Finitary Winning in ω-Regular Games*

Krishnendu Chatterjee[1] and Thomas A. Henzinger[1,2]

[1] University of California, Berkeley, USA
[2] EPFL, Switzerland
{c_krish, tah}@eecs.berkeley.edu

Abstract. Games on graphs with ω-regular objectives provide a model for the control and synthesis of reactive systems. Every ω-regular objective can be decomposed into a safety part and a liveness part. The liveness part ensures that something good happens "eventually." Two main strengths of the classical, infinite-limit formulation of liveness are robustness (independence from the granularity of transitions) and simplicity (abstraction of complicated time bounds). However, the classical liveness formulation suffers from the drawback that the time until something good happens may be unbounded. A stronger formulation of liveness, so-called *finitary* liveness, overcomes this drawback, while still retaining robustness and simplicity. Finitary liveness requires that there exists an unknown, fixed bound b such that something good happens within b transitions. While for one-shot liveness (reachability) objectives, classical and finitary liveness coincide, for repeated liveness (Büchi) objectives, the finitary formulation is strictly stronger. In this work we study games with finitary parity and Streett (fairness) objectives. We prove the determinacy of these games, present algorithms for solving these games, and characterize the memory requirements of winning strategies. Our algorithms can be used, for example, for synthesizing controllers that do not let the response time of a system increase without bound.

1 Introduction

Games played on graphs are suitable models for multi-component systems: vertices represent states; edges represent transitions; players represent components; and objectives represent specifications. The specification of a component is typically given as an ω-regular condition [9], and the resulting ω-regular games have been used for solving control and verification problems (see, e.g., [3, 11, 12]).

Every ω-regular specification (indeed, every specification) can be decomposed into a safety part and a liveness part [1]. The safety part ensures that the component will not do anything "bad" (such as violate an invariant) within any finite number of transitions. The liveness part ensures that the component will do something "good" (such as proceed, or respond, or terminate) within some finite number of transitions. Liveness can be violated only in the limit, by infinite sequences of transitions, as no bound is stipulated on when the "good" thing

* This research was supported in part by the AFOSR MURI grant F49620-00-1-0327 and the NSF ITR grant CCR-0225610.

H. Hermanns and J. Palsberg (Eds.): TACAS 2006, LNCS 3920, pp. 257–271, 2006.

must happen. This infinitary, classical formulation of liveness has both strengths and weaknesses. A main strength is robustness, in particular, independence from the chosen granularity of transitions. Another main strength is simplicity, allowing liveness to serve as an abstraction for complicated safety conditions. For example, a component may always respond in a number of transitions that depends, in some complicated manner, on the exact size of the stimulus. Yet for correctness, we may be interested only that the component will respond "eventually." However, these strengths also point to a weakness of the classical definition of liveness: it can be satisfied by components that in practice are quite unsatisfactory because no bound can be put on their response time. It is for this reason that alternative, stronger formulations of liveness have been proposed. One of these is *finitary* liveness [2, 4]: finitary liveness does not insist on response within a known bound b (i.e., every stimulus is followed by a response within b transitions), but on response within some unknown bound (i.e., there exists b such that every stimulus is followed by a response within b transitions). Note that in the finitary case, the bound b may be arbitrarily large, but the response time must not grow forever from one stimulus to the next. In this way, finitary liveness still maintains the robustness (independence of step granularity) and simplicity (abstraction of complicated safety) of traditional liveness, while removing unsatisfactory implementations.

In this paper, we study graph games with finitary winning conditions. The motivation is the same as for finitary liveness. Consider, for example, the synthesis of an elevator controller as a strategy in a game where one player represents the environment (i.e., the pushing of call buttons on various floors, and the pushing of target buttons inside the elevators), and the other player represents the elevator control (i.e., the commands to move an elevator up or down, and the opening and closing of elevator doors). Clearly, one objective of the controller is that whenever a call button is pushed on a floor, then an elevator will eventually arrive, and whenever a target button is pushed inside an elevator, then the elevator will eventually get to the corresponding floor. Note that this objective is formulated in an infinitary way (the key term is "eventually"). This is because, for robustness and simplicity, we do not wish to specify for each state the exact number of transitions until the objective must be met. However, a truly unbounded implementation of elevator control (where the response time grows from request to request, without bound) would be utterly unsatisfactory. A finitary interpretation of the objective prohibits such undesirable control strategies: there must exist a bound b such that the controller meets every call request, and every target request, within b transitions.

We formalize finitary winning for the normal form of ω-regular objectives called *parity* conditions [13]. A parity objective assigns a non-negative integer priority to every vertex, and the objective of player 1 is to make sure that the lowest priority that repeats infinitely often is even. This is an infinitary objective, as player 1 can win by ensuring that every odd priority that repeats infinitely often is followed by a smaller even priority "eventually" (arbitrarily many transitions later). The *finitary* parity objective, by contrast, insists that

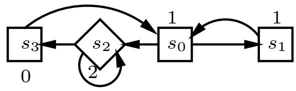

Fig. 1. A simple game graph

player 1 ensures that there exists a bound b such that every odd priority that repeats infinitely often is followed by a smaller even priority within b transitions. The finitary parity objective is stronger than the classical parity objective, as is illustrated by the following example.

Example 1. Consider the game shown in Figure 1. The square-shaped states are player 1 states, where player 1 chooses the successor state, and the diamond-shaped states are player 2 states (we will follow this convention throughout this paper). The priorities of states are shown next to each state in the figure. If player 1 follows a memoryless strategy σ that chooses the successor s_2 at state s_0, this ensures that against all strategies π for player 2, the minimum priority of the states that are visited infinitely often is even (either state s_3 is visited infinitely often, or both states s_0 and s_1 are visited finitely often). However, consider the strategy π_w for player 2: the strategy π_w is played in rounds, and in round $k \geq 0$, whenever player 1 chooses the successor s_2 at state s_0, player 2 stays in state s_2 for k transitions, and then goes to state s_3 and proceeds to round $k+1$. The strategy π_w ensures that for all strategies σ for player 1, either the minimum priority visited infinitely often is 1 (i.e., both states s_0 and s_1 are visited infinitely often and state s_3 is visited finitely often); or states of priority 1 are visited infinitely often, and the distances between visits to states of priority 1 and subsequent visits to states of priority 0 increase without bound (i.e., the limit of the distances is ∞). Hence it follows that in this game, although player 1 can win for the parity objective, she cannot win for the finitary parity objective. ∎

