



A survey of stochastic ω -regular games[☆]

Krishnendu Chatterjee^{a,*}, Thomas A. Henzinger^{a,b}

^a IST Austria (Institute of Science and Technology Austria), Austria

^b EPFL, Switzerland

ARTICLE INFO

Article history:

Received 18 November 2005

Received in revised form 7 November 2007

Accepted 6 May 2011

Available online 12 May 2011

Keywords:

Game theory

Stochastic games

ω -Regular objectives

ABSTRACT

We summarize classical and recent results about two-player games played on graphs with ω -regular objectives. These games have applications in the verification and synthesis of reactive systems. Important distinctions are whether a graph game is turn-based or concurrent; deterministic or stochastic; zero-sum or not. We cluster known results and open problems according to these classifications.

© 2011 Elsevier Inc. All rights reserved.

1. Introduction

We consider nonterminating two-player perfect-information games played on graphs. A game proceeds for an infinite number of rounds. The state of a game is a vertex of a graph. In each round, the state changes along an edge of the graph to a successor vertex. Thus, the outcome of the game being played for an infinite number of rounds, is an infinite path through the graph. We consider boolean objectives for the two players: for each player, the resulting infinite path is either winning or losing. The winning sets of paths are assumed to be ω -regular [87]. Depending on how the winning sets are specified, we distinguish between parity, Rabin, Streett, and Müller games, as well as some subclasses thereof. Depending on whether or not the two players have complementary winning sets, we distinguish between zero-sum and nonzero-sum games. Depending on the structure of the graph, we distinguish between turn-based and concurrent games. In *turn-based* games, the graph is partitioned into player-1 states and player-2 states: in player-1 states, player 1 chooses the successor vertex; and in player-2 states, player 2 chooses the successor vertex. In *concurrent* games, in every round both players choose simultaneously and independently from a set of available moves, and the combination of both choices determines the successor vertex. Finally, we distinguish between deterministic and stochastic games: in *stochastic* games, in every round the players' moves determine a probability distribution on the possible successor vertices, instead of determining a unique successor vertex.

These games play a central role in several areas of computer science. One important application arises when the vertices and edges of a graph represent the states and transitions of a reactive system, and the two players represent controllable versus uncontrollable decisions during the execution of the system. The *synthesis* problem (or *control* problem) for reactive systems asks for the construction of a winning strategy in the corresponding graph game. This problem was first posed independently by Alonzo Church [28] and Richard Büchi [8] in settings that can be reduced to *turn-based deterministic* games with ω -regular objectives. The problem was solved independently by Michael Rabin using logics on trees [81], and by Büchi

[☆] This research was supported in part by the ONR grant N00014-02-1-0671, by the AFOSR MURI grant F49620-00-1-0327, and by the NSF grants CCR-9988172, CCR-0085949, and CCR-0225610.

* Corresponding author.

E-mail address: krishnendu.chatterjee@ist.ac.at (K. Chatterjee).

and Lawrence Landweber using a more game-theoretic approach [9]; it was later resolved using improved methods [53,73] and in different application contexts [82,79]. Game-theoretic formulations have proved useful not only for synthesis, but also for the modeling [44,1], refinement [56], verification [40,2], testing [7], and compatibility checking [35,36] of reactive systems. The use of ω -regular objectives is natural in these application contexts. This is because the winning conditions of the games arise from requirements specifications for reactive systems, and the ω -regular sets of infinite paths provide an important and robust paradigm for such specifications [69]. However, both the restriction to deterministic games and the restriction to turn-based games are limiting in some respects: probabilistic transitions are useful to model uncertain behavior that is not strictly adversarial [88,31], and concurrent choice is useful to model certain forms of synchronous interaction between reactive systems [39,41]. The resulting *concurrent stochastic games* have long been familiar to game theorists and mathematicians, sometimes under the name of *competitive Markov decision processes* [52]. But they have usually been studied in nonalgorithmic contexts for very general kinds of objectives, such as Borel sets of winning paths [70,71]. Only recently has the algorithmic study of turn-based stochastic games and of concurrent games, with the interesting and well-behaved class of ω -regular objectives, caught the attention of computer scientists [29,37,34,42,15,16]. We attempt to summarize the resulting theory.

The central computational problem about a game is the question of whether a player has a strategy for winning the game. However, in stochastic graph games there are several degrees of “winning”: we may ask if a player has a strategy that ensures a winning outcome of the game, no matter how the other player resolves her choices (this is called *sure winning*); or we may ask if a player has a strategy that achieves a winning outcome of the game with probability 1 (*almost-sure winning*); or we may ask if the maximal probability with which a player can win is 1 in the limit (*limit-sure winning*), where the maximal probability in the limit is defined as the supremum over all possible strategies of the infimum over all adversarial strategies. While all three notions of winning coincide for turn-based deterministic games [70], and almost-sure winning coincides with limit-sure winning for turn-based stochastic games [24], all three notions are different for concurrent games, even in the deterministic case [37]. This is because for concurrent games, strategies that use randomization are more powerful than pure (i.e., nonrandomized) strategies. The computation of sure winning, almost-sure winning, and limit-sure winning states is called the *qualitative analysis* of graph games. This is in contrast to the *quantitative analysis*, which asks for computing for each state the maximal probability with which a player can win in the limit, even if that limit is less than 1. For a fixed player, the limit probability is called the sup-inf value, or the optimal value, or simply the *value* of the game at a state. A strategy that achieves the optimal value is an *optimal strategy*, and a strategy that ensures one of the three ways of winning, is a *sure (almost-sure; limit-sure) winning strategy*. Concurrent graph games are more difficult than turn-based graph games for several reasons. In concurrent games, optimal strategies may not exist, but for every real $\varepsilon > 0$, there may be a strategy that guarantees a winning outcome with a probability that lies within ε of the optimal value. Moreover, ε -optimal and limit-sure winning strategies may require infinite memory about the history of a game in order to prescribe the next move of a player. By contrast, in certain special cases—for example, in the case of turn-based stochastic games with parity objectives—optimal and winning strategies require neither randomization nor memory; such pure memoryless strategies can be implemented by control maps from states to moves. We refer to the randomization and memory requirements of strategies as the “structural complexity” of strategies.

So far we have discussed the notion of “winning” for a fixed player. In *zero-sum* games, the sets of winning paths for the two players are complementary. A zero-sum game that has a winning strategy for one of the two players at every vertex is called *determined*. There are two kinds of determinacy results for graph games. First, the turn-based deterministic games have a qualitative determinacy, namely, *determinacy for sure winning*: in every state of the game graph, one of the two players has a sure winning strategy [70]. Second, the turn-based stochastic games and the concurrent games have a quantitative determinacy, that is, *determinacy for optimal values*: in every state, the optimal values for both players add up to 1 [71]. Both the sure-winning determinacy result and the optimal-value determinacy results hold for all Borel objectives. The sure-winning determinacy for turn-based deterministic games with Borel objectives was established by Donald Martin [70]; the optimal-value determinacy for Borel objectives was established again by Martin [71] for a very general class of games called *Blackwell games*, which include all games we consider in this survey. For concurrent games, however, there is no determinacy for sure winning: even if a concurrent game is deterministic (i.e., nonstochastic) and the objectives are simple (e.g., single-step reachability), neither player may have a strategy for sure winning [37]. There is also no determinacy for sure winning for turn-based probabilistic games, even with reachability objectives. Determinacy is useful for solving zero-sum games: it allows us to switch, whenever convenient, between the dual views of the two players while computing the sure winning states of a game, or the optimal values.

In *nonzero-sum* games, both players may be winning. In this case, the notion of rational behavior of the players is captured by Nash equilibria: a pair of strategies for the two players is a Nash equilibrium if neither player can increase her payoff by unilaterally switching her strategy [61]. In our games, the payoff is the probability of winning. While for turn-based games, Nash equilibria are known to exist for all Borel objectives [27], for concurrent games the situation is again more complicated. A pair of strategies for the two players is an ε -Nash equilibrium, for $\varepsilon > 0$, if neither player can increase her payoff by at least ε by switching strategy. Even in the simple case of reachability objectives, no Nash equilibria, but only ε -Nash equilibria (for all $\varepsilon > 0$) may exist for concurrent games [27]. Many of the questions in this area are still open.

Our survey is organized as follows. Sections 2–6 contain the pertinent definitions: game graphs, strategies, objectives, winning, determinacy, and equilibria. The subsequent three sections summarize results: Section 7 on turn-based zero-sum games; Section 8 on concurrent zero-sum games; and Section 9 on nonzero-sum games. These three sections can be read

independently. We focus on two kinds of results: the algorithmic complexity of computing the winning states, the optimal values, and the Nash equilibria of a game; and the structural complexity required of winning strategies, of optimal strategies, and of equilibrium strategies. Of course, there are many types of closely related results and related games that are not discussed in this survey. We had to make several more or less arbitrary decisions where to draw the line about what material to include. In particular, the survey is restricted to games played on *finite* graphs, with *qualitative* (i.e., boolean) objectives, where both players have *perfect information* about the state of a game. Infinite-state games, games with quantitative objectives (such as mean-payoff games), and partial-information games are not treated in this survey, but a few pointers to the literature on these kinds of games are given in the last section.

2. Game graphs

We first define turn-based game graphs, and then the more general class of concurrent game graphs. We start with some preliminary notation. For a finite set A , a probability distribution on A is a function $\delta: A \rightarrow [0, 1]$ such that $\sum_{a \in A} \delta(a) = 1$. We write $\text{Supp}(\delta) = \{a \in A \mid \delta(a) > 0\}$ for the support set of δ . We denote the set of probability distributions on A by $\text{Dist}(A)$.

2.1. Turn-based probabilistic game graphs

We consider several classes of turn-based games, namely, two-player turn-based probabilistic games ($2\frac{1}{2}$ -player games), two-player turn-based deterministic games (2-player games), and Markov decision processes ($1\frac{1}{2}$ -player games).

A *turn-based probabilistic game graph* (or $2\frac{1}{2}$ -player game graph) $G = ((S, E), (S_1, S_2, S_P), \delta)$ consists of a directed graph (S, E) , a partition of the vertex set S into three subsets $S_1, S_2, S_P \subseteq S$, and a probabilistic transition function $\delta: S_P \rightarrow \text{Dist}(S)$. The vertices in S are called *states*. The state space S is finite. The states in S_1 are *player-1* states; the states in S_2 are *player-2* states; and the states in S_P are *probabilistic* states. For all states $s \in S$, we define $E(s) = \{t \in S \mid (s, t) \in E\}$ to be the set of possible successor states. We require that $E(s) \neq \emptyset$ for every nonprobabilistic state $s \in S_1 \cup S_2$, and that $E(s) = \text{Supp}(\delta(s))$ for every probabilistic state $s \in S_P$. At player-1 states $s \in S_1$, player 1 chooses a successor state from $E(s)$; at player-2 states $s \in S_2$, player 2 chooses a successor state from $E(s)$; and at probabilistic states $s \in S_P$, a successor state is chosen according to the probability distribution $\delta(s)$.

The turn-based *deterministic* game graphs (or 2-player game graphs) are the special case of the $2\frac{1}{2}$ -player game graphs with $S_P = \emptyset$. The *Markov decision processes* (MDPs for short; or $1\frac{1}{2}$ -player game graphs) are the special case of the $2\frac{1}{2}$ -player game graphs with either $S_1 = \emptyset$ or $S_2 = \emptyset$. We refer to the MDPs with $S_2 = \emptyset$ as *player-1* MDPs, and to the MDPs with $S_1 = \emptyset$ as *player-2* MDPs. A game graph that is both deterministic and an MDP is called a *transition system* (or 1-player game graph): a player-1 transition system has only player-1 states; a player-2 transition system has only player-2 states.

2.2. Concurrent game graphs

A *concurrent game graph* $G = (S, A, \Gamma_1, \Gamma_2, \delta)$ consists of the following components:

- A finite state space S .
- A finite set A of moves.
- Two move assignments $\Gamma_1, \Gamma_2: S \rightarrow 2^A \setminus \emptyset$. For $i \in \{1, 2\}$, the player- i move assignment Γ_i associates with every state $s \in S$ a nonempty set $\Gamma_i(s) \subseteq A$ of moves available to player i at state s .
- A probabilistic transition function $\delta: S \times A \times A \rightarrow \text{Dist}(S)$. At every state $s \in S$, player 1 chooses a move $a_1 \in \Gamma_1(s)$, and simultaneously and independently player 2 chooses a move $a_2 \in \Gamma_2(s)$. A successor state is then chosen according to the probability distribution $\delta(s, a_1, a_2)$.

For all states $s \in S$ and all moves $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$, we define $\text{Succ}(s, a_1, a_2) = \text{Supp}(\delta(s, a_1, a_2))$ to be the set of possible successor states of s when the moves a_1 and a_2 are chosen. For a concurrent game graph, we define the set of edges as $E = \{(s, t) \in S \times S \mid (\exists a_1 \in \Gamma_1(s))(\exists a_2 \in \Gamma_2(s))(t \in \text{Succ}(s, a_1, a_2))\}$, and as with turn-based game graphs, we write $E(s) = \{t \mid (s, t) \in E\}$ for the set of possible successors of a state $s \in S$.

We distinguish the following special classes of concurrent game graphs. The concurrent game graph G is *deterministic* if $|\text{Succ}(s, a_1, a_2)| = 1$ for all states $s \in S$ and all moves $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$. A state $s \in S$ is a *turn-based* state if there exists a player $i \in \{1, 2\}$ such that $|\Gamma_i(s)| = 1$; that is, player i has no choice of moves at s . If $|\Gamma_2(s)| = 1$, then s is a *player-1* turn-based state; and if $|\Gamma_1(s)| = 1$, then s is a *player-2* turn-based state. The concurrent game graph G is *turn-based* if every state in S is a turn-based state. Note that the turn-based concurrent game graphs are equivalent to the turn-based probabilistic game graphs: to obtain a $2\frac{1}{2}$ -player game graph from a turn-based concurrent game graph G , for every player- i turn-based state s of G , where $i \in \{1, 2\}$, introduce $|\Gamma_i(s)|$ many probabilistic successor states of s . Moreover, the concurrent game graphs that are both turn-based and deterministic are equivalent to the 2-player game graphs.

To measure the complexity of algorithms and problems, we need to define the size of a game graph. We do this for the case that all transition probabilities can be specified as rational numbers. Then the size of a concurrent game graph G is equal to the size of the probabilistic transition function δ , that is, $|G| = \sum_{s \in S} \sum_{a_1 \in \Gamma_1(s)} \sum_{a_2 \in \Gamma_2(s)} \sum_{t \in S} |\delta(s, a_1, a_2)(t)|$, where

$|\delta(s, a_1, a_2)(t)|$ denotes the space required to specify a rational probability value. The size of a turn-based probabilistic game graph G is equal to the sum of its state space and edges, and the size of the probabilistic transition function δ , that is, $|G| = |S| + |E| + \sum_{s \in S} \sum_{t \in S} |\delta(s)(t)|$, where $|\delta(s)(t)|$ denotes the space required to specify a rational probability value.

