show that  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\bigcup_{1 \leq j \leq i_x} A_j) = 1$ . Let  $B_1 = \{q_0 a_0 b_0 q_1 \cdots \in (QAA)^\omega \mid \exists^\infty i \geq 0 : q_i \in X_1\}$  and for  $j = 2, \dots, i_x$ , let

$$B_j = \{q_0 a_0 b_0 q_1 \cdots \in (QAA)^\omega \mid \exists^\infty i \geq 0 : q_i \in X_j\} \setminus \bigcup_{l < j} B_l$$

be the event that  $X_j$  is visited infinitely often, and the states in  $X_1 \cup \dots \cup X_{j-1}$  are visited only finitely often.

Under event  $B_1$ , the positive attractor  $X_1$  of  $W_1$  is visited infinitely often, and therefore the set  $W_1$  is reached with probability 1 (by an argument similar to the proof of Lemma 1, under the positive-attractor strategy, there is a bounded probability  $\eta_1 > 0$  to reach  $W_1$  within a fixed number of steps, which entails that the probability to never reach  $W_1$  is  $\lim_{k \rightarrow \infty} (1 - \eta_1)^k = 0$ ). Moreover, once the play is in  $W_1$ , it remains there forever (by definition of the strategy  $\sigma_{as}$ , and because  $W_1$  is a trap for player 2 in  $\mathcal{H}$ ). Thus, we have  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(A_1 \mid B_1) = 1$  (if  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(B_1) \neq 0$ ). By a similar argument for  $j = 2, \dots, i_x$ , under event  $B_j$  the play eventually remains in the subgame  $\mathcal{H}[S_j]$  since  $S_j = S_1 \setminus \bigcup_{l < j} X_l$ , and it follows that  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(A_j \mid B_j) = 1$  (if  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(B_j) \neq 0$ ). Finally, since  $\{X_j\}_{1 \leq j \leq i_x}$  is a partition of  $S_1$  we have  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\bigcup_j A_j) = 1$ , and thus  $\Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\bigcup_{1 \leq j \leq i_x} A_j) = \Pr_{\langle q, t \rangle}^{\sigma_{as}, \tau}(\bigcup_{1 \leq j \leq i_x} A_j \mid \bigcup_{1 \leq j \leq i_x} B_j) = 1$ .

In the game  $\mathcal{G}$ , if  $1_{q_1}$  and  $1_{q_2}$  are almost-sure winning for weakly synchronizing in  $T$ , it does not necessarily imply that the distribution with support  $\{q_1, q_2\}$  is almost-sure winning for weakly synchronizing in  $T$ . However, we show that in the game  $\mathcal{H}$ , if the counter value is the same in two almost-sure winning states  $\langle q_1, t_1 \rangle$  and  $\langle q_2, t_2 \rangle$  (i.e.,  $t_1 = t_2 = t$ ), then the distribution with support  $\{\langle q_1, t \rangle, \langle q_2, t \rangle\}$  is also almost-sure winning. This is because under the strategy  $\sigma_{as}$ , the end of the second phase of the strategies  $\sigma_N$  occurs at the same time from all states with the same counter value. Therefore, from all states in  $X$  with a given counter value, and for all strategies  $\tau$  of player 2, the probability mass is at least  $1 - \varepsilon$  in  $T \times \{0\}$  at the last round played by each  $\sigma_N$  for all  $N$  sufficiently large.

Given a distribution  $d_0 \in \mathcal{D}(Q)$  over the states of  $\mathcal{G}$  if  $\text{Supp}(d_0)$  is a slice in  $X$ , that is there exists a counter value  $t$  such that  $\{\langle q, t \rangle \mid q \in \text{Supp}(d_0)\} \subseteq X$ , then  $d_0$  is an almost-sure winning distribution in  $\mathcal{G}$ . On the other hand, if for every  $0 \leq t < p$ , there is a state  $q \in \text{Supp}(d_0)$  such that  $\langle q, t \rangle \in L$ , then from  $d_0$ , player 2 can spoil all strategies of player 1, using a superposition of spoiling strategies for each  $\langle q, t \rangle \in L$ .

In conclusion, the sets  $s \subseteq Q$  such that  $s \times \{t\} \subseteq X = S_1$  for some  $t$  are the supports of the almost-sure winning distributions (line 16), which establishes the correctness of Algorithm 2.