We prove that games with finitary parity objectives are determined: for every state either there is a player 1 strategy (a winning strategy for player 1) that ensures that the finitary parity objective is satisfied against all player 2 strategies, or there is a player 2 strategy (a winning strategy for player 2) that ensures that the finitary parity objective is violated against all player 1 strategies. Similar to games with infinitary parity objectives, we establish the existence of winning strategies that are *memoryless* (independent of the history of the play) for player 1. However, winning strategies for player 2 in general require infinite memory; this is in contrast to infinitary parity objectives, where memoryless winning strategies exist also for player 2 [5]. We present an algorithm to compute the winning sets in time $O(n^{2d-3} \cdot d \cdot m)$ for game graphs with n states and m edges, and for finitary parity objectives with d priorities. Games with infinitary parity objectives can be solved in time $O(n^{\lfloor \frac{d}{2} \rfloor} \cdot m)$ [8]. Since in the case of finitary parity objectives, winning strategies for player 2 require infinite memory in general, the analysis and the algorithm for games with finitary parity objectives is more involved. We also show that polynomial-size witnesses exist for the winning strategies of both players; in particular, even though the winning strategies for player 2 may require infinite memory, there exist polynomial witnesses for these strategies. This allows us to conclude that, similar to games with infinitary parity objectives, the winning sets for games with finitary parity objectives can be decided in NP ∩ coNP.

In addition to finitary parity, we study finitary Streett objectives. Streett objectives require that if some stimuli are repeated infinitely often, then the corresponding responses occur infinitely often. The finitary interpretation requires, in addition, that there exists a bound b on all required response times (i.e., on the number of transitions between stimulus and corresponding response). We show that games with finitary Streett objectives can be solved by a reduction to finitary parity objectives (on a different game graph). The reduction establishes that games with finitary Streett objectives are determined. It also gives an algorithm that computes the winning sets in time $(n \cdot d!)^{O(d)} \cdot O(m)$ for game graphs with n states, m edges, and finitary Streett objectives with d pairs. Hence, the winning sets can be decided in EXPTIME. The decision problem for winning sets for games with infinitary Streett objectives is coNP-complete [5], and the winning sets can be computed in time $O(n^d \cdot d! \cdot m)$ [7]. For classical as well as finitary Streett games, finite-memory winning strategies exist for player 1. However, while in the classical case memoryless winning strategies exist for player 2 [5], in the finitary case the winning strategies for player 2 may require infinite memory.

We focus on finitary parity and Streett objectives. The finitary parity objectives are a canonical form to express finitary versions of ω-regular objectives; they subsume finitary reachability, finitary Büchi, and finitary co-Büchi objectives as special cases. The Streett objectives capture liveness conditions that are of particular interest in system design, as they correspond to strong fairness (compassion) constraints [9]. The finitary Streett objectives, therefore, give the finitary formulation of strong fairness.

2 Games with ω-Regular Objectives

Game graphs. A *game graph* $G = ((S, E), (S_1, S_2))$ consists of a directed graph (S, E) with a finite state space S and a set E of edges, and a partition (S_1, S_2) of the state space S into two sets. The states in S_1 are player 1 states, and the states in S_2 are player 2 states. For a state $s \in S$, we write $E(s) = \{t \in S \mid (s, t) \in E\}$ for the set of successor states of s. We assume that every state has at least one out-going edge, i.e., $E(s)$ is non-empty for all states $s \in S$.

Plays. A game is played by two players: player 1 and player 2, who form an infinite path in the game graph by moving a token along edges. They start by placing the token on an initial state, and then they take moves indefinitely in the following way. If the token is on a state in S_1, then player 1 moves the token along one of the edges going out of the state. If the token is on a state in S_2, then player 2 does likewise. The result is an infinite path in the game graph; we refer to such infinite paths as plays. Formally, a *play* is an infinite sequence $\langle s_0, s_1, s_2, \ldots \rangle$ of states such that $(s_k, s_{k+1}) \in E$ for all $k \geq 0$. We write Ω for the set of all plays.

Strategies. A strategy for a player is a recipe that specifies how to extend plays. Formally, a *strategy* σ for player 1 is a function σ: $S^* \cdot S_1 \to S$ that, given a finite sequence of states (representing the history of the play so far) which ends in a player 1 state, chooses the next state. The strategy must choose only available successors, i.e., for all $w \in S^*$ and $s \in S_1$, if $\sigma(w \cdot s) = t$, then $t \in E(s)$. The strategies for player 2 are defined analogously. We write Σ and Π for the sets of all strategies for player 1 and player 2, respectively. Strategies in general require memory to remember the history of plays. An equivalent definition of strategies is as follows. Let M be a set called *memory.* A strategy with memory can be described as a pair of functions: (a) a *memory-update* function σ_u: $S \times M \to M$ that, given the memory and the current state, updates the memory; and (b) a *next-state* function σ_n: $S \times M \to S$ that, given the memory and the current state, specifies the successor state. The strategy is *finite-memory* if the memory M is finite. The strategy is *memoryless* if the memory M is a singleton set. The memoryless strategies do not depend on the history of a play, but only on the current state. Each memoryless strategy for player 1 can be specified as a function σ: $S_1 \to S$ such that $\sigma(s) \in E(s)$ for all $s \in S_1$, and analogously for memoryless player 2 strategies. Given a starting state $s \in S$, a strategy $\sigma \in \Sigma$ for player 1, and a strategy $\pi \in \Pi$ for player 2, there is a unique play, denoted $\omega(s, \sigma, \pi) = \langle s_0, s_1, s_2, \ldots \rangle$, which is defined as follows: $s_0 = s$ and for all $k \geq 0$, if $s_k \in S_1$, then $\sigma(s_0, s_1, \ldots, s_k) = s_{k+1}$, and if $s_k \in S_2$, then $\pi(s_0, s_1, \ldots, s_k) = s_{k+1}$.

Classical winning conditions. We first define the class of ω-regular objectives and the classical notion of winning.

Objectives. Objectives for the players in non-terminating games are specified by providing the sets $\Phi, \Psi \subseteq \Omega$ of *winning plays* for player 1 and player 2, respectively. We consider zero-sum games, where the objectives of both players are complementary, i.e., $\Psi = \Omega \setminus \Phi$. The class of *$\omega$-regular objectives* [13] are of special interest since they form a robust class of objectives for verification and synthesis. The ω-regular objectives, and subclasses thereof, can be specified in the following forms. For a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega$, we define $\mathrm{Inf}(\omega) = \{s \in S \mid s_k = s \text{ for infinitely many } k \geq 0\}$ to be the set of states that occur infinitely often in ω.