3. Strategies

When choosing their moves, the players follow recipes that are called strategies. We define strategies both for $2\frac{1}{2}$ -player game graphs and for concurrent game graphs. On a concurrent game graph, the players choose moves from a set A of moves, while on a $2\frac{1}{2}$ -player game graph, they choose successor states from a set S of states. Hence, for $2\frac{1}{2}$ -player game graphs, we define the set of moves as $A = S$. For $2\frac{1}{2}$ -player game graphs, a player-1 strategy prescribes the moves that player 1 chooses at the player-1 states S_1 , and a player-2 strategy prescribes the moves that player 2 chooses at the player-2 states S_2 . For concurrent game graphs, both players choose moves at every state, and hence for concurrent game graphs, we define the sets of player-1 states and player-2 states as $S_1 = S_2 = S$.

Consider a game graph G . A *player-1 strategy* on G is a function $\sigma : S^* \cdot S_1 \rightarrow \text{Dist}(A)$ that assigns to every nonempty finite sequence $\vec{s} \in S^* \cdot S_1$ of states ending in a player-1 state, a probability distribution $\sigma(\vec{s})$ over the moves A . By following the strategy σ , whenever the history of a game played on G is \vec{s} , then player 1 chooses the next move according to the probability distribution $\sigma(\vec{s})$. A strategy must prescribe only available moves. Hence, for all state sequences $\vec{s}' \in S^*$ and all states $s \in S_1$, if $\sigma(\vec{s}' \cdot s)(a) > 0$, then the following condition must hold: $a \in E(s)$ for $2\frac{1}{2}$ -player game graphs G , and $a \in \Gamma_1(s)$ for concurrent game graphs G . Symmetrically, a *player-2 strategy* on G is a function $\pi : S^* \cdot S_2 \rightarrow \text{Dist}(A)$ such that if $\pi(\vec{s}' \cdot s)(a) > 0$, then $a \in E(s)$ for $2\frac{1}{2}$ -player game graphs G , and $a \in \Gamma_2(s)$ for concurrent game graphs G . We write Σ for the set of player-1 strategies, and Π for the player-2 strategies on G . Note that $|\Pi| = 1$ if G is a player-1 MDP, and $|\Sigma| = 1$ if G is a player-2 MDP.

3.1. Types of strategies

We classify strategies according to their use of randomization and memory.

Use of randomization. Strategies that do not use randomization are called pure. A player-1 strategy σ is *pure* (or *deterministic*) if for all state sequences $\vec{s} \in S^* \cdot S_1$, there exists a move $a \in A$ such that $\sigma(\vec{s})(a) = 1$. The pure strategies for player 2 are defined analogously. We denote by Σ^P the set of pure player-1 strategies, and by Π^P the set of pure player-2 strategies. A strategy that is not necessarily pure is called *randomized*.

Use of memory. Strategies in general require memory to remember the history of a game. The following alternative definition of strategies makes this explicit. Let M be a set called *memory*. A player-1 strategy $\sigma = (\sigma_u, \sigma_n)$ can be specified as a pair of functions: a memory-update function $\sigma_u : S \times M \rightarrow M$, which given the current state of the game and the memory, updates the memory with information about the current state; and a next-move function $\sigma_n : S_1 \times M \rightarrow \text{Dist}(A)$, which given the current state and the memory, prescribes the next move of the player. The player-1 strategy σ is *finite-memory* if the memory M is a finite set; and the strategy σ is *memoryless* (or *positional*) if the memory M is singleton, i.e., $|M| = 1$. A finite-memory strategy remembers only a finite amount of information about the infinitely many different possible histories of the game; a memoryless strategy is independent of the history of the game and depends only on the current state of the game. Note that a memoryless player-1 strategy can be represented as a function $\sigma : S_1 \rightarrow \text{Dist}(A)$. We denote by Σ^F the set of finite-memory player-1 strategies, and by Σ^M the set of memoryless player-1 strategies. The finite-memory player-2 strategies Π^F and the memoryless player-2 strategies Π^M are defined analogously.

A pure finite-memory strategy is a pure strategy that is finite-memory; we write $\Sigma^{PF} = \Sigma^P \cap \Sigma^F$ for the pure finite-memory player-1 strategies, and Π^{PF} for the corresponding player-2 strategies. A pure memoryless strategy is a pure strategy that is memoryless. The pure memoryless strategies use neither randomization nor memory; they are the simplest strategies we consider. Note that a pure memoryless player-1 strategy can be represented as a function $\sigma : S_1 \rightarrow A$. We write $\Sigma^{PM} = \Sigma^P \cap \Sigma^M$ for the pure memoryless player-1 strategies, and Π^{PM} for the corresponding class of simple player-2 strategies.

3.2. Probability space induced by a strategy profile

A *path* of the game graph G is an infinite sequence $\omega = \langle s_0, s_1, s_2, \dots \rangle$ of states in S such that $(s_k, s_{k+1}) \in E$ for all $k \geq 0$. We denote the set of paths of G by Ω . We refer to a pair $(\sigma, \pi) \in \Sigma \times \Pi$ of strategies, one for each player, as a *strategy profile*. Once a starting state $s \in S$ and a strategy profile (σ, π) are fixed, the result of the game is a random walk in G , denoted $\rho_s^{\sigma, \pi}$, which generates a path in Ω .

Given a finite sequence $\vec{s} = \langle s_0, s_1, \dots, s_k \rangle$ of states in S , the *cone* defined by \vec{s} is the set $\text{Cone}(\vec{s}) = \{\langle s'_0, s'_1, s'_2, \dots \rangle \in \Omega \mid (\forall 0 \leq i \leq k)(s_i = s'_i)\}$ of paths with prefix \vec{s} . Let $\mathcal{C} = \{\text{Cone}(\vec{s}) \mid \vec{s} \in S^*\}$ be the set of all cones. The cones in \mathcal{C} are the basic open sets in the Cantor topology on the set Ω of paths. Let \mathcal{F} be the Borel σ -field generated by \mathcal{C} , that is, let \mathcal{F} be the smallest set such that (i) $\mathcal{C} \subseteq \mathcal{F}$ and (ii) \mathcal{F} is closed under complementation, countable union, and countable

intersection. Then (Ω, \mathcal{F}) is a σ -algebra. Given a strategy profile $(\sigma, \pi) \in \Sigma \times \Pi$ and a state $s \in S$, we define the function $\mu_s^{\sigma, \pi} : \mathcal{C} \rightarrow [0, 1]$ as follows: for all nonempty state sequences $\vec{u} = \vec{u}' \cdot u \in S^+$ and all states $t \in S$,

$$\begin{aligned}\mu_s^{\sigma, \pi}(\text{Cone}(\epsilon)) &= \mu_s^{\sigma, \pi}(\Omega) = 1; \\ \mu_s^{\sigma, \pi}(\text{Cone}(t)) &= \begin{cases} 1 & \text{if } t = s, \\ 0 & \text{otherwise;} \end{cases} \\ \mu_s^{\sigma, \pi}(\text{Cone}(\vec{u} \cdot t)) &= \mu_s^{\sigma, \pi}(\vec{u}) \cdot \sum_{a_1 \in \Gamma_1(u), a_2 \in \Gamma_2(u)} \delta(u, a_1, a_2)(t) \cdot \sigma(\vec{u})(a_1) \cdot \pi(\vec{u})(a_2).\end{aligned}$$

The function $\mu_s^{\sigma, \pi}$ is a measure on \mathcal{C} , and hence there is a unique extension of $\mu_s^{\sigma, \pi}$ to a probability measure on \mathcal{F} . We denote this probability measure on \mathcal{F} , which is induced by the strategies σ and π and the starting state s , by $\text{Pr}_s^{\sigma, \pi}$. Then $(\Omega, \mathcal{F}, \text{Pr}_s^{\sigma, \pi})$ is a probability space. An *event* Φ in this space is a measurable set of paths, that is, $\Phi \in \mathcal{F}$. The probability $\text{Pr}_s^{\sigma, \pi}(\Phi)$ of an event $\Phi \in \mathcal{F}$ is the probability that the random walk $\rho_s^{\sigma, \pi}$ generates a path in Φ .

Possible outcomes of a strategy profile. Consider two strategies $\sigma \in \Sigma$ and $\pi \in \Pi$ on a game graph G , and let $\omega = \langle s_0, s_1, s_2, \dots \rangle$ be a path of G . The path ω is (σ, π) -possible for a $2\frac{1}{2}$ -player game graph G if for every $k \geq 0$ the following two conditions hold: if $s_k \in S_1$, then $\sigma(s_0 s_1 \dots s_k)(s_{k+1}) > 0$; and if $s_k \in S_2$, then $\pi(s_0 s_1 \dots s_k)(s_{k+1}) > 0$. The path ω is (σ, π) -possible for a concurrent game graph G if for every $k \geq 0$, there exist moves $a_1 \in \Gamma_1(s_k)$ and $a_2 \in \Gamma_2(s_k)$ for the two players such that $\sigma(s_0 s_1 \dots s_k)(a_1) > 0$ and $\pi(s_0 s_1 \dots s_k)(a_2) > 0$ and $s_{k+1} \in \text{Succ}(s_k, a_1, a_2)$. Given a state $s \in S$ and a strategy profile $(\sigma, \pi) \in \Sigma \times \Pi$, we denote by $\text{Outcome}(s, \sigma, \pi) \subseteq \Omega$ the set of (σ, π) -possible paths whose first state is s . Note that $\text{Outcome}(s, \sigma, \pi)$ is a probability-1 event, that is, $\text{Pr}_s^{\sigma, \pi}(\text{Outcome}(s, \sigma, \pi)) = 1$.

Fixing a strategy. Given a game graph G and a player-1 strategy $\sigma \in \Sigma$, we write G_σ for the game played on G under the constraint that player 1 follows the strategy σ . Analogously, given G and a player-2 strategy $\pi \in \Pi$, we write G_π for the game played on G under the constraint that player 2 follows the strategy π . Observe that for a $2\frac{1}{2}$ -player game graph G or a concurrent game graph G , and a memoryless player-1 strategy $\sigma \in \Sigma$, the result G_σ is a player-2 MDP. Similarly, for a player-2 MDP G and a memoryless player-2 strategy $\pi \in \Pi$, the result G_π is a Markov chain. Hence, if G is a $2\frac{1}{2}$ -player game graph or a concurrent game graph, and the two players follow memoryless strategies σ and π , then the result $G_{\sigma, \pi} = (G_\sigma)_\pi$ is a Markov chain. Also the following observation will be used later. Given a game graph G and a strategy in $\Sigma \cup \Pi$ with finite memory M , the strategy can be interpreted as a *memoryless* strategy in the synchronous product $G \times M$ of the game graph G with the memory M .

4. Objectives

An *objective* Φ for a game graph G is a set of paths, that is, $\Phi \subseteq \Omega$. A player-1 objective $\Phi \subseteq \Omega$ specifies the set of paths that are *winning for player 1*, and a player-2 objective $\Psi \subseteq \Omega$ specifies the set of paths that are *winning for player 2*: player 1 wins the game played on the graph G with the objectives Φ and Ψ iff the path that results from playing the game lies in Φ , and player 2 wins if that path lies in Ψ . In the case of zero-sum games, the objectives of the two players are strictly competitive, that is, $\Psi = \Omega \setminus \Phi$. A general class of objectives are the Borel objectives. A *Borel objective* $\Phi \subseteq \Omega$ is a Borel set in the Cantor topology on the set Ω of paths, that is, $\Phi \in \mathcal{F}$ for the Borel σ -field \mathcal{F} defined in Subsection 3.2. Throughout this survey, we limit ourselves to Borel objectives, often without explicitly using the adjective “Borel.” An important subclass of the Borel objectives are the ω -regular objectives, which lie in the first $2\frac{1}{2}$ levels of the Borel hierarchy (i.e., in the intersection of Σ_3 and Π_3). The ω -regular objectives are of special interest for the verification and synthesis of reactive systems [69]. In particular, the following specifications of winning conditions for the players define ω -regular objectives, and subclasses thereof [87].

Reachability and safety objectives. A *reachability specification* for the game graph G is a set $T \subseteq S$ of states, called target states. The reachability specification T requires that some state in T be visited. Thus, the reachability specification T defines the set $\text{Reach}(T) = \{(s_0, s_1, s_2, \dots) \in \Omega \mid (\exists k \geq 0)(s_k \in T)\}$ of winning paths; this set is called a reachability objective. A *safety specification* for G is likewise a set $U \subseteq S$ of states; they are called safe states. The safety specification U requires that only states in U be visited. Formally, the safety objective defined by U is the set $\text{Safe}(U) = \{(s_0, s_1, \dots) \in \Omega \mid (\forall k \geq 0)(s_k \in U)\}$ of winning paths. Note that reachability and safety are dual objectives: $\text{Safe}(U) = \Omega \setminus \text{Reach}(S \setminus U)$.

Büchi and coBüchi objectives. A *Büchi specification* for G is a set $B \subseteq S$ of states, which are called Büchi states. The Büchi specification B requires that some state in B be visited infinitely often. For a path $\omega = \langle s_0, s_1, s_2, \dots \rangle$, we write $\text{Inf}(\omega) = \{s \in S \mid s_k = s \text{ for infinitely many } k \geq 0\}$ for the set of states that occur infinitely often in ω . Thus, the Büchi objective defined by B is the set $\text{Büchi}(B) = \{\omega \in \Omega \mid \text{Inf}(\omega) \cap B \neq \emptyset\}$ of winning paths. The dual of a Büchi specification is a *coBüchi specification* $C \subseteq S$, which specifies a set of so-called coBüchi states. The coBüchi specification C requires that the states outside C be visited only finitely often. Formally, the coBüchi objective defined by C is the set $\text{coBüchi}(C) = \{\omega \in \Omega \mid \text{Inf}(\omega) \subseteq C\}$ of winning paths. Note that $\text{coBüchi}(C) = \Omega \setminus \text{Büchi}(S \setminus C)$. It is also worth noting that reachability and safety objectives

can be turned into both Büchi and coBüchi objectives, by slightly modifying the game graph. For example, if the graph G' results from G by turning every target state $s \in T$ into a sink state (so that $E(s) = \{s\}$), then a game played on G with the reachability objective $\text{Reach}(T)$ is equivalent to a game played on G' with the Büchi objective $\text{Büchi}(T)$.

