Half-Positional Objectives Recognized by Deterministic Büchi Automata (Extended Abstract)*

Patricia Bouyer[1] , **Antonio Casares**[2] , **Mickael Randour**[3] and **Pierre Vandenhove**[1,3]
[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, Laboratoire Méthodes Formelles, 91190, Gif-sur-Yvette, France
[2]LaBRI, Université de Bordeaux, Bordeaux, France
[3]F.R.S.-FNRS & UMONS – Université de Mons, Mons, Belgium
bouyer@lmf.cnrs.fr, antonio.casares-santos@labri.fr,
{mickael.randour, pierre.vandenhove}@umons.ac.be

Abstract

In two-player zero-sum games on graphs, the protagonist tries to achieve an objective while the antagonist aims to prevent it. Objectives for which *both* players do not need to use memory to play optimally are well-understood and characterized both in finite and infinite graphs. Less is known about the larger class of *half-positional* objectives, i.e., those for which the protagonist does not need memory (but for which the antagonist might). In particular, no characterization of half-positionality is known for the central class of ω*-regular objectives*.

Here, we characterize objectives recognizable by deterministic Büchi automata (a class of ω-regular objectives) that are half-positional, both over finite and infinite graphs. This characterization yields a polynomial-time algorithm to decide half-positionality of an objective recognized by a given deterministic Büchi automaton.

1 Introduction

Graph Games and Reactive Synthesis. We study *zero-sum turn-based games on graphs* [Fijalkow *et al.*, 2023] confronting two players, $\mathcal{P}_1$ and $\mathcal{P}_2$. They interact by moving a pebble in turns through the edges of a graph, ad infinitum. Each vertex belongs to a player, and the owner of the current vertex decides where to go next. Edges of the graph are labeled with *colors*, and this interaction thus produces an infinite sequence of colors. The objective of the game is specified by a subset of infinite sequences of colors, and $\mathcal{P}_1$ wins if the produced sequence is in this set. We are interested in finding a *winning strategy* for $\mathcal{P}_1$, i.e., a function indicating how $\mathcal{P}_1$ should move in any situation, guaranteeing the achievement of the objective, whatever the strategy of $\mathcal{P}_2$.

This game-theoretic model is particularly fitted to study the *reactive synthesis problem* [Bloem *et al.*, 2018], which aims at the automated construction of a provably-correct controller for a system ($\mathcal{P}_1$) trying to satisfy a specification (the objective) while interacting continuously with an uncontrollable environment ($\mathcal{P}_2$). This comes down to finding a winning strategy for $\mathcal{P}_1$ in the derived game.

In general, in a graph game, a strategy may need *memory* in order to be winning. This means that only observing the current graph vertex may not yield sufficient information to make an optimal decision; additional information about the past of the interaction is also required. For instance, if there are two colors a and b, and the objective of $\mathcal{P}_1$ is to see twice the color a in a row, memory is needed to win in some game graphs, such as the one in Figure 1. From vertex v, $\mathcal{P}_1$ has a choice among ab and ba, and it is possible to win by playing any infinite word starting with $baab$. However, a strategy without memory (called *positional*) from v can only achieve the infinite words $baba\ldots$ or $abab\ldots$, both losing.

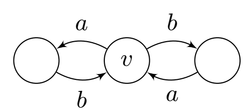


Figure 1: Memory is needed to see a twice in a row from v.

Some objectives do not need memory, no matter the game graph. This is for instance the case of the *Büchi objective* [Fijalkow *et al.*, 2023]: if the goal is to see a color infinitely often and there is a winning strategy for $\mathcal{P}_1$ in some game graph, then there is also a *positional* winning strategy for $\mathcal{P}_1$. This is a beneficial property to obtain a controller for the system that is as simple as possible to implement.

Half-Positionality. We intend to understand for which objectives positional (also called *memoryless*) strategies suffice for $\mathcal{P}_1$ to play optimally (i.e., to win whenever it is possible) — we call these objectives *half-positional*. We distinguish half-positionality from *bipositionality* (or *memoryless-determinacy*), which refers to objectives for which positional strategies suffice to play optimally for *both* players.