1. *Reachability and safety objectives.* Given a set $F \subseteq S$ of states, the reachability objective Reach(F) requires that some state in F be visited, and dually, the safety objective Safe(F) requires that only states in F be visited. Formally, the sets of winning plays are $\mathrm{Reach}(F) = \{\langle s_0, s_1, s_2, \ldots \rangle \in \Omega \mid \exists k \geq 0.\ s_k \in F\}$ and $\mathrm{Safe}(F) = \{\langle s_0, s_1, s_2, \ldots \rangle \in \Omega \mid \forall k \geq 0.\ s_k \in F\}$.
2. *Büchi and co-Büchi objectives.* Given a set $F \subseteq S$ of states, the Büchi objective Buchi(F) requires that some state in F be visited infinitely often, and dually, the co-Büchi objective coBuchi(F) requires that only states in F be visited infinitely often. Thus, the sets of winning plays are $\mathrm{Buchi}(F) = \{\omega \in \Omega \mid \mathrm{Inf}(\omega) \cap F \neq \emptyset\}$ and $\mathrm{coBuchi}(F) = \{\omega \in \Omega \mid \mathrm{Inf}(\omega) \subseteq F\}$.
3. *Rabin and Streett objectives.* Given a set $P = \{(E_1, F_1), \ldots, (E_d, F_d)\}$ of pairs of sets of states (i.e, for all $1 \leq j \leq d$, both $E_j \subseteq S$ and $F_j \subseteq S$), the

Rabin objective Rabin(P) requires that for some pair $1 \leq j \leq d$, all states in E_j be visited finitely often, and some state in F_j be visited infinitely often. Hence, the winning plays are Rabin(P) $= \{\omega \in \Omega \mid \exists 1 \leq j \leq d.\ (\mathrm{Inf}(\omega) \cap E_j = \emptyset$ and $\mathrm{Inf}(\omega) \cap F_j \neq \emptyset)\}$. Dually, given $P = \{(E_1, F_1), \ldots, (E_d, F_d)\}$, the Streett objective Streett(P) requires that for all pairs $1 \leq j \leq d$, if some state in F_j is visited infinitely often, then some state in E_j be visited infinitely often, i.e., Streett(P) $= \{\omega \in \Omega \mid \forall 1 \leq j \leq d.\ (\mathrm{Inf}(\omega) \cap E_j \neq \emptyset$ or $\mathrm{Inf}(\omega) \cap F_j = \emptyset)\}$.

4. *Parity objectives.* Given a function p: $S \to \{0, 1, 2, \ldots, d-1\}$ that maps every state to an integer *priority*, the parity objective Parity(p) requires that of the states that are visited infinitely often, the least priority be even. Formally, the set of winning plays is Parity(p) $= \{\omega \in \Omega \mid \min\{p(\mathrm{Inf}(\omega))\}$ is even$\}$. The dual, co-parity objective has the set coParity(p) $= \{\omega \in \Omega \mid \min\{p(\mathrm{Inf}(\omega))\}$ is odd$\}$ of winning plays.

Every parity objective is both a Rabin objective and a Streett objective. Hence, the parity objectives are closed under complementation. The Büchi and co-Büchi objectives are special cases of parity objectives with two priorities, namely, p: $S \to \{0, 1\}$ for Büchi objectives with $F = p^{-1}(0)$, and p: $S \to \{1, 2\}$ for co-Büchi objectives with $F = p^{-1}(2)$. The reachability and safety objectives can be turned into Büchi and co-Büchi objectives, respectively, on slightly modified game graphs.

Winning. Given an objective $\Phi \subseteq \Omega$ for player 1, a strategy $\sigma \in \Sigma$ is a *winning strategy* for player 1 from a set $U \subseteq S$ of states if for all player 2 strategies $\pi \in \Pi$ and all states $s \in U$, the play $\omega(s, \sigma, \pi)$ is winning, i.e., $\omega(s, \sigma, \pi) \in \Phi$. The winning strategies for player 2 are defined analogously. A state $s \in S$ is winning for player 1 with respect to the objective Φ if player 1 has a winning strategy from $\{s\}$. Formally, the set of *winning states* for player 1 with respect to the objective Φ is $W_1(\Phi) = \{s \in S \mid \exists \sigma \in \Sigma.\ \forall \pi \in \Pi.\ \omega(s, \sigma, \pi) \in \Phi\}$. Analogously, the set of winning states for player 2 with respect to an objective $\Psi \subseteq \Omega$ is $W_2(\Psi) = \{s \in S \mid \exists \pi \in \Pi.\ \forall \sigma \in \Sigma.\ \omega(s, \sigma, \pi) \in \Psi\}$. We say that there exists a (memoryless; finite-memory) winning strategy for player 1 with respect to the objective Φ if there exists such a strategy from the set $W_1(\Phi)$; and similarly for player 2.

Theorem 1 (Classical determinacy and strategy complexity).

1. [6] *For all game graphs, all Rabin objectives Φ for player 1, and the complementary Streett objective $\Psi = \Omega \setminus \Phi$ for player 2, we have $W_1(\Phi) = S \setminus W_2(\Psi)$.*
2. [5] *For all game graphs and all Rabin objectives for player 1, there exists a memoryless winning strategy for player 1.*
3. [6] *For all game graphs and all Streett objectives for player 2, there exists a finite-memory winning strategy for player 2. However, in general no memoryless winning strategy exists.*

3 Finitary Winning Conditions

We now define a stronger notion of winning, namely, *finitary winning*, in games with parity and Streett objectives.

Finitary winning for parity objectives. For parity objectives, the finitary winning notion requires that for each visit to an odd priority that is visited infinitely often, the distance to a stronger (i.e., lower) even priority be bounded. To define the winning plays formally, we need the concept of a distance sequence.

Distance sequences for parity objectives. Given a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle$ and a priority function p: $S \to \{0, 1, \ldots, d-1\}$, we define a sequence of distances $dist_k(\omega, p)$, for all $k \geq 0$, as follows: $dist_k(\omega, p) = 0$ if $p(s_k)$ is even, and $dist_k(\omega, p) = \inf\{k' \geq k \mid p(s_{k'})$ is even and $p(s_{k'}) < p(s_k)\}$ if $p(s_k)$ is odd. Intuitively, the distance for a position k in a play with an odd priority at position k, denotes the shortest distance to a stronger even priority in the play. We assume the standard convention that the infimum of the empty set is ∞.