Rabin and Streett objectives. We now move to boolean combinations of Büchi and coBüchi objectives. A *Rabin specification* for the game graph G is a finite set $R = \{(E_1, F_1), \dots, (E_d, F_d)\}$ of pairs of sets of states, that is, $E_j \subseteq S$ and $F_j \subseteq S$ for all $1 \leq j \leq d$. The pairs in R are called Rabin pairs. We assume without loss of generality that $\bigcup_{1 \leq j \leq d} (E_j \cup F_j) = S$. The Rabin specification R requires that for some Rabin pair $1 \leq j \leq d$, all states in the left-hand set E_j be visited finitely often, and some state in the right-hand set F_j be visited infinitely often. Thus, the Rabin objective defined by R is the set $\text{Rabin}(R) = \{\omega \in \Omega \mid (\exists 1 \leq j \leq d)(\text{Inf}(\omega) \cap E_j = \emptyset \wedge \text{Inf}(\omega) \cap F_j \neq \emptyset)\}$ of winning paths. Note that the coBüchi objective $\text{coBüchi}(C)$ is equal to the single-pair Rabin objective $\text{Rabin}(\{(\emptyset, C), (S, S)\})$, and the Büchi objective $\text{Büchi}(B)$ is equal to the two-pair Rabin objective $\text{Rabin}(\{(\emptyset, B), (S, S)\})$. The complements of Rabin objectives are called Streett objectives. A *Streett specification* for G is likewise a set $W = \{(E_1, F_1), \dots, (E_d, F_d)\}$ of pairs of set of states $E_j \subseteq S$ and $F_j \subseteq S$ such that $\bigcup_{1 \leq j \leq d} (E_j \cup F_j) = \emptyset$. The pairs in W are called Streett pairs. The Streett specification W requires that for every Streett pair $1 \leq j \leq d$, if some state in the right-hand set F_j is visited infinitely often, then some state in the left-hand set E_j is visited infinitely often. Formally, the Streett objective defined by W is the set $\text{Streett}(W) = \{\omega \in \Omega \mid (\forall 1 \leq j \leq d)(\text{Inf}(\omega) \cap E_j \neq \emptyset \vee \text{Inf}(\omega) \cap F_j = \emptyset)\}$ of winning paths. Note that $\text{Streett}(W) = \Omega \setminus \text{Rabin}(W)$.

Parity objectives. A *parity specification* for G consists of a nonnegative integer d and a function $p : S \rightarrow \{0, 1, 2, \dots, 2d\}$, which assigns to every state of G an integer between 0 and $2d$. For a state $s \in S$, the value $p(s)$ is called the *priority* of S . We assume without loss of generality that $p^{-1}(j) \neq \emptyset$ for all $0 < j \leq 2d$; this implies that a parity specification is completely specified by the priority function p (and d does not need to be specified explicitly). The positive integer $2d + 1$ is referred to as the number of priorities of p . The parity specification p requires that the minimum priority of all states that are visited infinitely often, is even. Formally, the parity objective defined by p is the set $\text{Parity}(p) = \{\omega \in \Omega \mid \min\{p(s) \mid s \in \text{Inf}(\omega)\} \text{ is even}\}$ of winning paths. Note that for a parity objective $\text{Parity}(p)$, the complementary objective $\Omega \setminus \text{Parity}(p)$ is again a parity objective: $\Omega \setminus \text{Parity}(p) = \text{Parity}(p + 1)$, where the priority function $p + 1$ is defined by $(p + 1)(s) = p(s) + 1$ for all states $s \in S$ (if $p^{-1}(0) = \emptyset$, then use $p - 1$ instead of $p + 1$). This self-duality of parity objectives is often convenient when solving games. It is also worth noting that the Büchi objectives are parity objectives with two priorities (let $p^{-1}(0) = B$ and $p^{-1}(1) = S \setminus B$), and the coBüchi objectives are parity objectives with three priorities (let $p^{-1}(0) = \emptyset$ and $p^{-1}(1) = S \setminus C$ and $p^{-1}(2) = C$).

Parity objectives are also called *Rabin-chain* objectives, as they are a special case of Rabin objectives [87]: if the sets of a Rabin specification $R = \{(E_1, F_1), \dots, (E_d, F_d)\}$ form a chain $E_1 \subsetneq F_1 \subsetneq E_2 \subsetneq F_2 \subsetneq \dots \subsetneq E_d \subsetneq F_d$, then $\text{Rabin}(R) = \text{Parity}(p)$ for the priority function $p : S \rightarrow \{0, 1, \dots, 2d\}$ that for every $1 \leq j \leq d$ assigns to each state in $E_j \setminus F_{j-1}$ the priority $2j - 1$, and to each state in $F_j \setminus E_j$ the priority $2j$, where $F_0 = \emptyset$. Conversely, given a priority function $p : S \rightarrow \{0, 1, \dots, 2d\}$, we can construct a chain $E_1 \subsetneq F_1 \subsetneq \dots \subsetneq E_{d+1} \subsetneq F_{d+1}$ of $d + 1$ Rabin pairs such that $\text{Parity}(p) = \text{Rabin}(\{(E_1, F_1), \dots, (E_{d+1}, F_{d+1})\})$ as follows: let $E_1 = \emptyset$ and $F_1 = p^{-1}(0)$, and for all $1 \leq j \leq d + 1$, let $E_j = F_{j-1} \cup p^{-1}(2j - 3)$ and $F_j = E_j \cup p^{-1}(2j - 2)$. Hence, the parity objectives are a subclass of the Rabin objectives that is closed under complementation. It follows that every parity objective is both a Rabin objective and a Streett objective. The parity objectives are of special interest, because every ω -regular objective can be turned into a parity objective by modifying the game graph (take the synchronous product of the game graph with a deterministic parity automaton that accepts the ω -regular objective) [75].

Müller objectives. The most general form for defining ω -regular objectives are Müller specifications. A *Müller specification* for the game graph G is a set $M \subseteq 2^S$ of sets of states. The sets in M are called Müller sets. The Müller specification M requires that the set of states that are visited infinitely often is one of the Müller sets. Formally, the Müller specification M defines the Müller objective $\text{Müller}(M) = \{\omega \in \Omega \mid \text{Inf}(\omega) \in M\}$. Note that Rabin and Streett objectives are special cases of Müller objectives.

5. Game values

For a state s and an objective Φ for player 1, the maximal probability with which player 1 can ensure that Φ holds from s is the value of the game at s for player 1. Formally, given a game graph G with objectives Φ for player 1 and Ψ for player 2, we define the *value* functions Val_1^G and Val_2^G for the players 1 and 2, respectively, as follows: for every state $s \in S$,

$$\text{Val}_1^G(\Phi)(s) = \sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi);$$

$$\text{Val}_2^G(\Psi)(s) = \sup_{\pi \in \Pi} \inf_{\sigma \in \Sigma} \text{Pr}_s^{\sigma, \pi}(\Psi).$$

A player-1 strategy $\sigma \in \Sigma$ is *optimal* from a state $s \in S$ for the objective Φ if

$$\text{Val}_1^G(\Phi)(s) = \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi).$$

The player-1 strategy σ is ε -optimal, for $\varepsilon \geq 0$, from the state s for the objective Φ if

$$\text{Val}_1^G(\Phi)(s) \leq \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi) + \varepsilon.$$

Note that an optimal strategy is ε -optimal for $\varepsilon = 0$. We refer to player-1 strategies as (ε) -optimal for Φ if they are (ε) -optimal from all states in S for the objective Φ . The optimal and ε -optimal strategies for player 2 are defined analogously. Computing values, optimal, and ε -optimal strategies is referred to as the *quantitative analysis* of games.

Sure, almost-sure, and limit-sure winning. Given a game graph G with an objective Φ for player 1, a player-1 strategy $\sigma \in \Sigma$ is a *sure winning strategy* from a state $s \in S$ if for every player-2 strategy $\pi \in \Pi$, $\text{Outcome}(s, \sigma, \pi) \subseteq \Phi$; that is, all possible outcomes lie in Φ when player 1 plays according to the strategy σ . The player-1 strategy σ is an *almost-sure winning strategy* from the state s for the objective Φ if for every player-2 strategy $\pi \in \Pi$, $\text{Pr}_s^{\sigma, \pi}(\Phi) = 1$; that is, the path that results from playing the game lies in Φ with probability 1 when player 1 plays according to the strategy σ . A family $\Sigma^X \subseteq \Sigma$ of player-1 strategies is *limit-sure winning* from the state s for the objective Φ if $\sup_{\sigma \in \Sigma^X} \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi) = 1$; that is, for every $\varepsilon > 0$, the family Σ^X contains a player-1 strategy σ such that the path results from playing the game lies in Φ with probability at least $1 - \varepsilon$ when player 1 plays according to the strategy σ . The sure winning, almost-sure winning, and limit-sure winning strategies for player 2 are defined analogously.

For a game graph G and an objective Φ , the *sure winning set* $\text{Sure}_1^G(\Phi) \subseteq S$ for player 1 is the set of states from which player 1 has a sure winning strategy for Φ . Similarly, the *almost-sure winning set* $\text{Almost}_1^G(\Phi) \subseteq S$ for player 1 is the set of states from which player 1 has an almost-sure winning strategy for Φ , and the *limit-sure winning set* $\text{Limit}_1^G(\Phi) \subseteq S$ for player 1 is the set of states from which player 1 has a family of limit-sure winning strategies for Φ . We refer to the states in $\text{Sure}_1^G(\Phi)$ ($\text{Almost}_1^G(\Phi)$; $\text{Limit}_1^G(\Phi)$) as *sure* (*almost-sure*; *limit-sure winning*) for player 1. The sure, almost-sure, and limit-sure winning sets $\text{Sure}_2^G(\Psi)$, $\text{Almost}_2^G(\Psi)$, and $\text{Limit}_2^G(\Psi)$ for player 2 with objective Ψ are defined analogously. It follows from the definitions that for all $2\frac{1}{2}$ -player and concurrent game graphs, and all objectives Φ and Ψ for the two players, both $\text{Sure}_1^G(\Phi) \subseteq \text{Almost}_1^G(\Phi) \subseteq \text{Limit}_1^G(\Phi)$ and $\text{Sure}_2^G(\Psi) \subseteq \text{Almost}_2^G(\Psi) \subseteq \text{Limit}_2^G(\Psi)$. Computing sure winning, almost-sure winning, and limit-sure winning states and strategies is referred to as the *qualitative analysis* of games.

Sufficiency of a family of strategies for winning. Let $X \in \{P, M, F, PM, PF\}$, and consider the family $\Sigma^X \subseteq \Sigma$ of special strategies for player 1. The family Σ^X of player-1 strategies *suffices* with respect to an objective Φ on a class \mathcal{G} of game graphs for

- *sure winning*, if for every game graph $G \in \mathcal{G}$ and every state $s \in \text{Sure}_1^G(\Phi)$, there is a player-1 strategy $\sigma \in \Sigma^X$ such that for every player-2 strategy $\pi \in \Pi$, we have $\text{Outcome}(s, \sigma, \pi) \subseteq \Phi$;
- *almost-sure winning*, if for every game graph $G \in \mathcal{G}$ and every state $s \in \text{Almost}_1^G(\Phi)$, there is a player-1 strategy $\sigma \in \Sigma^X$ such that for every player-2 strategy $\pi \in \Pi$, we have $\text{Pr}_s^{\sigma, \pi}(\Phi) = 1$;
- *limit-sure winning*, if for every game graph $G \in \mathcal{G}$ and every state $s \in \text{Limit}_1^G(\Phi)$, we have $\sup_{\sigma \in \Sigma^X} \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi) = 1$;
- *optimality*, if for every game graph $G \in \mathcal{G}$ and every state $s \in S$, there is a player-1 strategy $\sigma \in \Sigma^X$ such that $\text{Val}_1^G(\Phi)(s) = \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi)$;
- ε -*optimality*, for $\varepsilon \geq 0$, if for every game graph $G \in \mathcal{G}$ and every state $s \in S$, there is a player-1 strategy $\sigma \in \Sigma^X$ such that $\text{Val}_1^G(\Phi)(s) \leq \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi) + \varepsilon$.

For sure winning, $1\frac{1}{2}$ -player and $2\frac{1}{2}$ -player games coincide with 2-player (turn-based deterministic) games where the random player (who chooses the successor at the probabilistic states) is interpreted as an adversary, i.e., as player 2. This is formalized in the following proposition.

Proposition 1. *If a family Σ^X of player-1 strategies suffices for sure winning with respect to an objective Φ on all 2-player game graphs, then the family Σ^X suffices for sure winning with respect to Φ also on all $1\frac{1}{2}$ -player and $2\frac{1}{2}$ -player game graphs.*

The following proposition states that randomized strategies are not necessary for sure winning.

Proposition 2. *If a family Σ^X of player-1 strategies suffices for sure winning with respect to an objective Φ on all concurrent game graphs, then the family $\Sigma^X \cap \Sigma^P$ of pure strategies suffices for sure winning with respect to Φ on all concurrent game graphs.*

6. Rational behavior in games

6.1. Zero-sum games and determinacy

A game (G, Φ, Ψ) consists of a game graph G , a player-1 objective Φ , and a player-2 objective Ψ . The game is *zero-sum* if the two objectives are complementary, that is, $\Psi = \Omega \setminus \Phi$. Rational behavior in zero-sum games is captured by the notions of optimal and ε -optimal strategies. The key result, which establishes the existence of equilibria in zero-sum games,

is determinacy: the zero-sum game (G, Φ, Ψ) is *determinate* if at every state, the sum of the values for the two players is 1; that is, $\text{Val}_1^G(\Phi)(s) + \text{Val}_2^G(\Psi)(s) = 1$ for all $s \in S$. Determinacy implies the following equality: for every state $s \in S$,

$$\sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \text{Pr}_s^{\sigma, \pi}(\Phi) = \inf_{\pi \in \Pi} \sup_{\sigma \in \Sigma} \text{Pr}_s^{\sigma, \pi}(\Phi).$$

Determinacy also guarantees the existence of ε -optimal strategies, for all $\varepsilon > 0$, for both players from every state. A deep result by Martin [71] established determinacy for all concurrent game graphs with zero-sum Borel objectives; see Theorem 4.

A nonstochastic notion of determinacy is *sure determinacy*: the zero-sum game (G, Φ, Ψ) is *sure determinate* if every state is in the sure-winning set of one of the players; that is, $\text{Sure}_1^G(\Phi) \cup \text{Sure}_2^G(\Psi) = S$. Since $\text{Sure}_1^G(\Phi) \cap \text{Sure}_2^G(\Psi) = \emptyset$ for zero-sum games, if a game is sure determinate, then the sure-winning sets of the two players partition the state space. Martin [70] established sure determinacy for all turn-based deterministic game graphs with zero-sum Borel objectives. Sure determinacy, however, does not hold for turn-based probabilistic game graphs (and thus not for concurrent game graphs). In these graphs, there may be states from which neither player can win surely even if one player has a reachability objective, and the other player the complementary safety objective.

6.2. Nonzero-sum games and Nash equilibria

Rational behavior in nonzero-sum games is characterized by the notion of Nash equilibrium. Intuitively, a strategy profile is a Nash equilibrium if no player can gain by unilaterally deviating from her strategy. Formally, for a game (G, Φ, Ψ) and $\varepsilon \geq 0$, a strategy profile $(\sigma^*, \pi^*) \in \Sigma \times \Pi$ is an ε -Nash equilibrium if the following two conditions hold:

$$\begin{aligned} \sup_{\sigma \in \Sigma} \text{Pr}_s^{\sigma, \pi^*}(\Phi) &\leq \text{Pr}_s^{\sigma^*, \pi^*}(\Phi) + \varepsilon; \\ \sup_{\pi \in \Pi} \text{Pr}_s^{\sigma^*, \pi}(\Psi) &\leq \text{Pr}_s^{\sigma^*, \pi^*}(\Psi) + \varepsilon. \end{aligned}$$

A *Nash equilibrium* is an ε -Nash equilibrium with $\varepsilon = 0$.