Many natural objectives have been shown to be bipositional over games on finite and sometimes infinite graphs: e.g., discounted sum [Shapley, 1953], mean-payoff [Ehren-

*Extended abstract of the eponymous paper from the proceedings of CONCUR 2022. This work has been supported by the ANR Project MAVeriQ (ANR-20-CE25-0012) and by the Fonds de la Recherche Scientifique – FNRS under Grant n° T.0188.23 (PDR ControlleRS). Mickael Randour is an F.R.S.-FNRS Research Associate and a member of the TRAIL Institute. Pierre Vandenhove is an F.R.S.-FNRS Research Fellow.

feucht and Mycielski, 1979], parity [Emerson and Jutla, 1991], total payoff [Gimbert and Zielonka, 2004], energy [Bouyer *et al.*, 2008], or average-energy games [Bouyer *et al.*, 2018]. Bipositionality can be established using general criteria and characterizations, over games on both finite graphs [Gimbert and Zielonka, 2004; Gimbert and Zielonka, 2005; Aminof and Rubin, 2017] and infinite graphs [Colcombet and Niwiński, 2006]. Yet, there exist many objectives and combinations thereof for which one player, but not both, has positional optimal strategies (Rabin conditions [Klarlund and Kozen, 1991; Klarlund, 1994], mean-payoff parity [Chatterjee *et al.*, 2005], energy parity [Chatterjee and Doyen, 2012]...), and to which these results do not apply.

Various attempts have been made to understand common underlying properties of half-positional objectives and provide sufficient conditions [Kopczyński, 2006; Kopczyński, 2007; Kopczyński, 2008; Bianco *et al.*, 2011]. These sufficient conditions are not general enough to prove half-positionality of some very simple objectives, even in the well-studied class of *ω-regular objectives* [Bianco *et al.*, 2011, Lemma 13]. An interesting characterization uses *universal graphs* [Ohlmann, 2023]; although it brings insight into the structure of half-positional objectives, showing half-positionality through the use of universal graphs is not always straightforward, and has not yet been applied in a systematic way to ω-regular objectives. The proof of our characterization makes use of this novel tool.

Furthermore, multiple questions concerning half-positionality remain open [Kopczyński, 2008]. For instance, it is still unclear how to decide half-positionality, even for ω-regular objectives in general. A result in this direction is given by Kopczyński [Kopczyński, 2007], who showed that half-positionality over finite graphs is decidable in exponential time for a subclass of the ω-regular objectives (incomparable to the one considered in this article). It is unknown whether this is doable in polynomial time, and no algorithm is known for half-positionality over infinite graphs.

Half-Positionality in RL. Reinforcement learning (RL) shares goals similar to synthesis, in that a strategy achieving some specification must be built. While it is common that synthesis considers strategies with memory, half-positionality of the objective is typically a requirement to apply RL algorithms, as decisions are usually taken simply based on the current state of the game [Sutton and Barto, 2018]. Given an objective and a graph, two steps can be taken to apply RL algorithms [Hahn *et al.*, 2022a]: (i) inject sufficient information in the graph to guarantee that positional strategies suffice, and (ii) label vertices/edges with rewards such that strategies winning for the objective correspond to optimal strategies w.r.t. RL. Half-positional objectives correspond to the objectives for which no information must be added in step (i). Given step (i) (on which we focus in this article), note that step (ii) is not always straightforward [Hahn *et al.*, 2022b].

Omega-Regular Objectives and Deterministic Büchi Automata. A central class of objectives, whose half-positionality is not yet completely understood, is the class of *ω-regular objectives*. There are multiple equivalent definitions for them: they are the objectives defined, e.g., by ω-regular expressions, by non-deterministic Büchi automata [McNaughton, 1966], and by deterministic parity automata [Mostowski, 1984]. These objectives coincide with the class of objectives defined by monadic second-order formulas [Büchi, 1962], and they encompass linear-time temporal logic (LTL) specifications [Pnueli, 1977]. Part of their interest is due to the landmark result that finite-state machines are sufficient to implement optimal strategies in ω-regular games [Büchi and Landweber, 1969; Gurevich and Harrington, 1982], implying the decidability of related problems.

Here, we focus on the subclass of ω-regular objectives recognized by *deterministic Büchi automata* (DBA), that we call *DBA-recognizable*. The winner of a game with a DBA-recognizable objective can be decided in polynomial time in the size of the graph and the DBA by solving a Büchi game on their product [Bloem *et al.*, 2018], but this does not yield the smallest possible strategies in general.

Contributions. Our main contribution is a *characterization* (Theorem 1) of half-positionality for DBA-recognizable objectives through a conjunction of three easy-to-check conditions, presented in Section 3.

A few examples of simple DBA-recognizable objectives not encompassed by previous half-positionality criteria [Kopczyński, 2006; Bianco *et al.*, 2011] are, e.g., reaching a color twice [Bianco *et al.*, 2011, Lemma 13] and weak parity [Thomas, 2008]. We also refer to Example 3, which is half-positional but not bipositional, and whose half-positionality is straightforward using our characterization.

Various corollaries with practical and theoretical interest follow from our characterization. In particular, we obtain a painless path to show that given a DBA, the half-positionality (over both finite and infinite graphs) of the objective it recognizes is decidable in time $\mathcal{O}(k \cdot n^4)$, where k is the number of colors and n is the number of states of the DBA.