Finitary parity objectives. The finitary parity objective finParity(p) for a priority function p requires that the sequence of distances for the positions with odd priorities that occur infinitely often be bounded. This is equivalent to requiring that the sequence of all distances be bounded in the limit, and captures the notion that the "good" (even) priorities that appear infinitely often do not appear infinitely rarely. Formally, the sets of winning plays for the finitary parity objective and its complement are $\text{finParity}(p) = \{\omega \in \Omega \mid \limsup_{k\to\infty} dist_k(\omega, p) < \infty\}$ and $\text{infParity}(p) = \{\omega \in \Omega \mid \limsup_{k\to\infty} dist_k(\omega, p) = \infty\}$, respectively. Observe that if a play ω is winning for a co-parity objective, then the lim sup of the distance sequence for ω is ∞, that is, $\text{coParity}(p) \subseteq \text{infParity}(p)$. However, if a play ω is winning for a (classical) parity objective, then the lim sup of the distance sequence for ω can be ∞ (as shown in Example 1), that is, $\text{finParity}(p) \subsetneq \text{Parity}(p)$. Given a game graph G and a priority function p, solving the finitary parity game requires computing the two winning sets $W_1(\text{finParity}(p))$ and $W_2(\text{infParity}(p))$.

Remark 1. Recall that Büchi and co-Büchi objectives correspond to parity objectives with two priorities. A finitary Büchi objective is in general a strict subset of the corresponding classical Büchi objective; a finitary co-Büchi objective coincides with the corresponding classical co-Büchi objective. However, it can be shown that for parity objectives with two priorities, the classical winning sets and the finitary winning sets are the same; that is, for all game graphs G and all priority functions p with two priorities, we have $W_1(\text{finParity}(p)) = W_1(\text{Parity}(p))$ and $W_2(\text{infParity}(p)) = W_2(\text{coParity}(p))$. Note that in Example 1, we have $s_0 \in W_1(\text{Parity}(p))$ and $s_0 \notin W_1(\text{finParity}(p))$. This shows that for priority functions with three or more priorities, the winning set for a finitary parity objective can be a strict subset of the winning set for the corresponding classical parity objective, that is, $W_1(\text{finParity}(p)) \subsetneq W_1(\text{Parity}(p))$. ∎

Finitary winning for Streett objectives. The notion of distance sequence for parity objectives has a natural extension to Streett objectives.

Distance sequences for Streett objectives. Given a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle$ and a set $P = \{(E_1, F_1), \ldots, (E_d, F_d)\}$ of Streett pairs of state sets, the d sequences of distances $dist_k^j(\omega, P)$, for all $k \geq 0$ and $1 \leq j \leq d$, are defined as follows: $dist_k^j(\omega, P) = 0$ if $s_k \notin F_j$, and $dist_k^j(\omega, P) = \inf\{k' \geq k \mid s_{k'} \in E_j\}$ if $s_k \in F_j$. Let $dist_k(\omega, P) = \max\{dist_k^j(\omega, P) \mid 1 \leq j \leq d\}$ for all $k \geq 0$.

Finitary Streett objectives. The finitary Streett objective finStreett(P) for a set P of Streett pairs requires that the distance sequence be bounded in the limit, i.e., the winning plays are finStreett$(P) = \{\omega \in \Omega \mid \limsup_{k \to \infty} dist_k(\omega, P) < \infty\}$.

Example 2. Consider the game graph of Figure 2. Player 2 generates requests of type Req_1 and Req_2; these are shown as labeled edges in the figure. Player 1 services a request of type Req_i by choosing an edge labeled $Serv_i$, for $i = 1, 2$. Whenever a request is received, further requests of the same type are disabled until the request is serviced; then the requests of this type are enabled again. The state s_0 represents the case when there are no unserviced requests; the states s_1 and s_2 represent the cases when there are unserviced requests of type Req_1 and Req_2, respectively; and the states s_7 and s_8 represent the cases when there are unserviced requests of both types, having arrived in either order. On arrival of a request of type Req_i, a state in F_i is visited, and when a request of type Req_i is serviced, a state in E_i is visited, for $i = 1, 2$. Hence $F_1 = \{s_1, s_8\}$, $F_2 = \{s_2, s_7\}$, $E_1 = \{s_5, s_{12}\}$, and $E_2 = \{s_6, s_{11}\}$. The Streett objective Streett(P) with $P = \{(E_1, F_1), (E_2, F_2)\}$ requires that if a request of type Req_i is received infinitely often, then it be serviced infinitely often, for both $i = 1, 2$. The player 1 strategy $s_9 \to s_{11}$ and $s_{10} \to s_{12}$ is a *stack* strategy, which always services first the request type received last. The player 1 strategy $s_9 \to s_{12}$ and $s_{10} \to s_{11}$ is a *queue* strategy, which always services first the request type received first. Both the stack strategy and the queue strategy ensure that the classical Streett objective Streett(P) is satisfied. However, for the stack strategy, the number of transitions between the arrival of a request of type Req_i and its service can be unbounded. Hence the stack strategy is not a winning strategy for player 1 with respect to the finitary Streett objective finStreett(P). The queue strategy, by contrast, ensures not only that every request that is received infinitely often is serviced, but it also ensures that the number of transitions between the arrival

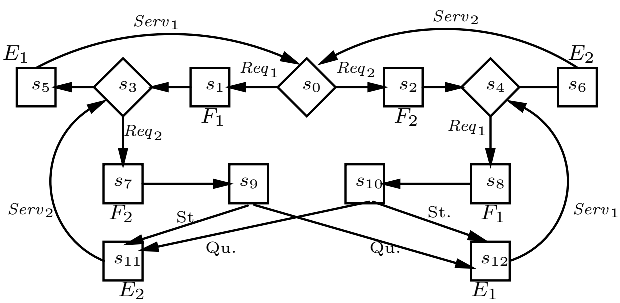

Fig. 2. A request-service game graph

of a request and its service is at most 6. Thus the queue strategy is winning for player 1 with respect to $\text{finStreett}(P)$. ∎

4 Finitary Determinacy and Algorithmic Analysis

We present an algorithm to solve games with finitary parity objectives. The correctness argument for the algorithm also proves determinacy for finitary parity games.[1] We then show that games with finitary Streett objectives can be solved via a reduction to finitary parity games.

Solving games with finitary parity objectives. We start with some preliminary notation and facts that will be required for the analysis of the algorithm.