Sufficiency of a family of strategies for Nash equilibria. Let $X \in \{P, M, F, PM, PF\}$, and consider the families $\Sigma^X \subseteq \Sigma$ and $\Pi^X \subseteq \Pi$ of special player-1 and player-2 strategies. Given $\varepsilon \geq 0$, the families Σ^X and Π^X suffice for the existence of ε -Nash equilibria with respect to objectives Φ and Ψ on a class \mathcal{G} of game graphs, if for every game graph $G \in \mathcal{G}$, there exists an ε -Nash equilibrium (σ^*, π^*) for the game (G, Φ, Ψ) such that $\sigma^* \in \Sigma^X$ and $\pi^* \in \Pi^X$. The sufficiency condition for the existence of Nash equilibria is obtained by taking $\varepsilon = 0$.

Multiplayer games. The notion of Nash equilibrium generalizes to more than two players. In an n -player turn-based probabilistic game graph, the state space S is partitioned into n sets S_1, \dots, S_n —one for each of the players—and a set S_p of probabilistic states (the special case of $2\frac{1}{2}$ -player games is obtained for $n = 2$). An n -player concurrent game graph contains a move assignment $\Gamma_i: S \rightarrow 2^A \setminus \emptyset$ for each player $i \in \{1, \dots, n\}$, and the transition function has the type $\delta: S \times A^n \rightarrow \text{Dist}(S)$. A n -player game $(G, \Phi_1, \dots, \Phi_n)$ consists of an n -player game graph G and an objective Φ_i for each player $i \in \{1, \dots, n\}$. In n -player games, a strategy profile $\bar{\sigma} = (\sigma_1, \dots, \sigma_n)$ is a tuple of strategies—one for each player. Given a strategy profile $\bar{\sigma} = (\sigma_1, \dots, \sigma_n)$ and a strategy σ'_i for player i , we write $\bar{\sigma} \upharpoonright \sigma'_i$ for the strategy profile that results from $\bar{\sigma}$ by replacing the i -component σ_i with σ'_i . For an n -player game $(G, \Phi_1, \dots, \Phi_n)$ and $\varepsilon \geq 0$, a strategy profile $\bar{\sigma}^*$ is an ε -Nash equilibrium if for all $1 \leq i \leq n$,

$$\sup_{\sigma'_i \in \Sigma_i} \text{Pr}_s^{\bar{\sigma}^* \upharpoonright \sigma'_i}(\Phi_i) \leq \text{Pr}_s^{\bar{\sigma}^*}(\Phi_i) + \varepsilon.$$

As before, a *Nash equilibrium* is an ε -Nash equilibrium with $\varepsilon = 0$. The sufficiency of a family of strategies for n -player Nash equilibria is defined as in the case of two players.

7. Turn-based zero-sum games

Reduction of games. A key method to obtain results in game theory is the principle of *reduction*. Several results on complex games are obtained via reduction to simpler games: simpler in terms of objectives, simpler in terms of game graphs (e.g., concurrent games to turn-based games), or simpler in terms of winning criteria (e.g., quantitative to qualitative winning criteria). For example, Martin [71] reduced concurrent games with Borel objectives to 2-player games with Borel objectives (and larger state spaces). The reduction together with the Borel determinacy of 2-player games [70] established the Borel determinacy of concurrent games. In this section, we present a reduction of $2\frac{1}{2}$ -player games with Rabin objectives and almost-sure winning to 2-player games with Rabin objectives (and sure winning) [15]. In Section 8, we will present a reduction of concurrent parity games with quantitative winning criteria to solving multiple parity subgames with qualitative winning criteria [16].

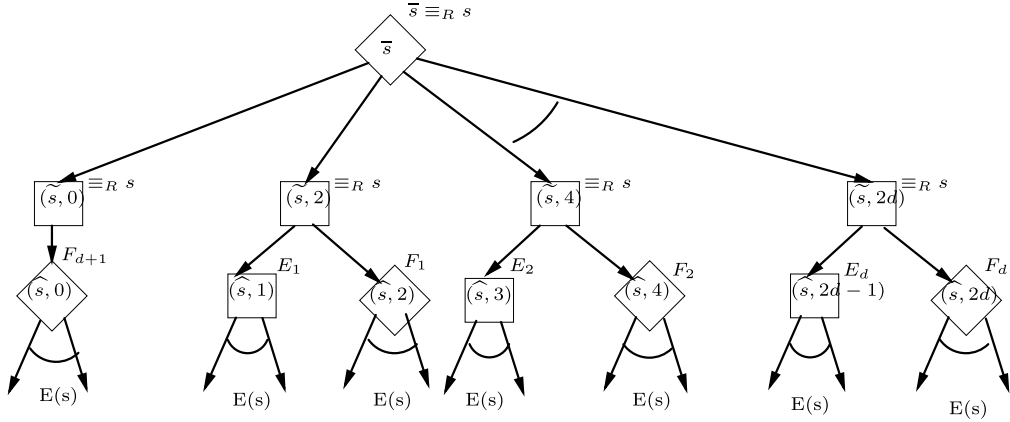


Fig. 1. Gadget for the reduction of $2\frac{1}{2}$ -player Rabin games to 2-player Rabin games.

7.1. Strategy complexity

We determine the strategy complexity of turn-based zero-sum games with ω -regular objectives.

A local reduction of $2\frac{1}{2}$ -player to 2-player Rabin games. Given a $2\frac{1}{2}$ -player game graph $G = ((S, E), (S_1, S_2, S_P), \delta)$, and a set $R = \{(E_1, F_1), \dots, (E_d, F_d)\}$ of Rabin pairs, we construct a 2-player game graph $\bar{G} = ((\bar{S}, \bar{E}), (\bar{S}_1, \bar{S}_2), \bar{\delta})$ together with an extended set $\bar{R} = R \cup \{(E_{d+1}, F_{d+1})\}$ of Rabin pairs. For two states $s, s' \in S$, we write $s \equiv_R s'$ to denote that for all $1 \leq j \leq d$, both $s \in E_j$ iff $s' \in E_j$, and $s \in F_j$ iff $s' \in F_j$. The construction of \bar{G} is as follows. For every nonprobabilistic state $s \in (S_1 \cup S_2)$, there is a corresponding state $\bar{s} \in \bar{S}$ such that (1) $\bar{s} \in \bar{S}_1$ iff $s \in S_1$, and (2) $\bar{s} \equiv_R s$, and (3) $(\bar{s}, \bar{t}) \in \bar{E}$ iff $(s, t) \in E$. Every probabilistic state $s \in S_P$ is replaced by the gadget shown in Fig. 1. In the figure, diamond-shaped states are player-2 states (in \bar{S}_2), and square-shaped states are player-1 states (in \bar{S}_1). From the state \bar{s} (with $\bar{s} \equiv_R s$), the players play the following three-step game in \bar{G} . First, in state \bar{s} player 2 chooses a successor $(\bar{s}, 2k)$, for $k \in \{0, 1, \dots, d\}$. For every state $(\bar{s}, 2k)$, we have $(\bar{s}, 2k) \equiv_R s$. Second, for $k \geq 1$, in state $(\bar{s}, 2k)$ player 1 chooses from two successors: state $(\bar{s}, 2k-1)$ with $(\bar{s}, 2k-1) \in E_k$, or state $(\bar{s}, 2k)$ with $(\bar{s}, 2k) \in F_k$. The state $(\bar{s}, 0)$ has only one successor, $(\bar{s}, 0)$. Third, in every state (\bar{s}, j) the choice is between all states \bar{t} such that $(s, t) \in E$, and it belongs to player 1 if k is odd, and to player 2 if k is even. The set E_{d+1} is empty; the set F_{d+1} contains all states $(\bar{s}, 0)$.

Given a set $\bar{U} \subseteq \bar{S}$ of states in the 2-player game graph \bar{G} , we denote by $U = \{s \in S \mid \bar{s} \in \bar{U}\}$ the set of corresponding states in the original $2\frac{1}{2}$ -player game graph G . Similarly, given a pure memoryless player-1 strategy $\bar{\sigma}$ on the 2-player game graph \bar{G} , the corresponding (pure memoryless) player-1 strategy σ on the $2\frac{1}{2}$ -player game graph G is defined for all $s \in S_1$ by $\sigma(s) = t$ iff $\sigma(\bar{s}) = \bar{t}$.

Lemma 1. (See [15].) For every $2\frac{1}{2}$ -player game graph G and every set R of Rabin pairs, let $\bar{U}_1 = \text{Sure}_1^{\bar{G}}(\text{Rabin}(\bar{R}))$ and $\bar{U}_2 = \text{Sure}_2^{\bar{G}}(\Omega \setminus \text{Rabin}(\bar{R}))$. The following two assertions hold:

- (i) $U_1 = \text{Almost}_1^G(\text{Rabin}(R)) = \text{Limit}_1^G(\text{Rabin}(R)) = S \setminus U_2$.
- (ii) If $\bar{\sigma}$ is a pure memoryless sure winning strategy for player 1 from the states \bar{U}_1 in \bar{G} , then σ is an almost-sure winning strategy for player 1 from the states U_1 in G .

Note that Lemma 1 states that for all $2\frac{1}{2}$ -player game graphs G with Rabin objectives Φ , the almost-sure winning set $\text{Almost}_1^G(\Phi)$ and the limit-sure winning set $\text{Limit}_1^G(\Phi)$ coincide. Since all Müller objectives can be reduced to Rabin objectives [75,87], it follows that for all $2\frac{1}{2}$ -player game graphs with Müller objectives Φ , we also have $\text{Almost}_1^G(\Phi) = \text{Limit}_1^G(\Phi)$.

From almost-sure winning to optimal strategies. The analysis in [15] also showed that for $2\frac{1}{2}$ -player games with Müller objectives, optimal strategies are no more complex than almost-sure winning strategies. We sketch the idea behind the result.

A set $U \subseteq S$ of states is δ -live if for every nonprobabilistic state $u \in U \cap (S_1 \cup S_2)$, there exists a state $t \in U$ with $(u, t) \in E$. Let $\text{Bnd}(U) = \{s \in (U \cap S_P) \mid (\exists t \in E(s))(t \notin U)\}$ be the set of *boundary* probabilistic states of U , which have an edge out of U . We define the transformation $\text{Win}_1^G(U)$ of the game graph G as follows: all states outside U are removed, and every boundary probabilistic state $s \in \text{Bnd}(U)$ is converted into an absorbing state (i.e., $E(s) = \{s\}$) that is sure winning

Table 1

The strategy complexity of turn-based zero-sum games with ω -regular objectives, where Σ^{PM} denotes the family of pure memoryless strategies, Σ^{PF} denotes the family of pure finite-memory strategies, and Σ^M denotes the family of randomized memoryless strategies.

Objective	1-player	$1\frac{1}{2}$ -player	2-player	$2\frac{1}{2}$ -player
Reachability/Safety	Σ^{PM}	Σ^{PM}	Σ^{PM}	Σ^{PM}
Parity	Σ^{PM}	Σ^{PM}	Σ^{PM}	Σ^{PM}
Rabin	Σ^{PM}	Σ^{PM}	Σ^{PM}	Σ^{PM}
Streett	Σ^{PF} / Σ^M	Σ^{PF} / Σ^M	Σ^{PF}	Σ^{PF}
Müller	Σ^{PF} / Σ^M	Σ^{PF} / Σ^M	Σ^{PF}	Σ^{PF}

for player 1. Observe that if U is δ -live, then $\text{Win}_1^G(U)$ is a game graph. For a Müller objective Φ and a real number $r \in \mathbb{R}$, the value class $VC(\Phi, r) = \{s \in S \mid \text{Val}_1^G(\Phi)(s) = r\}$ is the set of states with value r for player 1. For all $2\frac{1}{2}$ -player game graphs, all Müller objectives Φ , and all reals $r > 0$, the value class $VC(\Phi, r)$ is δ -live. The following lemma establishes a connection between value classes, the transformation Win_1^G , and almost-sure winning.

Lemma 2. (See [15].) For all $2\frac{1}{2}$ -player game graphs, all Müller objectives Φ , and all reals $r > 0$, all states of the game graph $\text{Win}_1^G(VC(\Phi, r))$ are almost-sure winning for player 1 for the objective Φ .

Lemma 3. (See [15].) Consider a $2\frac{1}{2}$ -player game graph G and a Müller objective Φ . Let σ be a player-1 strategy on G such that for all reals $r > 0$, the strategy σ is almost-sure winning on the game graph $\text{Win}_1^G(VC(\Phi, r))$ for the objective Φ . Then σ is an optimal strategy on G for Φ .

Lemmas 2 and 3 imply the following theorem.

Theorem 1. (See [15].) If a family Σ^C of player-1 strategies suffices for almost-sure winning with respect to a Müller objective Φ on all $2\frac{1}{2}$ -player game graphs, then Σ^C suffices for optimality with respect to Φ on all $2\frac{1}{2}$ -player game graphs.

Summary of results. Martin [70] proved that for 2-player zero-sum games with Borel objectives, the sure winning sets for the two players partition the state space. Moreover, the pure strategies suffice for sure winning in 2-player games with Borel objectives; however, in general sure winning strategies require infinite memory. Gurevich and Harrington [53] showed that for 2-player games with ω -regular objectives, pure finite-memory strategies suffice for sure winning. They based the construction of pure finite-memory sure winning strategies on a data structure, which is called *latest appearance record* (LAR) and remembers the order of the latest appearances of the states in a play. Emerson and Jutla [47] established that for 2-player games with Rabin objectives, pure memoryless strategies suffice for sure winning. The results of Dziembowski et al. [45] give precise memory requirements for pure strategies in 2-player games with ω -regular objectives: their construction of strategies is based on a tree representation of a Müller objective, called the *Zielonka tree*, which was introduced in [95].

Condon [29] showed that pure memoryless strategies suffice for optimality in $2\frac{1}{2}$ -player games with reachability and safety objectives. For $2\frac{1}{2}$ -player games with parity objectives, the existence of pure memoryless optimal strategies was proved in [25,72,94]. Lemma 1 and the result of Emerson and Jutla [47] establish that pure memoryless strategies suffice for almost-sure winning in $2\frac{1}{2}$ -player games with Rabin objectives. From Theorem 1 it follows that pure memoryless strategies suffice for optimality in $2\frac{1}{2}$ -player games with Rabin objectives. This implies the existence of pure finite-memory optimal strategies for $2\frac{1}{2}$ -player games with Streett or Müller objectives, because every Müller and Streett objective can be specified as a parity objective [75,87]. The precise memory bound of [45] for pure strategies can be extended from 2-player game graphs to $2\frac{1}{2}$ -player game graphs [11]. In the special case of $1\frac{1}{2}$ -player games (MDPs), randomized memoryless optimal strategies exist for Müller and Streett objectives [14].