LEMMA 10. *Given a stochastic game and a set  $T$  of target states, Algorithm 2 computes the supports of the distributions from which player 1 is almost-sure winning for weakly synchronizing in  $T$ . This algorithm can be implemented in PSPACE.*

PROOF. The correctness of the algorithm follows from the arguments given above. We show that this algorithm can be implemented in PSPACE by a similar proof as for deterministic games (Lemma 6). We already showed that the number of executions of the main loop (line 5) is at most  $n^2$  (see *Termination*), and since the period of the set  $U$  (line 6) is at most  $2^n$  times the period of the game  $\mathcal{H}$  (slices are subsets of  $Q$ ), the final value of  $r$  (upon termination of the algorithm) is at most  $(2^n)^{n^2} = 2^{n^3}$ .

A PSPACE implementation of Algorithm 2 can be obtained by using a PSPACE procedure to determine the transitions of  $\mathcal{H} \upharpoonright [K]$  and  $\mathcal{H} \upharpoonright [S]$ , as in the proof of Lemma 6. The computation of the positive attractors, and almost-sure winning regions can then be done in PSPACE as well.  $\square$

We obtain the following theorem, where the PSPACE upper bound is given by Lemma 10, a pure counting almost-sure winning strategy  $\sigma_{\text{as}}$  for player 1 is presented on p. 30, and the lower bound and memory requirement hold in the special case of MDPs [25, Theorem 6].

THEOREM 3. *The membership problem for almost-sure weakly synchronizing in stochastic games is PSPACE-complete, and pure counting strategies are sufficient for player 1. Infinite memory is necessary in general.*

### 4.3 Other synchronization objectives

The almost-sure winning region for the other synchronization objectives (always, eventually, and strongly) can be computed relatively easily, using the following lemmas.

LEMMA 11. *For always synchronizing, the sure and almost-sure winning modes coincide:*  
 $\langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{always}}(T) = \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{always}}(T)$ .

PROOF. The inclusion  $\langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{always}}(T) \subseteq \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{always}}(T)$  follows from the definitions (Section 2). For the converse inclusion, consider an initial distribution  $d_0$  from which player 1 has an almost-sure winning strategy for always synchronizing in  $T$ . We claim that player 1 has a strategy  $\sigma_{\text{safe}}$  to ensure, for all plays  $\rho = q_0 a_0 b_0 q_1 \dots q_k$  compatible with  $\sigma$ , that  $q_i \in T$  for all  $0 \leq i \leq k$  (i.e., player 1 is sure-winning for the safety objective  $\square T$  [22]). By contradiction, if that is not the case, then player 2 has a strategy to ensure reaching a state in  $Q \setminus T$  within at most  $n$  steps with positive probability against all strategies of player 1, in contradiction with player 1 being almost-sure winning for always synchronizing in  $T$ . Hence such a strategy  $\sigma_{\text{safe}}$  exists and we conclude the proof by observing that  $\sigma_{\text{safe}}$  is sure winning for always synchronizing in  $T$ .  $\square$

The following lemma generalizes to games a result that holds for MDPs [41, Section 5.1.2].

LEMMA 12. [42] *In stochastic games, we have  $\langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{event}}(T) = \langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{event}}(T) \cup \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(T)$ .*

PROOF. The inclusions  $\langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{event}}(T) \subseteq \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{event}}(T)$  and  $\langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(T) \subseteq \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{event}}(T)$  follow from the definitions (Section 2). For the converse, consider an initial distribution  $d_0$  from which player 1 has an almost-sure winning strategy for eventually synchronizing in  $T$ . Towards contradiction, assume that  $d_0 \notin \langle\langle 1 \rangle\rangle_{\text{sure}}^{\text{event}}(T) \cup \langle\langle 1 \rangle\rangle_{\text{almost}}^{\text{weakly}}(T)$ . Then for all player-1 strategies  $\sigma$ , there exist player-2 strategies  $\tau_e$  and  $\tau_w$  such that  $\mathcal{G}_{d_0}^{\sigma, \tau_e}$  is not sure eventually synchronizing in  $T$ , and  $\mathcal{G}_{d_0}^{\sigma, \tau_w}$  is not almost-sure weakly synchronizing in  $T$ . Let  $\tau$  be the strategy playing the superposition of  $\frac{1}{2}\tau_e$  and  $\frac{1}{2}\tau_w$ . Consider  $\tau_w$  and following the definitions, there exists  $\varepsilon_w > 0$  and  $i^* \geq 0$  such that