Closed sets. A set $U \subseteq S$ of states is a *closed set* for player 2 if the following two conditions hold: (a) for all states $u \in (U \cap S_2)$, we have $E(u) \subseteq U$, i.e., all successors of player 2 states in U are again in U; and (b) for all $u \in (U \cap S_1)$, we have $E(u) \cap U \neq \emptyset$, i.e., every player 1 state in U has a successor in U. The closed sets for player 1 are defined analogously. Every closed set U for player ℓ, for $\ell \in \{1, 2\}$, induces a sub-game graph, denoted $G \upharpoonright U$. For winning sets W_1 and W_2, we write W_1^G and W_2^G to explicitly specify the game graph G.

Proposition 1. *Consider a game graph G, and a closed set U for player 2. For every objective Φ for player 1, we have $W_1^{G \upharpoonright U}(\Phi) \subseteq W_1^G(\Phi)$.*

Attractors. Given a game graph G, a set $U \subseteq S$ of states, and a player $\ell \in \{1, 2\}$, the set $Attr_\ell(U, G)$ contains the states from which player ℓ has a strategy to reach a state in U against all strategies of the other player; that is, $Attr_\ell(U, G) = W_\ell^G(\text{Reach}(U))$. The set $Attr_1(U, G)$ can be computed inductively as follows: let $R_0 = U$; let $R_{i+1} = R_i \cup \{s \in S_1 \mid E(s) \cap R_i \neq \emptyset\} \cup \{s \in S_2 \mid E(s) \subseteq R_i\}$ for all $i \geq 0$; then $Attr_1(U, G) = \bigcup_{i \geq 0} R_i$. The inductive computation of $Attr_2(U, G)$ is analogous. For all states $s \in Attr_1(U, G)$, define $rank(s) = i$ if $s \in R_i \setminus R_{i-1}$, that is, $rank(s)$ denotes the least $i \geq 0$ such that s is included in R_i. Define a memoryless strategy $\sigma \in \Sigma$ for player 1 as follows: for each state $s \in (Attr_1(U, G) \cap S_1)$ with $rank(s) = i$, choose a successor $\sigma(s) \in (R_{i-1} \cap E(s))$ (such a successor exists by the inductive definition). It follows that for all states $s \in Attr_1(U, G)$ and all strategies $\pi \in \Pi$ for player 2, the play $\omega(s, \sigma, \pi)$ reaches U in at most $|Attr_1(U, G)|$ transitions.

Proposition 2. *For all game graphs G, all players $\ell \in \{1, 2\}$, and all sets $U \subseteq S$ of states, the set $S \setminus Attr_\ell(U, G)$ is a closed set for player ℓ.*

Notation. Given a priority function p: $S \to \{0, 1, \ldots, d-1\}$, and a priority $j \in \{0, 1, \ldots, d-1\}$, we write $p^{-1}(j) \subseteq S$ for the set of states with priority j. For $\bowtie \in \{<, \leq, >, \geq\}$, let $p^{-1}(\bowtie j) = \bigcup_{j' \bowtie j} p^{-1}(j')$. Moreover, let

[1] The determinacy of games with finitary parity objectives can also be proved by reduction to Borel objectives, using the determinacy of Borel games [10]; however, our proof is direct.

$even(p) = \bigcup_{2j<d} p^{-1}(2j)$ be the set of states with even priorities. We define the set $\text{ReachSafe}(p) = \bigcup_{2j+1<d}(\text{Reach}(p^{-1}(2j+1)) \cap \text{Safe}(p^{-1}(\geq 2j+1)))$ of plays, i.e., the objective ReachSafe(p) requires that a state of some odd priority $2j+1$ be reached, and that the play stay within the set of states with priorities at least $2j+1$. The complementary objective is $\Omega \setminus \text{ReachSafe}(p) = \bigcap_{2j+1<d}((\Omega \setminus \text{Reach}(p^{-1}(2j+1))) \cup \text{Reach}(p^{-1}(\leq 2j) \cap even(p)))$.

Informal description of Algorithm 1. The algorithm takes as input a game graph G and a priority function $p: S \to \{0, 1, \ldots, d-1\}$ with d priorities. The algorithm iteratively computes the winning sets $W_1(\text{finParity}(p))$ and $W_2(\text{infParity}(p))$ for player 1 and player 2, respectively. We describe one iteration of the algorithm (i.e., one execution of the loop body at Step 2). Let G^i be the game graph at iteration i, and let (S^i, E^i) be the underlying directed graph. In Step 2.1, the set $A = Attr_1(p^{-1}(0), G^i)$ is computed as the set of states from which player 1 can reach a state of priority 0. In the sub-game $G^i \upharpoonright B$, where $B = S^i \setminus A$, the set C denotes the set of player 2 states that have an edge into A. In Step 2.3, the set $D = Attr_2(C, G^i \upharpoonright B)$ is computed and the sub-game $G^i \upharpoonright H$ is solved recursively, where $H = B \setminus D$. If a non-empty player 1 winning set U_1 is discovered in the sub-game $G^i \upharpoonright H$, then U_1 and $Attr_1(U_1, G^i)$ are identified as subsets of $W_1(\text{finParity}(p))$, removed from G^i, and the algorithm proceeds to iteration $i+1$ (Step 2.5). Otherwise, the game graph $G^i \upharpoonright B$ is solved with the objective ReachSafe(p) for player 2 (and the complementary objective for player 1). If the winning set for player 2 is empty, then all of S^i is identified as a subset of $W_1(\text{finParity}(p))$, and the algorithm stops (Step 2.7). Otherwise, let X_2 be the winning set for player 2 in the sub-game $G^i \upharpoonright B$ with respect to the objective ReachSafe(p), and let $L = Attr_2(X_2, G^i)$. The sub-game $G^i \upharpoonright Q$ is solved recursively, where $Q = S^i \setminus L$. If a non-empty player 1 winning set Z_1 is discovered in $G^i \upharpoonright Q$, then Z_1 and $Attr_1(Z_1, G^i)$ are identified as subsets of $W_1(\text{finParity}(p))$, removed from G^i, and the algorithm proceeds to iteration $i+1$ (Step 2.8.3). Otherwise, all of S^i is identified as a subset of $W_2(\text{infParity}(p))$, and the algorithm stops (Step 2.8.4).