All results are summarized in Theorem 2 and also shown in Table 1.

Theorem 2. The following assertions hold.

- (i) (See [70].) The family Σ^P of pure strategies suffices for sure winning with respect to Borel objectives on all 2-player game graphs. Moreover, for all 2-player game graphs G and all Borel objectives Φ , we have $\text{Sure}_1^G(\Phi) = S \setminus \text{Sure}_2^G(S \setminus \Phi)$.
- (ii) (See [53].) The family Σ^{PF} of pure finite-memory strategies suffices for sure winning with respect to Streett and Müller objectives on all 2-player game graphs.
- (iii) (See [47].) The family Σ^{PM} of pure memoryless strategies suffices for sure winning with respect to reachability, safety, parity, and Rabin objectives on all 2-player game graphs.
- (iv) (See [29,25,72,94,15].) The family Σ^{PM} of pure memoryless strategies suffices for optimality with respect to reachability, safety, parity, and Rabin objectives on all $2\frac{1}{2}$ -player game graphs.

Table 2The computational complexity of solving $2\frac{1}{2}$ -player games with ω -regular objectives.

Objective	1-player	$1\frac{1}{2}$ -player	2-player	$2\frac{1}{2}$ -player game graph	
	qualitative analysis	quantitative analysis	qualitative analysis	qualitative analysis	quantitative analysis
Reachability/Safety	PTIME	PTIME	PTIME	PTIME	$\text{NP} \cap \text{coNP}$
Parity	PTIME	PTIME	$\text{NP} \cap \text{coNP}$	$\text{NP} \cap \text{coNP}$	$\text{NP} \cap \text{coNP}$
Rabin	PTIME	PTIME	NP-complete	NP-complete	NP-complete
Streett	PTIME	PTIME	coNP-complete	coNP-complete	coNP-complete
Müller	PTIME	PTIME	PSPACE-compl.	PSPACE-compl.	PSPACE-compl.

- (v) (See [11].) The family Σ^{PF} of pure finite-memory strategies suffices for optimality with respect to Streett and Müller objectives on all $2\frac{1}{2}$ -player game graphs.
- (vi) (See [14].) The family Σ^{PF} of pure finite-memory strategies and the family of Σ^M of randomized memoryless strategies each suffice for optimality with respect to Streett and Müller objectives on all $1\frac{1}{2}$ -player game graphs.

7.2. Computational complexity

We present complexity results for solving turn-based zero-sum games with ω -regular objectives.

The quantitative analysis of $1\frac{1}{2}$ -player games with reachability and safety objectives can be solved by linear programming [29]. For the quantitative analysis of $1\frac{1}{2}$ -player games with Rabin objectives, de Alfaro [32] gave a polynomial-time algorithm. The quantitative analysis of $1\frac{1}{2}$ -player games with Streett objectives can also be achieved in polynomial time [15, 13]. The results of [31] characterize the complexity of solving $1\frac{1}{2}$ -player games with ω -regular objectives that are specified in various forms (e.g., as LTL formulae).

The analysis of 2-player games with reachability objectives is the And-Or graph reachability problem, which is PTIME-complete [4,60]. Emerson and Jutla [47] showed that the solution problem for 2-player games with Rabin objectives is NP-complete, and dually, coNP-complete for Streett objectives. It follows that 2-player games with parity objectives can be decided in $\text{NP} \cap \text{coNP}$. Hunter and Dawar [59] proved that the solution problem for 2-player games with Müller objectives is PSPACE-complete.

From Lemma 1 it follows that the qualitative analysis of $2\frac{1}{2}$ -player games with Rabin objectives can be reduced to the analysis of 2-player games with Rabin objectives. This, together with Proposition 1, gives us results for qualitative analysis of $2\frac{1}{2}$ -player games from the corresponding results for 2-player games. The existence of pure memoryless optimal strategies for $2\frac{1}{2}$ -player games with reachability and safety objectives combined with a polynomial-time algorithm for the quantitative analysis of MDPs establishes that the quantitative analysis of $2\frac{1}{2}$ -player games with reachability objectives lies in $\text{NP} \cap \text{coNP}$ [29]. Similarly, the existence of pure memoryless optimal strategies for $2\frac{1}{2}$ -player games with Rabin objectives combined with a polynomial-time algorithm for solving MDPs with Streett objectives establishes that $2\frac{1}{2}$ -player games with Rabin objectives can be solved in NP. A lower bound of NP-hardness follows from the special case of 2-player games, thus showing that the quantitative analysis of $2\frac{1}{2}$ -player games with Rabin objectives is NP-complete, and dually, coNP-complete for Streett objectives. The results of [12] prove that the quantitative analysis of $2\frac{1}{2}$ -player games with Müller objectives is PSPACE-complete.

All results are summarized in Theorem 3 and shown in Table 2.

Theorem 3. The following assertions hold.

- (i) (See [52,32,13].) Given a player-1 MDP G , the value function $\text{Val}_1^G(\Phi)$ can be computed in polynomial time for reachability, safety, parity, Rabin, Streett, and Müller objectives Φ .
- (ii) (See [4,60,47,59].) Given a 2-player game graph G , the sure winning set $\text{Sure}_1^G(\Phi)$ can be computed in linear time for reachability and safety objectives Φ . Given a 2-player game graph G and a state s , the decision problem whether $s \in \text{Sure}_1^G(\Phi)$ is NP-complete if Φ is a Rabin objective; coNP-complete if Φ is a Streett objective; PSPACE-complete if Φ is a Müller objective; and can be decided in $\text{NP} \cap \text{coNP}$ if Φ is a parity objective.
- (iii) (See [25,15,12].) Given a $2\frac{1}{2}$ -player game graph G , the sure winning set $\text{Sure}_1^G(\Phi)$ can be computed in linear time and the almost-sure winning set $\text{Almost}_1^G(\Phi)$ can be computed in quadratic time for reachability and safety objectives Φ . Given a $2\frac{1}{2}$ -player game graph G and a state s , the decision problems whether $s \in \text{Sure}_1^G(\Phi)$ or $s \in \text{Almost}_1^G(\Phi)$ are NP-complete if Φ is a Rabin objective; coNP-complete if Φ is a Streett objective; PSPACE-complete if Φ is a Müller objective; and can be decided in $\text{NP} \cap \text{coNP}$ if Φ is a parity objective.
- (iv) (See [29,25,15,12].) Given a $2\frac{1}{2}$ -player game graph G , a rational number $r > 0$, and a state s , the decision problem whether $\text{Val}_1^G(\Phi)(s) \geq r$ is NP-complete if Φ is a Rabin objective; coNP-complete if Φ is a Streett objective; PSPACE-complete if Φ is a Müller objective; and can be decided in $\text{NP} \cap \text{coNP}$ if Φ is a reachability, safety or a parity objective.

7.3. Algorithms and open problems

Emerson and Jutla [48] established the equivalence of solving 2-player parity games and μ -calculus model checking. This intriguing connection led to much research attempting to solve 2-player parity games in polynomial time. Alas, the problem is still open.

Algorithms for 2-player parity games. The classical algorithm for solving parity games proceeds by a recursive decomposition of the problem and repeatedly solving games with reachability objectives [73,87]. The running time of the algorithm for games with n states, m edges, and e priorities is $O(n^{e-1} \cdot m)$. Jurdziński [63] gave an improved algorithm to solve parity games based on a notion of ranking functions and progress measures. This algorithm, called the *small-progress measure* algorithm, has a running time of $O((\frac{2n}{e})^{\lfloor \frac{e}{2} \rfloor} \cdot m)$; moreover, there exists a family of games on which the running time of the algorithm is exponential. Another notable algorithm for solving parity games is the *strategy improvement* algorithm [91]. This algorithm iterates local optimizations of pure memoryless strategies which converge to a globally optimal strategy. Though the best known bound for the running time of the strategy improvement algorithm is exponential, it behaves well in practice. In fact, no family of games is known on which more than a polynomial number of local strategy improvements is required. Based on the strategy improvement algorithm, a randomized subexponential-time algorithm (with an expected running time of $O(2^{\sqrt{n \cdot \log n}})$) for solving parity games was presented by Björklund et al. [5]. Recently, Jurdziński et al. [64] gave a deterministic subexponential-time algorithm for solving 2-player games with parity objectives.

Algorithms for $2\frac{1}{2}$ -player reachability games. It should be noted that computing sure winning sets for 2-player parity games can be reduced to computing value functions for $2\frac{1}{2}$ -player reachability games. The reduction can be obtained in two steps: a simple reduction of 2-player parity games to 2-player mean-payoff games was given by Puri [80,62], and 2-player mean-payoff games can be reduced to $2\frac{1}{2}$ -player reachability games [96]. The notable algorithms for the quantitative analysis of $2\frac{1}{2}$ -player reachability games are a strategy improvement algorithm by Condon [30], and based on Condon's algorithm, a randomized subexponential-time algorithm by Ludwig [67]. Ludwig's original algorithm worked on *binary game graphs* (game graphs where each state has at most two outgoing edges), but can be combined with the technique of [5] to obtain a randomized subexponential-time algorithm for all $2\frac{1}{2}$ -player games with reachability objectives [6]. The algorithm of [64] does not generalize in any obvious way to provide a deterministic subexponential-time algorithm for $2\frac{1}{2}$ -player reachability games.

Algorithms for $2\frac{1}{2}$ -player parity games. The notable algorithms for $2\frac{1}{2}$ -player parity games are a strategy improvement algorithm [20] that combines the techniques used by the strategy improvement algorithms for 2-player parity games and for $2\frac{1}{2}$ -player reachability games; and based on the strategy improvement algorithm, a randomized subexponential-time algorithm [20].

Algorithms for Rabin and Streett games. Notable algorithms for 2-player games with Rabin and Streett objectives include the adaptation of the classical algorithm of Zielonka [95] for Müller games specialized to Rabin and Streett games [58]; an algorithm that is based on a reduction to the emptiness problem for weak-alternating automata [66]; a generalization of the small-progress measure algorithm for parity games to Rabin and Streett games [78]; and a generalization of the subexponential-time algorithm for parity games [64] to Rabin and Streett games [23]. The reduction of $2\frac{1}{2}$ -player games with Rabin and Streett objectives for qualitative analysis to 2-player games (presented in Lemma 1) makes all algorithms of 2-player games with Rabin and Streett objectives available for the qualitative analysis of $2\frac{1}{2}$ -player Rabin and Streett games. An algorithm for the quantitative analysis of $2\frac{1}{2}$ -player Rabin and Streett games, which combines the strategy improvement algorithm for $2\frac{1}{2}$ -player reachability games with any algorithm for solving 2-player Rabin and Streett games, is presented in [21].

Open problems. The most important open problems for turn-based zero-sum games are the following:

- (i) a polynomial-time algorithm for computing the sure winning sets of 2-player game graphs with parity objectives;
- (ii) a polynomial-time algorithm for computing the value functions of $2\frac{1}{2}$ -player game graphs with reachability and safety objectives;
- (iii) a polynomial-time algorithm for computing the almost-sure winning sets and the value functions of $2\frac{1}{2}$ -player game graphs with parity objectives.

8. Concurrent zero-sum games

Concurrent games differ considerably from $2\frac{1}{2}$ -player games. For example, in concurrent games with reachability objectives, optimal strategies need not exist. Only ε -optimal strategies, for all $\varepsilon > 0$, are guaranteed to exist [51], and in general they require randomization. In concurrent games with Büchi objectives, in general ε -optimal strategies require both randomization and infinite memory [34]. We start with several examples to illustrate these observations; the examples are adapted from [37,34].

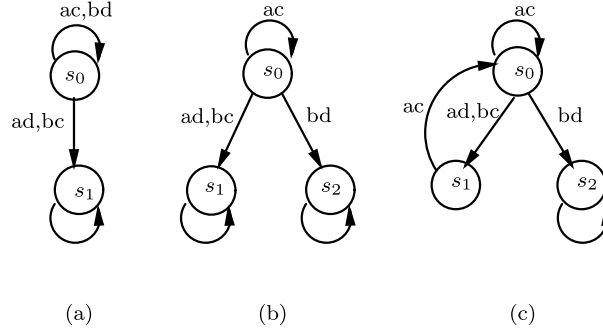


Fig. 2. Examples of concurrent games.

Example 1 (*Almost-sure winning*). Consider the concurrent game graph shown in Fig. 2(a). At the state s_0 , the sets of available moves for player 1 and player 2 are $\{a, b\}$ and $\{c, d\}$, respectively. The transition function at s_0 is defined as follows:

$$\delta(s_0, a, c)(s_0) = \delta(s_0, b, d)(s_0) = 1; \quad \delta(s_0, a, d)(s_1) = \delta(s_0, b, c)(s_1) = 1.$$

The state s_1 is absorbing, where a state s of a concurrent game graph is *absorbing* if for all moves $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$, we have $\delta(s, a_1, a_2)(s) = 1$. The objective for player 1 is to reach the state s_1 ; that is, player 1 has the reachability objective $\text{Reach}(\{s_1\})$.

Consider a pure strategy σ for player 1. Let π be the following strategy for player 2: each time player 1 chooses move a , player 2 chooses move c ; each time player 1 chooses move b , player 2 chooses move d . The path starting in s_0 and resulting from the players following the strategy profile (σ, π) stays in s_0 forever, and never visits the target state s_1 . Hence for every pure strategy for player 1, from s_0 there is a winning counterstrategy for player 2.

Now consider the following randomized memoryless strategy $\sigma_{0.5}$ for player 1: at the state s_0 , player 1 chooses each of the moves a and b with probability $\frac{1}{2}$. For every strategy π for player 2, the random walk starting in s_0 and resulting from the strategy profile $(\sigma_{0.5}, \pi)$ proceeds, in each round, with probability $\frac{1}{2}$ to s_1 , and stays with probability $\frac{1}{2}$ in s_0 . Hence the target state s_1 is reached with probability 1. For every player-2 strategy π , there exists a path $\omega \in \text{Outcome}(s_0, \sigma_{0.5}, \pi)$ that never visits s_1 ; however, the set $\{\omega\}$ has measure 0. Thus, although player 1 cannot win this game with certainty, she can win with probability 1: the state s_0 is not sure winning, but almost-sure winning for player 1.

Example 2 (*Limit-sure winning*). Consider the concurrent game graph shown in Fig. 2(b). The transition function at the state s_0 is defined as follows:

$$\delta(s_0, a, c)(s_0) = 1; \quad \delta(s_0, b, d)(s_2) = 1; \quad \delta(s_0, a, d)(s_1) = \delta(s_0, b, c)(s_1) = 1.$$

The states s_1 and s_2 are absorbing. The objective for player 1 is to reach s_1 ; that is, player 1 has the reachability objective $\text{Reach}(\{s_1\})$.