$\mathcal{G}_i^{\sigma, \tau}(T) < 1 - \varepsilon_w$  for all  $i \geq i^*$ . Moreover, considering  $\tau_c$ , we have  $\mathcal{G}_i^{\sigma, \tau}(T) < 1 - \frac{\eta_0}{2} \cdot \eta^{i^*}$  for all  $i \leq i^*$ . For  $\varepsilon = \min\{\varepsilon_w, \frac{\eta_0}{2} \cdot \eta^{i^*}\} > 0$ , we get  $\mathcal{G}_i^{\sigma, \tau}(T) < 1 - \varepsilon$  for all  $i \geq 0$ , in contradiction to player 1 being almost-sure winning for eventually synchronizing in  $T$ .  $\square$

The reduction presented in the proof of Theorem 2 also shows PSPACE-hardness for almost-sure eventually synchronizing in deterministic games.

**LEMMA 13.** *Let  $\mathcal{G}$  be a stochastic game. Given a target set  $T$ , an initial distribution  $d_0$  is almost-sure winning for strongly synchronizing in  $T$  if and only if  $d_0$  is almost-sure winning for the coBüchi objective  $\diamond \square T$ .*

**PROOF.** First, if player 1 is almost-sure winning for the coBüchi objective  $\diamond \square T$  in  $\mathcal{G}$ , then there exists a memoryless winning strategy  $\sigma_{as}$  for player 1, which is such that all states in the (reachable) end-components of the MDP obtained from the game  $\mathcal{G}$  after fixing the strategy  $\sigma_{as}$  are in  $T$  [19]. It is then easy to show that the strategy  $\sigma_{as}$  is almost-sure winning for strongly synchronizing in  $T$ .

For the converse direction, if player 1 is not almost-sure winning for the coBüchi objective  $\diamond \square T$  in  $\mathcal{G}$ , then there exists a strategy  $\tau$  for player 2, which we can assume to be memoryless [16], such that for all player-1 strategies  $\sigma$  we have  $\Pr_{d_0}^{\sigma, \tau}(\diamond \square T) < 1$ . Hence in the MDP  $\mathcal{G}_\tau$  obtained from  $\mathcal{G}$  by fixing the strategy  $\tau$ , player 1 is not almost-sure winning for the coBüchi objective  $\diamond \square T$ , which is equivalent to saying that in  $\mathcal{G}_\tau$  no player-1 strategy is almost-sure winning for strongly synchronizing in  $T$  [25, Lemma 27].  $\square$

We note that in deterministic games, the sure and almost-sure winning modes coincide for state-based objectives, thus it follows from Lemma 13 that for strongly synchronizing the sure and almost-sure winning modes coincide in deterministic games.

We summarize the results of Section 4 for almost-sure synchronizing.

**THEOREM 4.** *The membership problem for almost-sure always and strongly synchronizing can be solved in polynomial time, and pure memoryless strategies are sufficient for player 1.*

*The membership problem for almost-sure eventually and weakly synchronizing is PSPACE-complete, and pure counting strategies are sufficient for player 1. Infinite memory is necessary in general.*

## 5 CONCLUSION

Stochastic games with synchronization objectives combine stochasticity with the presence of an adversary and a flavour of imperfect information, which together tend to bring undecidability [33, 37]. The form of imperfect information in these games differs from the traditional setting where the strategy of player 1 is uniform (the same action is played in all states) [4, 18]. Here, player 1 can see the local state of the game, but needs to enforce a global objective defined on state distributions, which are not visible to player 1. Beyond decidability, it is perhaps surprising that the membership problem for games is no harder than for MDPs (PSPACE-complete), although the proof techniques are significantly more involved, mainly due to the presence of an adversary, and the lack of determinacy.

The main question raised by this model is whether it is possible to extend it with a form of communication, while remaining decidable. This would bring us closer to a wide range of applications in synthetic biology [36, 45] and chemical reaction networks [12].

## ACKNOWLEDGMENTS

The author is grateful to Matthias Függer for pointing out relevant references in the literature on synthetic biology, and to Mahsa Shirmohammadi and Marie van den Bogaard for preliminary discussions about games with synchronization objectives and for inspiring the example of Figure 3.