Claim 1: Correctness of Step 2.5. We first argue that the set H defined in Step 2.3 is a closed set for player 2. Observe that for all states $s \in (S_2 \cap B)$, if $E(s)$ is not a subset of B, then $s \in C$. Hence for all states in $s \in B \setminus C$, we have $E(s) \subseteq B$. It follows from Proposition 2 that $H = B \setminus Attr_2(C, G^i \upharpoonright B)$ is a player 2 closed set in the game graph G^i. It follows from Proposition 1 that the set $U_1 = W_1^{G^i \upharpoonright H}(\text{finParity}(p))$ in the sub-game $G^i \upharpoonright H$ is winning for player 1. Hence U_1 and $Attr_1(U_1, G^i)$ are correctly identified as subsets of the player 1 winning set $W_1(\text{finParity}(p))$. ∎

Claim 2: Correctness of Step 2.7. Observe that if Step 2.7 is executed, then $X_2 = \emptyset$, and hence player 1 wins with respect to the objective $\Phi = \bigcap_{2j+1 \leq d}((\Omega \setminus \text{Reach}(p^{-1}(2j+1))) \cup \text{Reach}(p^{-1}(\leq 2j) \cap even(p)))$ from every state $s \in B$ in the sub-game $G^i \upharpoonright B$. Recall that $B = S^i \setminus A$, and $A = Attr_1(p^{-1}(0), G^i)$. It follows that player 1 wins with respect to the objective Φ from all states $s \in S^i$

Algorithm 1. FinitaryParity

Input: a game graph G and a priority function p.
Output: the sets $W_1 = W_1(\text{finParity}(p))$ and $W_2 = W_2(\text{infParity}(p))$.

1. $W_1 = \emptyset$; $W_2 = \emptyset$; $G^0 = G$; $i = 0$;
2. **repeat**
   2.1. $A = Attr_1(p^{-1}(0), G^i)$; $B = S^i \setminus A$;
   2.2. $C = \{s \in (B \cap S_2) \mid E^i(s) \cap A \neq \emptyset\}$;
   2.3. $D = Attr_2(C, G^i \upharpoonright B)$; $H = B \setminus D$;
   2.4. $(U_1, U_2) = \text{FinitaryParity}(G^i \upharpoonright H, p)$;
   2.5. **if** $U_1 \neq \emptyset$ **then**
      2.5.1. $W_1 = W_1 \cup Attr_1(U_1, G^i)$; $S^{i+1} = S^i \setminus Attr_1(U_1, G^i)$;
      2.5.2. **goto** Step 2.9;
   2.6. $(X_1, X_2) = \text{GameSolve}(G^i \upharpoonright B, \Omega \setminus \text{ReachSafe}(p))$;
   2.7. **if** $X_2 = \emptyset$ **then**
      2.7.1. $W_1 = W_1 \cup S^i$;
      2.7.2. **return** (W_1, W_2);
   2.8. **else**
      2.8.1. $L = Attr_2(X_2, G^i)$; $Q = S^i \setminus L$;
      2.8.2. $(Z_1, Z_2) = \text{FinitaryParity}(G^i \upharpoonright Q, p)$;
      2.8.3. **if** $Z_1 \neq \emptyset$ **then**
         2.8.3.1. $W_1 = W_1 \cup Attr_1(Z_1, G^i)$; $S^{i+1} = S^i \setminus Attr_1(Z_1, G^i)$;
         2.8.3.2. **goto** Step 2.9;
      2.8.4. **else**
         2.8.4.1. $W_2 = W_2 \cup S^i$;
         2.8.4.2. **return** (W_1, W_2);
   2.9. $G^{i+1} = G^i \upharpoonright S^{i+1}$; $i := i + 1$;
   until $S^i = \emptyset$;
3. **return** (W_1, W_2).

in the game graph G^i. Hence $p^{-1}(2j+1) \subseteq Attr_1(p^{-1}(\leq 2j) \cap even(p), G^i)$ for all $2j+1 < d$. We inductively define the following sets: let $A_0 = A = Attr_1(p^{-1}(0), G^i)$; and let $A_{2j} = Attr_1(p^{-1}(2j), G^i \upharpoonright (S^i \setminus A_{2j-2}))$ for $2 \leq 2j < d$. Observe that $p^{-1}(2j+1) \subseteq \bigcup_{j' \leq j} A_{2j'}$. A memoryless strategy σ for player 1 can be constructed as follows: in A_0, reach $p^{-1}(0)$ within $|A_0|$ transitions; and in the sub-game $G^i \upharpoonright (S^i \setminus A_{2j-2})$, reach $p^{-1}(2j)$ within $|A_{2j}|$ transitions from all states in A_{2j}. If player 1 follows the strategy σ, then for all player 2 strategies π, if the play visits a state in $p^{-1}(2j+1)$, then it visits $p^{-1}(\leq 2j) \cap even(p)$ within $|S^i|$ transitions. Thus, for all states s and all player 2 strategies π, we have $dist_k(\omega(s,\sigma,\pi), p) \leq |S^i|$ for all $k \geq 0$, and therefore $\limsup_{k\to\infty} dist_k(\omega(s,\sigma,\pi), p) < \infty$. ∎

Claim 3: Correctness of Step 2.8.3. Observe that $L = Attr_2(X_2, G^i)$, and hence $Q = S^i \setminus L$ is a closed set for player 2 (by Proposition 2). It follows from arguments similar to the correctness of Step 2.5 that Z_1 and $Attr_1(Z_1, G^i)$ are correctly identified as subsets of $W_1(\text{finParity}(p))$. ∎

Claim 4: Correctness of Step 2.8.4. Observe that if Step 2.8.4 is executed, then the following two conditions hold: (i) player 2 has a winning strategy π_H from H with respect to the objective infParity(p) in the sub-game $G^i \upharpoonright H$ (because the test of Step 2.5 fails); and (ii) player 2 has a winning strategy π_Q from Q with respect to the objective infParity(p) in the sub-game $G^i \upharpoonright Q$ (because the test of Step 2.8.3 fails). We construct a winning strategy π for player 2 from S^i which is played in rounds. In every round, there are five stages, and we describe each stage of the strategy π for round r as follows:

Stage 1. [play in Q] As long as the play stays in Q, play the strategy π_Q (the player 2 winning strategy from Q in the sub-game $G^i \upharpoonright Q$ with respect to the objective infParity(p)). If the play enters $L = S^i \setminus Q$, proceed to Stage 2.
Stage 2. [play in L] Play a strategy π_L to reach X_2 within $|L|$ transitions, and proceed to Stage 3 when the play reaches X_2.
Stage 3. [play in X_2] Play a winning strategy π_{X_2} with respect to the objective ReachSafe(p) in the sub-game $G^i \upharpoonright B$. After the play reaches a state in $p^{-1}(2j+1) \cap B$, and stays in $p^{-1}(\geq 2j+1) \cap B$ for r transitions, proceed to Stage 4.
Stage 4. [play in H] As long as the play stays in H, play the strategy π_H (the player 2 winning strategy from H in the sub-game $G^i \upharpoonright H$ with respect to the objective infParity(p)). If the play enters $D = B \setminus H$, proceed to Stage 5.
Stage 5. [play in D] Play a strategy to reach C within $|D|$ transitions, then leave B via an edge from C to A, and proceed to Stage 1 of round $r+1$.