For any $\varepsilon > 0$, consider the following randomized memoryless strategy σ_ε for player 1: at s_0 , choose move a with probability $1 - \varepsilon$, and move b with probability ε . The game starts at s_0 . In each round in which player 2 chooses move c , the game proceeds to s_1 with probability ε , and stays in s_0 with probability $1 - \varepsilon$. In each round in which player 2 chooses move d , the game proceeds to s_1 with probability $1 - \varepsilon$, and to s_2 with probability ε . Hence, against every strategy π for player 2, given the strategy σ_ε for player 1, the game reaches s_1 with probability at least $1 - \varepsilon$. It follows that for all reals $\varepsilon > 0$, there exists a player-1 strategy σ such that for all player-2 strategies π , $\Pr_{s_0}^{\sigma, \pi}(\text{Reach}(\{s_1\})) \geq 1 - \varepsilon$; that is, $s_0 \in \text{Limit}_1^G(\text{Reach}(\{s_1\}))$.

We now argue that $s_0 \notin \text{Almost}_1^G(\text{Reach}(\{s_1\}))$. To see this, given a strategy σ for player 1, consider the following strategy π for player 2: for all $k \geq 0$, in round k , if player 1 chooses move a with probability 1, then player 2 chooses move c and ensures that s_1 is reached with probability 0; otherwise, in round k , if player 1 chooses move b with positive probability, then player 2 chooses move d , and the game reaches s_2 with positive probability.

Example 3 (*Büchi objectives*). Consider the concurrent game graph shown in Fig. 2(c). The transition function at s_0 is same as in Fig 2(b). The state s_2 is absorbing, and from state s_1 , the next state is always s_0 . The objective for player 1 is to visit s_1 infinitely often; that is, player 1 has the Büchi objective $\text{Büchi}(\{s_1\})$.

For any $\varepsilon > 0$, we construct a strategy σ_ε for player 1 as follows. Let $(\varepsilon_0, \varepsilon_1, \varepsilon, \dots)$ be an infinite sequence of reals $\varepsilon_i > 0$ such that $\prod_{i=0}^{\infty} (1 - \varepsilon_i) \geq 1 - \varepsilon$ (e.g., let $1 - \varepsilon_i = (1 - \varepsilon)^{\frac{1}{2^{i+1}}}$ for all $i \geq 0$). At the state s_0 , between the i -th and $(i + 1)$ -st visit to s_1 , fix an ε_i -optimal player-1 strategy to reach s_1 as described for Fig. 2(b), i.e., a strategy that ensures that s_1 is reached with probability $1 - \varepsilon_i$. The strategy ensures that against every player-2 strategy π , the state s_1 is visited infinitely often with probability $1 - \varepsilon$. Hence $s_0 \in \text{Limit}_1^G(\text{Büchi}(\{s_1\}))$. Note that the strategy σ_ε needs to count the number of visits to s_1 and therefore requires infinite memory. On the other hand, given any finite-memory strategy for player 1, there is a

strategy for player 2 that ensures with probability 1 that s_1 is visited only finitely often. It follows that in general ε -optimal strategies for concurrent games with Büchi objectives require infinite memory.

Characterization of values. Although infinite-memory strategies are required for almost-sure and limit-sure winning of concurrent games with parity objectives, there exist polynomial witnesses for such strategies [34]. This result was obtained by an analysis of certain μ -calculus formulas and established that the qualitative analysis of concurrent parity games can be achieved in $\text{NP} \cap \text{coNP}$. In contrast to turn-based games, in concurrent games with reachability objectives, values can be irrational even if all transition probabilities are rational [42]. The values of concurrent games with parity objectives can be characterized by formulas of a quantitative μ -calculus [42]. As a consequence, a 3EXPTIME algorithm was obtained for the quantitative analysis of concurrent parity games. This was later improved to PSPACE [16,13], and we follow that line of reasoning here. We present a reduction to obtain efficient witnesses for ε -optimal strategies in concurrent games from witnesses for limit-sure winning strategies in subgames [16]. A key concept in the reduction is the notion of so-called locally-optimal strategies.

Locally-optimal strategies. Consider a concurrent game graph G . A move selector ξ for player 1 at a state $s \in S$ is a distribution $\xi \in \text{Dist}(A)$ such that for all moves $a \in A$, if $\xi(a) > 0$, then $a \in \Gamma_1(s)$. Given a parity objective Φ for player 1, the player-1 move selector ξ at s is *locally optimal* if for all opponent moves $a_2 \in \Gamma_2(s)$, we have

$$\sum_{t \in S} \sum_{a_1 \in \Gamma_1(s)} \text{Val}_1^G(\Phi)(t) \cdot \delta(s, a_1, a_2)(t) \cdot \xi(a_1) \geq \text{Val}_1^G(\Phi)(s);$$

that is, for all opponent moves, the expected value of the game at the next state is at least the value of the game at the current state. We denote by \mathcal{E}_s the set of locally-optimal move selectors for player 1 at state s . A player-1 strategy $\sigma \in \Sigma$ is *locally optimal* if $\sigma(\tilde{s}' \cdot s) \in \mathcal{E}_s$ for all state sequences $\tilde{s}' \in S^*$ and states $s \in S$; that is, the strategy plays only locally-optimal move selectors.

From limit-sure winning to ε -optimal strategies. Let $G = (S, A, \Gamma_1, \Gamma_2, \delta)$ be a concurrent game graph and let p be a priority function for G . For a state $s \in S$, we write $\text{OptSupp}(s) = \{\text{Supp}(\xi) \mid \xi \in \mathcal{E}_s\}$ for the set of support sets of locally-optimal move selectors for player 1 at s . Consider the parity objective $\Phi = \text{Parity}(p)$ for player 1 and a value class $\text{VC}(\Phi, r)$, for some real $0 < r < 1$. We construct a new concurrent game graph $\tilde{G}_r = (\tilde{S}_r, \tilde{A}, \tilde{\Gamma}_1, \tilde{\Gamma}_2, \tilde{\delta})$ with a priority function \tilde{p} as follows:

(i) *State space.*

$$\tilde{S}_r = \{\tilde{s} \mid s \in \text{VC}(\Phi, r)\} \cup \{\langle s, \gamma \rangle \mid s \in \text{VC}(\Phi, r) \text{ and } \gamma \in \text{OptSupp}(s)\} \cup \{w_1, w_2\}.$$

(ii) *Priority function.*

- (a) $\tilde{p}(\tilde{s}) = p(s)$ for all $s \in \text{VC}(\Phi, r)$;
- (b) $\tilde{p}(\langle s, \gamma \rangle) = p(s)$ for all $s \in \text{VC}(\Phi, r)$ and $\gamma \in \text{OptSupp}(s)$;
- (c) $\tilde{p}(w_1) = 0$ and $\tilde{p}(w_2) = 1$.

(iii) *Move assignments.*

- (a) $\tilde{\Gamma}_1(\tilde{s}) = \text{OptSupp}(s)$ and $\tilde{\Gamma}_2(\tilde{s}) = \{\star\}$, where $\star \notin A$ is a new move; i.e., player 2 has no choice of moves.
- (b) $\tilde{\Gamma}_1(\langle s, \gamma \rangle) = \{\gamma\} \cup (\Gamma_1(s) \setminus \gamma)$ and $\tilde{\Gamma}_2(\langle s, \gamma \rangle) = \Gamma_2(s)$; i.e., for player 1, all moves in γ are collapsed into a single new move, and the moves not in γ are still available.

(iv) *Transition function.*

- (a) $\tilde{\delta}(\tilde{s}, \star)(\langle s, \gamma \rangle) = 1$; i.e., at state \tilde{s} , player 1 chooses an element of $\text{OptSupp}(s)$.
- (b) Transition function at state $\langle s, \gamma \rangle$;
 - (i) For all moves $a_2 \in \Gamma_2(s)$, if there exists a move $a_1 \in \gamma$ such that $\sum_{t \notin \text{VC}(\Phi, r)} \delta(s, a_1, a_2)(t) > 0$, then $\tilde{\delta}(\langle s, \gamma \rangle, \gamma, a_2)(w_1) = 1$; i.e., if player 1 chooses a move $a_1 \in \gamma$ and the original game on G proceeds with positive probability to a different value class, then the new game on \tilde{G}_r proceeds to w_1 . Note that since $a_1 \in \gamma$ and $\gamma \in \text{OptSupp}(s)$, if the game on G proceeds with positive probability to a different value class, then it proceeds with positive probability to a value class $\text{VC}(\Phi, r')$ with $r' > r$.
 - (ii) For all moves $a_2 \in \Gamma_2(s)$, if $\sum_{t \in \text{VC}(\Phi, r)} \delta(s, a_1, a_2)(t) = 1$ for all moves $a_1 \in \gamma$, then for each state $t \in \text{VC}(\Phi, r)$, let $\tilde{\delta}(\langle s, \gamma \rangle, \gamma, a_2)(\tilde{t}) = \sum_{a_1 \in \gamma} \xi(a_1) \cdot \delta(s, a_1, a_2)(t)$, where ξ is a locally-optimal move selector for player 1 at state s with $\text{Supp}(\xi) = \gamma$.
 - (iii) For all moves $a_1 \in (\Gamma_1(s) \setminus \gamma)$ and $a_2 \in \Gamma_2(s)$, let $\tilde{\delta}(\langle s, \gamma \rangle, a_1, a_2)(\tilde{t}) = \delta(s, a_1, a_2)(t)$ for each state $t \in \text{VC}(\Phi, r)$, and let $\tilde{\delta}(\langle s, \gamma \rangle, a_1, a_2)(w_2) = \sum_{t \notin \text{VC}(\Phi, r)} \delta(s, a_1, a_2)(t)$.
- (c) The states w_1 and w_2 are absorbing.

Observe that for the player-1 objective $\tilde{\Phi} = \text{Parity}(\tilde{p})$, the player-1 value at the state w_1 is 1, and the player-1 value at w_2 is 0.

Lemma 4. (See [16].) For all concurrent game graphs G , all parity objectives Φ , all reals $0 < r < 1$, and all states $s \in \text{VC}(\Phi, r)$, the state \tilde{s} is limit-sure winning for player 1 for the objective $\tilde{\Phi}$ in the game graph \tilde{G}_r .

Table 3

The strategy complexity of concurrent games with ω -regular objectives, where Σ^{PM} denotes the family of pure memoryless strategies, Σ^M denotes the family of randomized memoryless strategies, and Σ^{HI} denotes the family of randomized, history-dependent, infinite-memory strategies.

Objectives	Sure	Almost-sure	Limit-sure	ε -optimal
Reachability	Σ^{PM}	Σ^M	Σ^M	Σ^M
Safety	Σ^{PM}	Σ^{PM}	Σ^{PM}	Σ^M
Büchi	Σ^{PM}	Σ^M	Σ^{HI}	Σ^{HI}
coBüchi	Σ^{PM}	Σ^M	Σ^M	Σ^M
Parity	Σ^{PM}	Σ^{HI}	Σ^{HI}	Σ^{HI}
Rabin	Σ^{PM}	Σ^{HI}	Σ^{HI}	Σ^{HI}
Streett	Σ^{PF}	Σ^{HI}	Σ^{HI}	Σ^{HI}
Müller	Σ^{PF}	Σ^{HI}	Σ^{HI}	Σ^{HI}

Lemma 4 reduces the quantitative analysis of a concurrent game G with a parity objective to the qualitative analysis of subgames of the form \tilde{G}_r . Using Lemma 4, ε -optimal strategies on G can be obtained from limit-sure winning strategies on \tilde{G}_r and the approximation of locally-optimal strategies [16]. Limit-sure winning strategies for concurrent parity games can be found using the algorithm of [34], and the approximation of locally-optimal strategies can be defined by a formula in the alternation-free fragment of the theory of real-closed fields. This shows that the quantitative analysis of concurrent games with parity objectives can be performed in PSPACE.¹

8.1. Strategy complexity

Concurrent games with sure winning criteria coincide with 2-player (turn-based) games with sure winning criteria. Hence the results for the sure winning of concurrent games follow from the results for winning 2-player games. The most restrictive families of strategies that suffice for the almost-sure and limit-sure winning of concurrent games with parity objectives were characterized in [34]. Since concurrent games with Büchi objectives require randomized infinite-memory strategies for limit-sure winning (recall Example 3), it follows that concurrent games with parity, Rabin, and Streett objectives require infinite memory for limit-sure winning and for ε -optimality. The existence of memoryless optimal strategies for concurrent games with safety objectives and the existence of memoryless ε -optimal strategies, for all $\varepsilon > 0$, for concurrent games with reachability objectives, are classical [52]. The existence of memoryless ε -optimal strategies, for all $\varepsilon > 0$, for concurrent games with reachability objectives can be shown using an analysis of the limit behavior of discounted games with the aid of Puiseux series [52]; an elementary proof is available in [17]. The existence of memoryless ε -optimal strategies, for all $\varepsilon > 0$, for concurrent games with coBüchi objectives was established in [16] using Lemma 4 as a key observation. The results are summarized in Theorem 4.

Theorem 4. For all concurrent game graphs G , all Borel objectives Φ , and all states s , we have $\text{Val}_1^G(\Phi)(s) + \text{Val}_2^G(\Omega \setminus \Phi)(s) = 1$ [71]. The most restrictive families of strategies that suffice for sure winning, almost-sure winning, limit-sure winning, and ε -optimality on concurrent game graphs with respect to different classes of ω -regular objectives are presented in Table 3 [34,16].

8.2. Computational complexity

The complexity results for the sure winning of concurrent games follow from the corresponding results for 2-player games. Given a concurrent game graph of size n and a parity objective of e priorities, the almost-sure and limit-sure winning sets can be computed in time $O(n^{e+1})$, and the almost-sure and limit-sure winning properties of a state can be decided in $\text{NP} \cap \text{coNP}$ [34]. The quantitative analysis of concurrent games with reachability objectives can be performed in PSPACE [50]. From the results of [16] it follows that the quantitative analysis of concurrent games with parity objectives can also be accomplished in PSPACE (see [13] for details). A concurrent game with a Rabin or Streett objectives with d pairs can be transformed to a concurrent game with a parity objective with $O(d)$ priorities, where the size of the resulting game graph is exponentially larger than the size of the original game graph; the reduction uses an *index-appearance record* construction [86], which is an adaptation of the latest-appearance record construction of [53]. The transformation together with the qualitative analysis of concurrent parity games shows that the almost-sure and limit-sure winning sets of concurrent games with Rabin and Streett objectives can be computed in EXPTIME. Moreover, the transformation together with the quantitative analysis of concurrent parity games yields an EXPSPACE upper bound for the quantitative analysis of concurrent games with Rabin and Streett objectives. The upper bounds for concurrent games with Rabin and Streett objectives also hold for Müller objectives. The results are summarized in Theorem 5 and also presented in Table 4.

Theorem 5. Given a concurrent game graph G , an objective Φ , a rational number $r > 0$, and a state s , the following assertions hold:

¹ In [16] the complexity was wrongly claimed as $\text{NP} \cap \text{coNP}$; the details of the corrected result is available in [13].

Table 4The computational complexity of solving concurrent games with ω -regular objectives.