## REFERENCES

- [1] E. Ábrahám, E. Bartocci, B. Bonakdarpour, and O. Dobe. 2020. Probabilistic Hyperproperties with Nondeterminism. In *Proc. of ATVA: Automated Technology for Verification and Analysis (LNCS 12302)*. Springer, 518–534.
- [2] M. Agrawal, S. Akshay, B. Genest, and P. S. Thiagarajan. 2012. Approximate Verification of the Symbolic Dynamics of Markov Chains. In *Proc. of LICS: Logic in Computer Science*. IEEE, 55–64.
- [3] S. Akshay, B. Genest, and N. Vyas. 2018. Distribution-based objectives for Markov Decision Processes. In *Proc. of LICS: Logic in Computer Science*. ACM, 36–45.
- [4] C. Baier, M. Gröber, and N. Bertrand. 2012. Probabilistic  $\omega$ -automata. *J. ACM* 59, 1 (2012), 1:1–1:52.
- [5] C. Baier and J.-P. Katoen. 2008. *Principles of Model Checking*. MIT.
- [6] N. Bertrand, M. Dewaskar, B. Genest, and H. Gimbert. 2017. Controlling a Population. In *Proc. of CONCUR: Concurrency Theory (LIPIcs, Vol. 85)*. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 12:1–12:16.
- [7] N. Bertrand, B. Genest, and H. Gimbert. 2017. Qualitative Determinacy and Decidability of Stochastic Games with Signals. *J. ACM* 64, 5 (2017), 33:1–33:48.
- [8] J. C. Bradfield and C. Stirling. 2007. Modal mu-calculi. In *Handbook of Modal Logic*. Studies in logic and practical reasoning, Vol. 3. North-Holland, 721–756.
- [9] J. C. Bradfield and I. Walukiewicz. 2018. The mu-calculus and Model Checking. In *Handbook of Model Checking*. Springer, 871–919.
- [10] Tomáš Brázdil, Václav Brozek, Vojtech Forejt, and Antonín Kucera. 2006. Stochastic Games with Branching-Time Winning Objectives. In *Proc. of LICS: Logic in Computer Science*. IEEE Computer Society, 349–358.
- [11] J. R. Büchi. 1962. On a decision method in restricted second order arithmetic. In *Proc. of International Congress of Logic, Methodology and Philosophical Science 1960*. Stanford University Press, 1–11.
- [12] L. Cardelli, M. Kwiatkowska, and L. Laurenti. 2018. Programming discrete distributions with chemical reaction networks. *Nat. Comput.* 17, 1 (2018), 131–145.
- [13] K. Chatterjee. 2007. *Stochastic  $\omega$ -regular Games*. Ph.D. Dissertation. University of California, Berkeley.
- [14] K. Chatterjee, L. Doyen, T. A. Henzinger, and J.-F. Raskin. 2007. Algorithms for Omega-regular Games of Imperfect Information. *Logical Methods in Computer Science* 3, 3:4 (2007).
- [15] K. Chatterjee and T. A. Henzinger. 2012. A survey of stochastic  $\omega$ -regular games. *J. Comput. System Sci.* 78, 2 (2012), 394–413.
- [16] K. Chatterjee, M. Jurdzinski, and T. A. Henzinger. 2004. Quantitative stochastic parity games. In *Proc. of SODA: Symposium on Discrete Algorithms*. SIAM, 121–130.
- [17] A. Church. 1963. Logic, arithmetics, and automata. In *Proc. of International Congress of Mathematicians, 1962*. Institut Mittag-Leffler, 23–35.
- [18] T. Colcombet, N. Fijalkow, and P. Ohlmann. 2020. Controlling a Random Population. In *Proc. of FoSSaCS: Foundations of Software Science and Computation Structures (LNCS 12077)*. Springer, 119–135.
- [19] C. Courcoubetis and M. Yannakakis. 1995. The Complexity of Probabilistic Verification. *J. ACM* 42, 4 (1995), 857–907.
- [20] I. D. Couzin. 2009. Collective cognition in animal groups. *Trends in cognitive sciences* 13, 1 (2009), 36–43.
- [21] I. D. Couzin, J. Krause, N. R. Franks, and S. A. Levin. 2005. Effective leadership and decision-making in animal groups on the move. *Nature* 433 (2005), 513–516.
- [22] L. de Alfaro and T. A. Henzinger. 2000. Concurrent Omega-Regular Games. In *Proc. of LICS: Logic in Computer Science*. IEEE, 141–154.
- [23] L. de Alfaro, T. A. Henzinger, and O. Kupferman. 2007. Concurrent reachability games. *Theoretical Computer Science* 386, 3 (2007), 188–217.
- [24] R. Dimitrova, B. Finkbeiner, and H. Torfah. 2020. Probabilistic Hyperproperties of Markov Decision Processes. In *Proc. of ATVA: Automated Technology for Verification and Analysis (LNCS 12302)*. Springer, 484–500.
- [25] L. Doyen, T. Massart, and M. Shirmohammadi. 2019. The Complexity of Synchronizing Markov Decision Processes. *J. Comput. System Sci.* 100 (2019), 96–129.
- [26] M. Elowitz and S. Leibler. 2000. A synthetic oscillatory network of transcriptional regulators. *Nature* 403, 335–338 (2000).
- [27] J. Esparza. 2014. Keeping a Crowd Safe: On the Complexity of Parameterized Verification (Invited Talk). In *Proc. of STACS: Symposium on Theoretical Aspects of Computer Science (LIPIcs, Vol. 25)*. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 1–10.