Given the player 2 strategy π, consider a player 1 strategy σ and a state $s \in S^i$. Observe that if the play $\omega(s,\sigma,\pi)$ reaches Stage 2 of a round r, then Stages 3 and 4 of round r are also reached. Similarly, if stage 5 of round r is reached, then Stages 1 or 2 of round $r+1$ are also reached. If the play $\omega(s,\sigma,\pi)$ remains forever in Stage 1 or Stage 4 for some round r, then by properties of π_H and π_Q (conditions (i) and (ii) from above), it follows that $\omega(s,\sigma,\pi) \in$ infParity(p). Otherwise, the play proceeds through infinitely many rounds. Stage 3 of the strategy π ensures that in round r, there is a position $k \geq 0$ such that $dist_k(\omega(s,\sigma,\pi),p) \geq r$. Hence it follows that $\limsup_{k\to\infty} dist_k(\omega(s,\sigma,\pi),p) = \infty$, and thus again $\omega(s,\sigma,\pi) \in$ infParity(p). ∎

The claims 1–4 establish the correctness of Algorithm 1, and also establish the determinacy of games with finitary parity objectives.

Theorem 2 (Finitary determinacy). *For all game graphs and all priority functions p, we have $W_1(\text{finParity}(p)) = S \setminus W_2(\text{infParity}(p))$.*

Running time of Algorithm 1. Recall from Remark 1 that for priority functions p with two priorities, the winning sets for the classical parity objective Parity(p) and for the finitary parity objective finParity(p) coincide. Hence, for two priorities the winning set $W_1(\text{finParity}(p))$ can be computed in $O(n \cdot m)$ time, where n is the number of states and m is the number of edges (by algorithms for solving Büchi and co-Büchi games). For priority functions with d priorities, let $T(n,m,d)$ be the running time of Algorithm 1 to compute $W_1(\text{finParity}(p))$ and

$W_2(\text{infParity}(p))$. The running time of one iteration of the algorithm (i.e., one execution of the loop body at Step 2) can be bounded by $n \cdot (T(n, m, d-1) + O(d \cdot m))$. Since in each iteration at least one state is removed from the game graph, we obtain the recurrence $T(n, m, d) = n^2 \cdot (T(n, m, d-1) + O(d \cdot m))$ for $d > 2$. This yields the time bound in the following theorem.

Theorem 3 (Algorithm 1). *Given a game graph with n states and m edges, and a priority function p with d priorities, Algorithm 1 computes the winning sets $W_1(\text{finParity}(p))$ and $W_2(\text{infParity}(p))$ in $O\big(n^{2d-3} \cdot d \cdot m\big)$ time.*

Winning strategies for finitary parity objectives. We first show that winning strategies for player 2 with respect to the objective infParity(p) in general require infinite memory. To see this, recall Example 1: the player 2 winning strategy π_w constructed in the example requires infinite memory, and against any finite-memory strategy π_f for player 2, the player 1 strategy σ that chooses the successor s_2 at state s_0, ensures that in the play $\omega(s_0, \sigma, \pi_f)$ the distances between states of priority 1 and priority 0 are always bounded. In contrast to winning strategies for player 2, which may require infinite memory, we now argue that memoryless winning strategies exist for player 1. This follows from the analysis of Steps 2.5, 2.7, and 2.8.3 of Algorithm 1. In the correctness argument for Step 2.7, a memoryless winning strategy is constructed. The existence of memoryless winning strategies for Steps 2.5 and 2.8.3 follow from inductive arguments (induction on the number of priorities for Step 2.5, and induction on the size of the state space for Step 2.8.3).

Witness sizes for winning strategies. Since memoryless winning strategies exist for player 1, there exist polynomial-size witnesses (in fact, linear-size witnesses) for player 1 winning strategies. We now argue that although player 2 winning strategies may require infinite memory, there exist polynomial-size witnesses for these strategies as well. Consider the correctness argument for Step 2.8.4 of Algorithm 1. The sets used in the analysis can serve as witness for the player 2 winning strategy. The witness consists of the following components: (a) the sets $A = \mathit{Attr}_1(p^{-1}(0), G^i)$ and $B = S^i \setminus A$; (b) the sets C and $D = \mathit{Attr}_2(C, G^i \restriction B)$ and $H = B \setminus D$; (c) the set $X_2 = W_2^{G^i \restriction B}(\text{ReachSafe}(p))$, and a player 2 winning strategy in X_2 with respect to the objective ReachSafe(p); (d) the set $L = \mathit{Attr}_2(X_2, G^i)$, and a player 2 winning strategy in L to reach X_2; and (e) player 2 winning strategies in the sub-games $G^i \restriction H$ and $G^i \restriction Q$. Given such a witness, the existence of a player 2 winning strategy follows from the construction presented in the correctness argument for Step 2.8.4. It is easy to argue that linear-size witnesses exist for parts (a)–(d). The witness for part (e) is recursive. A key observation to obtain a polynomial-size witness is the following: in Stage 1 of the strategy construction, in the set $Q \cap H$ player 2 can follow the winning strategy in H of the sub-game $G^i \restriction H$. Hence the witness in Q can follow the witness of H in the set $Q \cap H$, and we need to exhibit a different witness in Q only for the subset that is disjoint from H. Let $\mathit{Size}(t)$ denote the size of the witness for a set of size t. Hence the witness consists of the witness in H of $\mathit{Size}(|H|)$, the witness in Q of size $\mathit{Size}(|Q \setminus H|)$, and witnesses of linear size. Thus we have the recurrence $\mathit{Size}(n) \leq \max\{\mathit{Size}(h) + \mathit{Size}(n-h) + O(n) \mid 1 \leq h \leq n\}$, where h denotes the size of the set H, that is, $h = |H|$. This recurrence is satisfied by $\mathit{Size}(n) = O(n^2)$.