Objectives	Sure	Almost-sure	Limit-sure	Values
Reachability	PTIME	PTIME	PTIME	PSPACE
Safety	PTIME	PTIME	PTIME	PSPACE
Büchi	PTIME	PTIME	PTIME	PSPACE
coBüchi	PTIME	PTIME	PTIME	PSPACE
Parity	$NP \cap coNP$	$NP \cap coNP$	$NP \cap coNP$	PSPACE
Rabin	NP-complete	EXPTIME	EXPTIME	EXPSpace
Streett	coNP-complete	EXPTIME	EXPTIME	EXPSpace
Müller	PSPACE-complete	EXPTIME	EXPTIME	EXPSpace

- (i) (See [37,34].) whether $s \in \text{Sure}_1^G(\Phi)$, $s \in \text{Almost}_1^G(\Phi)$, or $s \in \text{Limit}_1^G(\Phi)$ can be decided in polynomial time if Φ is a reachability, safety, Büchi, or coBüchi objective;
- (ii) (See [34].) whether $s \in \text{Sure}_1^G(\Phi)$, $s \in \text{Almost}_1^G(\Phi)$, or $s \in \text{Limit}_1^G(\Phi)$ can be decided in $NP \cap coNP$ if Φ is a parity objective;
- (iii) whether $s \in \text{Sure}_1^G(\Phi)$ is NP-complete if Φ is a Rabin objective, coNP-complete if Φ is a Streett objective, and PSPACE-complete if Φ is a Müller objective;
- (iv) whether $s \in \text{Almost}_1^G(\Phi)$ or $s \in \text{Limit}_1^G(\Phi)$ can be decided in EXPTIME if Φ is a Rabin, Streett, or Müller objective;
- (v) (See [50,16,13].) whether $\text{Val}_1^G(\Phi)(s) \geq r$ can be decided in PSPACE if Φ is a reachability, safety, Büchi, coBüchi, or parity objective;
- (vi) whether $\text{Val}_1^G(\Phi)(s) \geq r$ can be decided in EXPSpace if Φ is a Rabin, Streett, or Müller objective.

8.3. Algorithms and open problems

Given a concurrent game graph of size n and a parity objective Φ with e priorities, the winning sets $\text{Sure}_1^G(\Phi)$, $\text{Almost}_1^G(\Phi)$, and $\text{Limit}_1^G(\Phi)$ can be computed in time $O(n^{e+1})$ [34]. This result was obtained by defining the three winning sets in the μ -calculus. The evaluation of the μ -calculus formulas by iterative fixpoint approximation give algorithms for the qualitative analysis of concurrent games with parity objectives [34]. Similarly, the value function of a concurrent game with a parity objective can be defined using a quantitative μ -calculus [42], but iterative fixpoint iteration may not terminate. From the fixpoint characterization of value functions, a 3EXPTIME algorithm for the quantitative analysis of concurrent parity games can be obtained by a reduction to the theory of the real-closed fields (a 2EXPTIME decision procedure is applied to an exponential-size formula with addition and multiplication over the reals) [42].

The most interesting open problems for concurrent games are the following.

- (i) The best known lower bounds for computing the almost-sure and limit-sure winning sets for concurrent games with Rabin and Streett objectives are NP-hard and coNP-hard, respectively (this follows from the hardness of the corresponding 2-player games). The best known upper bounds are EXPTIME. It is open if the problems are NP-complete and coNP-complete, respectively.
- (ii) The best known lower bounds for computing the values of concurrent games with Rabin and Streett objectives are again NP-hard and coNP-hard, respectively. The best known upper bounds are EXPSpace. No PSPACE or EXPTIME algorithms are known for these problems.

9. Nonzero-sum games

9.1. Nash equilibria in turn-based games

A key technique to prove the existence of Nash equilibria in n -player turn-based probabilistic games is a general construction from repeated games based on so-called “threat” strategies. The basic idea is that each player plays an optimal strategy in the zero-sum game against all other players. Any deviation of a player i from this strategy is punished indefinitely by the other players, who will switch to optimal strategies in the zero-sum game against player i (see, e.g., [76,89]). The following lemma shows that such threat strategies can be effectively applied against pure strategies.

Lemma 5. (See [27].) For all $\varepsilon \geq 0$, if the family Σ^P of pure player-1 strategies suffices for ε -optimality on all $2\frac{1}{2}$ -player game graphs with respect to a class \mathcal{O} of player-1 objectives that is closed under complementation, then ε -Nash equilibria exist in all n -player turn-based probabilistic games where each player has an objective from \mathcal{O} .

Proof. We prove the first part; the proof of the second part is similar. Let the objective of player i be Φ_i , for $i \in \{1, \dots, n\}$. Consider the n zero-sum games played between player i and the team $\{1, 2, \dots, n\} \setminus \{i\}$ of players, with the objective Φ_i for player i and the objective $\Omega \setminus \Phi_i$ for the opposing team of players. By assumption, there is a pure ε -optimal strategy π_i^i for player i in the game with the objective Φ_i , and a pure ε -optimal strategy π_j^j for each player $j \neq i$ in the game with the objective $\Omega \setminus \Phi_i$. Now consider the following strategy τ^i for each player $i \in \{1, \dots, n\}$. Player i plays according to the

Table 5

Nash equilibria in n -player turn-based probabilistic games, where “NE” denotes the existence of Nash equilibria, “ ε -NE” denotes the existence of ε -Nash equilibria for all $\varepsilon > 0$, Σ^P denotes the family of pure strategies, and Σ^{PF} denotes the family of pure finite-memory strategies.

Game graph	Objectives	Existence	Strategies
Turn-based deterministic n -player	Borel	NE	Σ^P
Turn-based probabilistic n -player	ω -regular	NE	Σ^{PF}
Turn-based probabilistic n -player	Borel	ε -NE	Σ^P

Table 6

Nash equilibria in n -player concurrent games, where “NE” denotes the existence of Nash equilibria and “ ε -NE” denotes the existence of ε -Nash equilibria for all $\varepsilon > 0$.

Game graph	Objective	Existence
Concurrent 2-player	ω -regular	ε -NE
Concurrent 2-player	Borel	??
Concurrent n -player	safety	NE
Concurrent n -player	reachability	ε -NE
Concurrent n -player	ω -regular	??
Concurrent n -player	Borel	??

strategy π_i^i as long as all other players $j \neq i$ play according to π_j^j , and player i switches to π_j^i as soon as some player j deviates from π_j^j . Since the strategies are pure, any deviation is immediately noted. The strategies τ^i , for $i = 1, \dots, n$, form an ε -Nash equilibrium. \square

Lemma 5 and the existence of pure optimal strategies for 2-player games with Borel objectives and $2\frac{1}{2}$ -player games with ω -regular objectives (Theorem 2) proves the existence of Nash equilibria in n -player turn-based deterministic games with Borel objectives and in n -player turn-based probabilistic games with ω -regular objectives. The results are summarized in Theorem 6 and also shown in Table 5.

Theorem 6. (See [27].) *The following assertions hold.*

- (i) *The family of pure strategies suffices for the existence of Nash equilibria on n -player turn-based deterministic game graphs with respect to Borel objectives.*
- (ii) *The family of pure finite-memory strategies suffices for the existence of Nash equilibria on n -player turn-based probabilistic game graphs with respect to ω -regular objectives.*
- (iii) *The family of pure strategies suffices for the existence of ε -Nash equilibria, for all $\varepsilon > 0$, on n -player turn-based probabilistic game graphs with respect to Borel objectives. There are 2-player turn-based probabilistic games with objectives on the third level of the Borel hierarchy (i.e., in Σ_3) for both players for which no Nash equilibria exist.*

In [22] a refined notion of Nash equilibrium is presented for the special case of 2-player (turn-based deterministic) nonzero-sum games. The proof techniques of [22] also use the notion of threat strategies.

9.2. Nash equilibria in concurrent games

In the case of concurrent games, results about the existence of Nash equilibria are known only for low levels of the Borel hierarchy. Seechi and Sudderth [84] established the existence of Nash equilibria in n -player concurrent games where each player has a safety objective. The existence of ε -Nash equilibria, for all $\varepsilon > 0$, in n -player concurrent games with reachability objectives for all players was shown in [27]. In the special case of 2-player concurrent games, the existence of ε -Nash equilibria, for all $\varepsilon > 0$, was proved for all ω -regular objectives [10]. The latter result uses threat strategies and the reduction principle. First, [10] identifies sufficient conditions that guarantee the existence of ε -Nash equilibria, and shows that if the conditions are not satisfied, then the a nonzero-sum game with ω -regular objectives can be reduced to a nonzero-sum game with reachability objectives. Then the existence of ε -Nash equilibria is established using threat strategies; however, the construction of threat strategies is more involved as the strategies can be randomized. The results are summarized in Theorem 7. The known results and open problems are listed in Table 6.

Theorem 7. *The following assertions hold.*

- (i) (See [84].) *Nash equilibria exist for n -player concurrent games with safety objectives for all players.*
- (ii) (See [27].) *ε -Nash equilibria, for all $\varepsilon > 0$, exist for n -player concurrent games with reachability objectives for all players.*
- (iii) (See [10].) *ε -Nash equilibria, for all $\varepsilon > 0$, exist for 2-player concurrent games with ω -regular objectives.*

10. Related topics

In this survey we focused on two-player games played on graphs with finite state spaces, where each player has perfect information about the state of the game, and the objectives of the players are qualitative (i.e., for each player, every path is either winning or losing). We briefly discuss several extensions of such games which have been studied in the literature, and give a few relevant references (there is no attempt at being exhaustive).

Partial-information games. In *partial-information* games, the players choose their moves based on incomplete information about the state of the game. Such games are harder to solve than the corresponding perfect-information games. For example, turn-based deterministic (2-player) games with partial information and zero-sum reachability/safety objectives are 2EXPTIME-complete [83]. In the presence of more than two players, turn-based deterministic games with partial information and reachability objectives (for one of the players) are even undecidable [83]. A key technique to solve partial-information games (when possible) is by reduction to perfect-information games, using a subset construction on the state space similar to the determinization of finite automata. The results in [19] present a close connection between a subclass of partial-information turn-based games and perfect-information concurrent games. The algorithmic analysis of partial-information turn-based games with ω -regular objectives is studied in [18]; the complexity of partial-information MDPs, in [77]. Another interesting variety of partial-information games is the class of games where the starting state is unknown [55].

Infinite-state games. There are several extensions of games played on finite state spaces to games played on infinite state spaces. Notable examples are *pushdown* games and *timed* games. In the case of pushdown games, the state of a game encodes an unbounded amount of information in the form of the contents of a stack. Deterministic pushdown games are solved in [92] (see [93] for a survey); probabilistic pushdown games, in [49,50]. In the case of timed games, the state of a game encodes an unbounded amount of information in the form of real-numbered values for finitely many clocks. Timed games are studied in [68,33].

Quantitative objectives. Quantitative objectives, where each player tries to maximize a numerical payoff, are standard in game theory and economics. In the case of graph games, the states or edges of the game graph are labeled with numbers that represent rewards, and the rewards that occur along a path determine each player's payoff. Notable examples of quantitative objectives on such labeled game graphs are *discounted-reward* objectives and *limit-average* (or *mean-payoff*) objectives. Games with discounted rewards were introduced by Lloyd Shapley [85] and have been extensively studied in economics, and recently also in systems theory [38]. They have several pleasant mathematical properties, such as robustness with respect to slight perturbations in the numerical labels of a game graph. For turn-based graph games with limit-average objectives, the existence of pure memoryless optimal strategies was shown in [46]. The determinacy of concurrent games with limit-average objectives was proved in [74]; see also [52] for a detailed analysis of these games. The results of [89,90] prove the existence of ε -Nash equilibria in two-player nonzero-sum concurrent games with limit-average objectives, for all $\varepsilon > 0$. The complexity of turn-based deterministic graph games with limit-average objectives is studied in [96]; the complexity of concurrent graph games with limit-average objectives, in [26].

Logic and games. The connection between logical quantifiers and games is deep and well-established. Game theory also provides a useful framework for studying properties of sets. The results of Martin [70,71] establishing Borel determinacy for turn-based deterministic and concurrent games illuminate several key properties of sets. The close connection between logics on trees and turn-based deterministic graph games is well-exposed in [87]. The μ -calculus is a logic of fixed points which is expressive enough to capture all ω -regular objectives [65]. Allen Emerson and Charanjit Jutla [48] established the equivalence of μ -calculus model checking and solving turn-based deterministic graph games with parity objectives. A quantitative μ -calculus is proposed in [42] to solve concurrent graph games with parity objectives, and in [72] to solve turn-based probabilistic graph games with parity objectives. The alternating-time temporal logic ATL requires game solving to solve the model-checking problem [2].

Relationships between games. Establishing qualitative and quantitative relationships between games is an intriguing area of research. The notions of abstractions for game graphs [57,54], refinement relations between game graphs [3], and distances between game graphs [38,43] are explored in the literature.

References

- [1] M. Abadi, L. Lamport, P. Wolper, Realizable and unrealizable specifications of reactive systems, in: ICALP'89, in: Lecture Notes in Comput. Sci., vol. 372, Springer, 1989, pp. 1–17.
- [2] R. Alur, T.A. Henzinger, O. Kupferman, Alternating-time temporal logic, J. ACM 49 (2002) 672–713.
- [3] R. Alur, T.A. Henzinger, O. Kupferman, M.Y. Vardi, Alternating refinement relations, in: CONCUR'98, in: Lecture Notes in Comput. Sci., vol. 1466, Springer, 1998, pp. 163–178.
- [4] C. Beeri, On the membership problem for functional and multivalued dependencies in relational databases, ACM Trans. Database Syst. 5 (1980) 241–259.