- [28] V. V. Isaeva. 2012. Self-organization in biological systems. *Biology Bulletin of the Russian Academy of Sciences* 39 (2012), 110–118.
- [29] P. Jancar and Z. Sawa. 2007. A note on emptiness for alternating finite automata with a one-letter alphabet. *Inf. Process. Lett.* 104, 5 (2007), 164–167.
- [30] R. Kannan. 1992. Lattice translates of a polytope and the Frobenius problem. *Combinatorica* 12 (1992), 161–177.
- [31] D. König. 1936. *Theorie der endlichen und unendlichen Graphen*. Akademische Verlagsgesellschaft, Leipzig.
- [32] V. A. Korthikanti, M. Viswanathan, G. Agha, and Y. Kwon. 2010. Reasoning about MDPs as Transformers of Probability Distributions. In *Proc. of QEST: Quantitative Evaluation of Systems*. IEEE Computer Society, 199–208.
- [33] O. Madani, S. Hanks, and A. Condon. 2003. On the undecidability of probabilistic planning and related stochastic optimization problems. *Artif. Intell.* 147, 1-2 (2003), 5–34.
- [34] D. A. Martin. 1998. The determinacy of Blackwell games. *The Journal of Symbolic Logic* 63, 4 (1998), 1565–1581.
- [35] C. J. Myers. 2016. *Engineering genetic circuits*. CRC Press.
- [36] A. A. K. Nielsen, B. S. Der, J. Singh, P. Vaidyanathan, V. Paralanov, E. A. Strychalski, D. Ross, D. Densmore, and C. A. Voigt. 2016. Genetic circuit design automation. *Science* 352(6281), aac7341 (2016).
- [37] A. Paz. 1971. *Introduction to probabilistic automata*. Academic Press.
- [38] A. Pnueli and R. Rosner. 1990. Distributed Reactive Systems are hard to Synthesize. In *Proc. of FOCS: Foundation of Computer Science*. 746–757.
- [39] John H. Reif. 1984. The complexity of two-player games of incomplete information. *J. Comput. System Sci.* 29, 2 (1984), 274–301.
- [40] S. Schewe. 2014. Distributed synthesis is simply undecidable. *Inf. Process. Lett.* 114, 4 (2014), 203–207.
- [41] M. Shirmohammadi. 2014. *Qualitative analysis of probabilistic synchronizing systems*. Ph.D. Dissertation. Université Libre de Bruxelles.
- [42] M. Shirmohammadi and M. van den Bogaard. 2018. Personal communication.
- [43] W. Thomas. 1997. Languages, Automata, and Logic. In *Handbook of Formal Languages*. Vol. 3, Beyond Words. Springer, Chapter 7, 389–455.
- [44] M. Y. Vardi. 1985. Automatic Verification of Probabilistic Concurrent Finite-State Programs. In *Proc. of FOCS: Foundations of Computer Science*. IEEE Computer Society, 327–338.
- [45] O. Vo, H.-M. Lee, and D. Na. 2019. Synthetic Bacteria for Therapeutics. *Journal of Microbiology and Biotechnology* 29, 6 (2019), 845–855.