Theorem 4 (Finitary strategy complexity). *For all game graphs and all priority functions p, there exists a memoryless winning strategy for player 1 with respect to the objective* $\mathrm{finParity}(p)$. *However, in general no finite-memory winning strategy exists for player 2 with respect to the complementary objective* $\mathrm{infParity}(p)$. *For game graphs with n states, there are witnesses of size $O(n)$ and $O(n^2)$ for the winning strategies for player 1 and player 2, respectively.*

Computational complexity. The existence of memoryless winning strategies for player 1 implies that whether a given state lies in $W_1(\mathrm{finParity}(p))$ can be decided in NP. Moreover, because of the existence of polynomial-size witnesses for player 2 winning strategies, also whether a given state lies in $W_2(\mathrm{infParity}(p))$ can be decided in NP.

Corollary 1. *For all game graphs, all priority functions p, and all states s, whether $s \in W_1(\mathrm{finParity}(p))$ can be decided in* $\mathrm{NP} \cap \mathrm{coNP}$.

It remains an open problem if there is a polynomial-time algorithm to compute $W_1(\mathrm{finParity}(p))$. The existence of memoryless winning strategies for finitary parity objectives also gives the following refined characterization of the winning set, which shows that distances can be bounded by the size of the state space.

Corollary 2. *For all game graphs with n states, and all priority functions p, we have $W_1(\mathrm{finParity}(p)) = \{s \in S \mid \exists \sigma \in \Sigma.\ \forall \pi \in \Pi.\ \limsup_{k\to\infty} \mathit{dist}_k(\omega(s,\sigma,\pi), p) \leq n\}$.*

Solving games with finitary Streett objectives. The *index appearance record* (IAR) construction [13] translates games with player 1 Streett objectives into games with parity objectives, preserving the abilities of both players to win. Given a game graph G with n states and m edges, and a set $P = \{(E_1, F_1), \ldots, (E_d, F_d)\}$ of d Streett pairs, the IAR construction yields a game graph G' with $n \cdot d! \cdot d^2$ states and $m \cdot d! \cdot d^2$ edges, and a priority function p with $O(d)$ priorities. We only sketch the construction here. An IAR is a triple (τ, e, f), where τ is a permutation of $(1, 2, \ldots, d)$, and $e, f \in \{1, 2, \ldots, d\}$. The permuation τ remembers the order of the latest appearances of the sets E_j, for $1 \leq j \leq d$, and the indices e and f remember the previous positions in τ of the most recent sets E_j and F_j, respectively. The new game graph G' is obtained as the synchronous product of the original game graph G and the IAR; see [13]. For a state $\langle s, (\tau, e, f)\rangle$ of G' (where s is a state of G), the priority function p is defined such that $p(\langle s, (\tau, e, f)\rangle) = 2e$ if $f \leq e$, and otherwise $p(\langle s, (\tau, e, f)\rangle) = 2f - 1$. The IAR reduction from Streett to parity games ensures that for every play in G, the limsup of the Streett distance sequence is bounded by $d! \cdot d^2$ times the limsup of the parity distance sequence for the corresponding play in G'.

Theorem 5 (Finitary Streett games). *Given a game graph G with n states, m edges, and a set P of d Streett pairs, let G' be the game graph with $n \cdot d! \cdot d^2$*

states and $m \cdot d! \cdot d^2$ *edges, and let* p *be the corresponding priority function with* $O(d)$ *priorities, obtained by the IAR construction. For a play* ω' *in* G'*, let* ω *be the corresponding play in* G*. If* $\limsup_{k\to\infty} dist_k(\omega', p) = \alpha < \infty$*, then* $\limsup_{k\to\infty} dist_k(\omega, P) \leq \alpha \cdot d! \cdot d^2$*, and if* $\limsup_{k\to\infty} dist_k(\omega', p) = \infty$*, then* $\limsup_{k\to\infty} dist_k(\omega, P) = \infty$*.*

Hence solving games with finitary Streett objectives can be reduced to solving games with finitary parity objectives. Using Theorem 2, Theorem 3, and Theorem 4 we obtain the following corollary.

Corollary 3. *For all game graphs with* n *states and* m *edges, and all sets* P *of* d *Streett pairs, the following assertions hold.*

1. $W_1(\text{finStreett}(P)) = S \setminus W_2(\Omega \setminus \text{finStreett}(P))$.
2. $W_1(\text{finStreett}(P))$ *can be computed in* $O((n \cdot d! \cdot d^2)^{2d-3} \cdot m \cdot d! \cdot d^3)$ *time.*
3. *There exists a finite-memory winning strategy for player 1 with respect to the objective* finStreett(P)*. However, in general no finite-memory strategy exists for player 2 with respect to the complementary objective* $\Omega \setminus \text{finStreett}(P)$*.*

It follows that whether a state lies in $W_1(\text{finStreett}(P))$ can be decided in EXPTIME. The exact complexity of the problem remains open.

References

1. B. Alpern and F.B. Schneider. Defining liveness. *Information Processing Letters*, 21:181–185, 1985.
2. R. Alur and T.A. Henzinger. Finitary fairness. In *Proc. Logic in Computer Science*, pages 52–61. IEEE Computer Society, 1994.
3. R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *J. ACM*, 49:672–713, 2002.
4. N. Dershowitz, D.N. Jayasimha, and S. Park. Bounded fairness. In *Verification: Theory and Practice*, pages 304–317. LNCS 2772, Springer, 2003.
5. E.A. Emerson and C. Jutla. The complexity of tree automata and logics of programs. In *Proc. Foundations of Computer Science*, pages 328–337. IEEE Computer Society, 1988.
6. Y. Gurevich and L. Harrington. Trees, automata, and games. In *Proc. Symp. Theory of Computing*, pages 60–65. ACM, 1982.
7. F. Horn. Streett games on finite graphs. *Workshop on Games in Design and Verification*, 2005.
8. M. Jurdzinski. Small progress measures for solving parity games. In *Symp. Theoretical Aspects of Computer Science*, pages 290–301. LNCS 1770, Springer, 2000.
9. Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Specification.* Springer, 1992.
10. D.A. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975.
11. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *Proc. Principles of Programming Languages*, pages 179–190. ACM, 1989.
12. P.J. Ramadge and W.M. Wonham. Supervisory control of a class of discrete-event processes. *SIAM J. Control and Optimization*, 25:206–230, 1987.
13. W. Thomas. Languages, automata, and logic. In *Handbook of Formal Languages*, volume 3, pages 389–455. Springer, 1997.