- [5] H. Björklund, S. Sandberg, S. Vorobyov, A discrete subexponential algorithms for parity games, in: STACS'03, in: Lecture Notes in Comput. Sci., vol. 2607, Springer, 2003, pp. 663–674.
- [6] H. Björklund, S. Sandberg, S. Vorobyov, Randomized subexponential algorithms for infinite games, Technical report: DIMACS 2004–09, 2004.
- [7] A. Blass, Y. Gurevich, L. Nachmanson, M. Veanes, Play to test, in: FATES'05, 2005.
- [8] J.R. Büchi, On a decision method in restricted second-order arithmetic, in: E. Nagel, P. Suppes, A. Tarski (Eds.), Proceedings of the First International Congress on Logic, Methodology, and Philosophy of Science 1960, Stanford University Press, 1962, pp. 1–11.
- [9] J.R. Büchi, L.H. Landweber, Solving sequential conditions by finite-state strategies, Trans. Amer. Math. Soc. 138 (1969) 295–311.
- [10] K. Chatterjee, Two-player nonzero-sum ω -regular games, in: CONCUR'05, in: Lecture Notes in Comput. Sci., vol. 3653, Springer, 2005, pp. 413–427.
- [11] K. Chatterjee, Optimal strategy synthesis for stochastic Müller games, in: FoSSaCS'07, in: Lecture Notes in Comput. Sci., vol. 4423, Springer, 2007, pp. 138–152.
- [12] K. Chatterjee, Stochastic Müller games are PSPACE-complete, in: FSTTCS 2007: Foundations of Software Technology and Theoretical Computer Science, in: Lecture Notes in Comput. Sci., vol. 4855/2007, 2007, pp. 436–448.
- [13] K. Chatterjee, Stochastic ω -regular games, PhD thesis, UC Berkeley, 2007.
- [14] K. Chatterjee, L. de Alfaro, T.A. Henzinger, Trading memory for randomness, in: QEST'04, IEEE, 2004, pp. 206–217.
- [15] K. Chatterjee, L. de Alfaro, T.A. Henzinger, The complexity of stochastic Rabin and Streett games, in: ICALP'05, in: Lecture Notes in Comput. Sci., vol. 3580, Springer, 2005, pp. 878–890.
- [16] K. Chatterjee, L. de Alfaro, T.A. Henzinger, The complexity of quantitative concurrent parity games, in: SODA'06, ACM-SIAM, 2006, pp. 678–687.
- [17] K. Chatterjee, L. de Alfaro, T.A. Henzinger, Strategy improvement in concurrent reachability games, in: QEST'06, IEEE, 2006, pp. 291–300.
- [18] K. Chatterjee, L. Doyen, T.A. Henzinger, J.F. Raskin, Algorithms for omega-regular games with imperfect information, in: CSL'06, in: Lecture Notes in Comput. Sci., vol. 4207, Springer, 2006, pp. 287–302.
- [19] K. Chatterjee, T.A. Henzinger, Semiperfect-information games, in: FSTTCS'05, in: Lecture Notes in Comput. Sci., vol. 3821, Springer, 2005, pp. 1–18.
- [20] K. Chatterjee, T.A. Henzinger, Strategy improvement and randomized subexponential algorithms for stochastic parity games, in: STACS'06, in: Lecture Notes in Comput. Sci., vol. 3884, Springer, 2006, pp. 512–523.
- [21] K. Chatterjee, T.A. Henzinger, Strategy improvement for stochastic Rabin and Streett games, in: CONCUR'06, in: Lecture Notes in Comput. Sci., vol. 4137, Springer, 2006, pp. 375–389.
- [22] K. Chatterjee, T.A. Henzinger, M. Jurdziński, Games with secure equilibria, in: LICS'04, IEEE, 2004, pp. 160–169.
- [23] K. Chatterjee, T.A. Henzinger, N. Piterman, Generalized parity games, in: FoSSaCS'07, in: Lecture Notes in Comput. Sci., vol. 4423, Springer, 2007, pp. 153–167.
- [24] K. Chatterjee, M. Jurdziński, T.A. Henzinger, Simple stochastic parity games, in: CSL'03, in: Lecture Notes in Comput. Sci., vol. 2803, Springer, 2003, pp. 100–113.
- [25] K. Chatterjee, M. Jurdziński, T.A. Henzinger, Quantitative stochastic parity games, in: SODA'04, SIAM, 2004, pp. 121–130.
- [26] K. Chatterjee, R. Majumdar, T.A. Henzinger, Stochastic limit-average games are in EXPTIME, Internat. J. Game Theory 37 (2008) 219–234.
- [27] K. Chatterjee, R. Majumdar, M. Jurdziński, On Nash equilibria in stochastic games, in: CSL'04, in: Lecture Notes in Comput. Sci., vol. 3210, Springer, 2004, pp. 26–40.
- [28] A. Church, Logic, arithmetic, and automata, in: Proceedings of the International Congress of Mathematicians, Institut Mittag-Leffler, 1962, pp. 23–35.
- [29] A. Condon, The complexity of stochastic games, Inform. and Comput. 96 (2) (1992) 203–224.
- [30] A. Condon, On algorithms for simple stochastic games, in: Advances in Computational Complexity Theory, in: DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 13, Amer. Math. Soc., 1993, pp. 51–73.
- [31] C. Courcoubetis, M. Yannakakis, The complexity of probabilistic verification, J. ACM 42 (4) (1995) 857–907.
- [32] L. de Alfaro, Formal verification of probabilistic systems, PhD thesis, Stanford University, 1997.
- [33] L. de Alfaro, M. Faella, T.A. Henzinger, R. Majumdar, M. Stoelinga, The element of surprise in timed games, in: CONCUR'03, in: Lecture Notes in Comput. Sci., vol. 2761, Springer-Verlag, 2003, pp. 144–158.
- [34] L. de Alfaro, T.A. Henzinger, Concurrent omega-regular games, in: LICS'00, IEEE, 2000, pp. 141–154.
- [35] L. de Alfaro, T.A. Henzinger, Interface automata, in: FSE'01, ACM Press, 2001, pp. 109–120.
- [36] L. de Alfaro, T.A. Henzinger, Interface theories for component-based design, in: EMSOFT'01, in: Lecture Notes in Comput. Sci., vol. 2211, Springer, 2001, pp. 148–165.
- [37] L. de Alfaro, T.A. Henzinger, O. Kupferman, Concurrent reachability games, in: FOCS'98, IEEE, 1998, pp. 564–575.
- [38] L. de Alfaro, T.A. Henzinger, R. Majumdar, Discounting the future in systems theory, in: ICALP'03, in: Lecture Notes in Comput. Sci., vol. 2719, Springer, 2003, pp. 1022–1037.
- [39] L. de Alfaro, T.A. Henzinger, F.Y.C. Mang, The control of synchronous systems, in: CONCUR'00, in: Lecture Notes in Comput. Sci., vol. 1877, Springer, 2000, pp. 458–473.
- [40] L. de Alfaro, T.A. Henzinger, F.Y.C. Mang, Detecting errors before reaching them, in: CAV'00, 2000, pp. 186–201.
- [41] L. de Alfaro, T.A. Henzinger, F.Y.C. Mang, The control of synchronous systems, part ii, in: CONCUR'01, in: Lecture Notes in Comput. Sci., vol. 2154, Springer, 2001, pp. 566–580.
- [42] L. de Alfaro, R. Majumdar, Quantitative solution of omega-regular games, in: STOC'01, ACM Press, 2001, pp. 675–683.
- [43] L. de Alfaro, R. Majumdar, M. Stoelinga, V. Raman, Game relations and metrics, in: LICS'07, IEEE, 2007, pp. 99–108.
- [44] D.L. Dill, Trace Theory for Automatic Hierarchical Verification of Speed-Independent Circuits, The MIT Press, 1989.
- [45] S. Dziembowski, M. Jurdziński, I. Walukiewicz, How much memory is needed to win infinite games? in: LICS'97, IEEE, 1997, pp. 99–110.
- [46] A. Ehrenfeucht, J. Mycielski, Positional strategies for mean payoff games, Internat. J. Game Theory 8 (2) (1979) 109–113.
- [47] E.A. Emerson, C. Jutla, The complexity of tree automata and logics of programs, in: FOCS'88, IEEE, 1988, pp. 328–337.
- [48] E.A. Emerson, C. Jutla, Tree automata, mu-calculus and determinacy, in: FOCS'91, IEEE, 1991, pp. 368–377.
- [49] K. Etessami, M. Yannakakis, Recursive Markov decision processes and recursive stochastic games, in: ICALP'05, in: Lecture Notes in Comput. Sci., vol. 3580, Springer, 2005, pp. 891–903.
- [50] K. Etessami, M. Yannakakis, Recursive concurrent stochastic games, in: ICALP'06 (2), in: Lecture Notes in Comput. Sci., vol. 4052, Springer, 2006, pp. 324–335.
- [51] H. Everett, Recursive games, in: Contributions to the Theory of Games III, in: Ann. of Math. Stud., vol. 39, 1957, pp. 47–78.
- [52] J. Filar, K. Vrieze, Competitive Markov Decision Processes, Springer-Verlag, 1997.
- [53] Y. Gurevich, L. Harrington, Trees, automata, and games, in: STOC'82, ACM Press, 1982, pp. 60–65.
- [54] T.A. Henzinger, R. Jhala, R. Majumdar, Counterexample-guided control, in: ICALP'03, in: Lecture Notes in Comput. Sci., vol. 2719, Springer, 2003, pp. 886–902.
- [55] T.A. Henzinger, S.C. Krishnan, O. Kupferman, F.Y.C. Mang, Synthesis of uninitialized systems, in: ICALP'02, in: Lecture Notes in Comput. Sci., vol. 2380, Springer, 2002, pp. 644–656.
- [56] T.A. Henzinger, O. Kupferman, S. Rajamani, Fair simulation, Inform. and Comput. 173 (2002) 64–81.
- [57] T.A. Henzinger, R. Majumdar, F.Y.C. Mang, J.-F. Raskin, Abstract interpretation of game properties, in: SAS'00, in: Lecture Notes in Comput. Sci., vol. 1824, Springer, 2000, pp. 220–239.

- [58] F. Horn, Streett games on finite graphs, in: GDV'05, 2005.
- [59] P. Hunter, A. Dawar, Complexity bounds for regular games, in: MFCS'05, 2005, pp. 495–506.
- [60] N. Immerman, Number of quantifiers is better than number of tape cells, *J. Comput. System Sci.* 22 (1981) 384–406.
- [61] J.F. Nash Jr., Equilibrium points in n -person games, *Proc. Natl. Acad. Sci. USA* 36 (1950) 48–49.
- [62] M. Jurdziński, Deciding the winner in parity games is in $UP \cap co-UP$, *Inform. Process. Lett.* 68 (3) (1998) 119–124.
- [63] M. Jurdziński, Small progress measures for solving parity games, in: STACS'00, in: *Lecture Notes in Comput. Sci.*, vol. 1770, Springer, 2000, pp. 290–301.
- [64] M. Jurdziński, M. Paterson, U. Zwick, A deterministic subexponential algorithm for solving parity games, in: SODA'06, ACM–SIAM, 2006, pp. 117–123.
- [65] D. Kozen, Results on the propositional μ -calculus, *Theoret. Comput. Sci.* 27 (3) (1983) 333–354.
- [66] O. Kupferman, M.Y. Vardi, Weak alternating automata and tree automata emptiness, in: STOC'98, ACM Press, 1998, pp. 224–233.
- [67] W. Ludwig, A subexponential randomized algorithm for the simple stochastic game problem, *Inform. and Comput.* 117 (1995) 151–155.
- [68] O. Maler, A. Pnueli, J. Sifakis, On the synthesis of discrete controllers for timed systems, in: STACS'95, in: *Lecture Notes in Comput. Sci.*, vol. 900, Springer, 1995, pp. 229–242.
- [69] Z. Manna, A. Pnueli, *The Temporal Logic of Reactive and Concurrent Systems: Specification*, Springer-Verlag, 1992.
- [70] D.A. Martin, Borel determinacy, *Ann. of Math.* 102 (2) (1975) 363–371.
- [71] D.A. Martin, The determinacy of Blackwell games, *J. Symbolic Logic* 63 (4) (1998) 1565–1581.
- [72] A. McIver, C. Morgan, Games, probability, and the quantitative μ -calculus $q\mu$, in: LPAR'02, in: *Lecture Notes in Comput. Sci.*, vol. 2514, Springer, 2002, pp. 292–310.
- [73] R. McNaughton, Infinite games played on finite graphs, *Ann. Pure Appl. Logic* 65 (1993) 149–184.
- [74] J.F. Mertens, A. Neyman, Stochastic games, *Internat. J. Game Theory* 10 (1981) 53–66.
- [75] A.W. Mostowski, Regular expressions for infinite trees and a standard form of automata, in: 5th Symposium on Computation Theory, in: *Lecture Notes in Comput. Sci.*, vol. 208, Springer, 1984, pp. 157–168.
- [76] G. Owen, *Game Theory*, Academic Press, 1995.
- [77] C.H. Papadimitriou, J.N. Tsitsiklis, The complexity of Markov decision processes, *Math. Oper. Res.* 12 (1987) 441–450.
- [78] N. Piterman, A. Pnueli, Faster solution of Rabin and Streett games, in: LICS'06, IEEE, 2006, pp. 275–284.
- [79] A. Pnueli, R. Rosner, On the synthesis of a reactive module, in: POPL'89, ACM Press, 1989, pp. 179–190.
- [80] A. Puri, Theory of hybrid systems and discrete event systems, PhD thesis, University of California, Berkeley, 1995.
- [81] M.O. Rabin, Automata on Infinite Objects and Church's Problem, *Conference Ser. Math.*, vol. 13, Amer. Math. Soc., 1969.
- [82] P.J. Ramadge, W.M. Wonham, Supervisory control of a class of discrete-event processes, *SIAM J. Control Optim.* 25 (1) (1987) 206–230.
- [83] J.H. Reif, Universal games of incomplete information, in: STOC'79, ACM Press, 1979, pp. 288–308.
- [84] P. Secchi, W.D. Sudderth, Stay-in-a-set games, *Internat. J. Game Theory* 30 (2001) 479–490.
- [85] L.S. Shapley, Stochastic games, *Proc. Natl. Acad. Sci. USA* 39 (1953) 1095–1100.
- [86] W. Thomas, On the synthesis of strategies in infinite games, in: STACS'95, in: *Lecture Notes in Comput. Sci.*, vol. 900, Springer, 1995, pp. 1–13.
- [87] W. Thomas, Languages, automata, and logic, in: G. Rozenberg, A. Salomaa (Eds.), *Handbook of Formal Languages*, vol. 3, Beyond Words, Springer, 1997, pp. 389–455 (Chapter 7).
- [88] M.Y. Vardi, Automatic verification of probabilistic concurrent finite-state systems, in: FOCS'85, IEEE Computer Society Press, 1985, pp. 327–338.
- [89] N. Vieille, Two player stochastic games I: A reduction, *Israel J. Math.* 119 (2000) 55–91.
- [90] N. Vieille, Two player stochastic games II: The case of recursive games, *Israel J. Math.* 119 (2000) 93–126.
- [91] J. Vöge, M. Jurdziński, A discrete strategy improvement algorithm for solving parity games, in: CAV'00, in: *Lecture Notes in Comput. Sci.*, vol. 1855, Springer, 2000, pp. 202–215.
- [92] I. Walukiewicz, Pushdown processes: Games and model checking, in: CAV'96, in: *Lecture Notes in Comput. Sci.*, vol. 1102, Springer, 1996, pp. 62–74.
- [93] I. Walukiewicz, A landscape with games in the background, in: LICS'04, IEEE, 2004, pp. 356–366.
- [94] W. Zielonka, Perfect-information stochastic parity games, in: FoSSaCS'04, in: *Lecture Notes in Comput. Sci.*, vol. 2987, Springer, 2004, pp. 499–513.
- [95] W. Zielonka, Infinite games on finitely coloured graphs with applications to automata on infinite trees, *Theoret. Comput. Sci.* 200 (1–2) (1998) 135–183.
- [96] U. Zwick, M. Paterson, The complexity of mean payoff games on graphs, *Theoret. Comput. Sci.* 158 (1996) 343–359.