For additional technical discussions, examples, and complete proofs, we direct the interested reader to the conference version of this paper [Bouyer *et al.*, 2022a].

Other Related Works. We have discussed relevant literature on half-positionality and bipositionality. A more general quest is to understand *memory requirements* when positional strategies are not powerful enough: e.g., [Le Roux *et al.*, 2018; Bouyer *et al.*, 2022c].

Memory requirements have been precisely characterized for some classes of ω-regular objectives (not encompassing DBA-recognizable objectives), such as Muller conditions [Dziembowski *et al.*, 1997; Zielonka, 1998; Casares, 2022; Casares *et al.*, 2022] and general safety and reachability objectives [Colcombet *et al.*, 2014; Bouyer *et al.*, 2023a].

2 Preliminaries

Letter C refers to a finite non-empty set of *colors*. Given a set A, we write respectively A^*, A^+, and A^ω for the set of finite, non-empty finite, and infinite sequences of elements of A. We denote by ε the empty word.

Arenas. We consider two players $\mathcal{P}_1$ and $\mathcal{P}_2$. An *arena* is a tuple $\mathcal{A} = (V, V_1, V_2, E)$ such that V is a non-empty set of *vertices* (of any cardinality), $E \subseteq V \times C \times V$ is a set of *colored edges*, and V is the disjoint union of V_1 and V_2. Vertices in V_1 are controlled by $\mathcal{P}_1$ and vertices in V_2 are controlled by $\mathcal{P}_2$. We assume arenas to be *non-blocking*: for all $v \in V$, there exists some $(v, c, v') \in E$. For $v_0 \in V$, a *play of $\mathcal{A}$ from v_0* is an infinite sequence of edges $\pi = (v_0, c_1, v_1)(v_1, c_2, v_2)(v_2, c_3, v_3)\ldots \in E^\omega$. A *history* is a finite prefix of a play. For convenience, we define an *empty path* λ_v for every $v \in V$. If $\gamma = (v_0, c_1, v_1)\ldots(v_{n-1}, c_n, v_n)$ is a non-empty history, we define $\mathsf{last}(\gamma) = v_n$. For an empty path λ_v, we define $\mathsf{last}(\lambda_v) = v$. For $i \in \{1, 2\}$, we denote by $\mathsf{Hists}_i(\mathcal{A})$ the set of histories γ of $\mathcal{A}$ such that $\mathsf{last}(\gamma) \in V_i$.

Strategies. Let $i \in \{1, 2\}$. A *strategy of $\mathcal{P}_i$ on $\mathcal{A}$* is a function $\sigma_i\colon \mathsf{Hists}_i(\mathcal{A}) \to E$ such that for all $\gamma \in \mathsf{Hists}_i(\mathcal{A})$, the first component of $\sigma_i(\gamma)$ coincides with $\mathsf{last}(\gamma)$. Given a strategy σ_i of $\mathcal{P}_i$, we say that a play $\pi = e_1e_2\ldots$ is *consistent with* σ_i if for all finite prefixes $\gamma = e_1\ldots e_n$ of π such that $\mathsf{last}(\gamma) \in V_i$, $\sigma_i(\gamma) = e_{n+1}$. A strategy σ_i is *positional* if its outputs only depend on the current vertex and not on the whole history, i.e., if there exists a function $f\colon V_i \to E$ such that for $\gamma \in \mathsf{Hists}_i(\mathcal{A})$, $\sigma_i(\gamma) = f(\mathsf{last}(\gamma))$.

Objectives. An *objective* is a set $W \subseteq C^\omega$. An infinite word $w \in C^\omega$ is *winning* if $w \in W$, and *losing* if $w \notin W$. A *game* is a tuple $(\mathcal{A}, W)$ of an arena $\mathcal{A}$ and an objective W.

Optimality and Half-Positionality. Let $\mathcal{A} = (V, V_1, V_2, E)$ be an arena, $(\mathcal{A}, W)$ be a game, and $v \in V$. A strategy σ_1 of $\mathcal{P}_1$ is *winning from v* if all plays consistent with σ_1 induce a sequence of colors in W. A strategy of $\mathcal{P}_1$ is *optimal for $\mathcal{P}_1$ in $(\mathcal{A}, W)$* if it is winning from all the vertices from which $\mathcal{P}_1$ has a winning strategy. We stress that this notion of optimality requires a *single* strategy to be winning from *all* the winning vertices (a property sometimes called *uniformity*).

An objective W is *half-positional* if for all arenas $\mathcal{A}$, there exists an optimal strategy of $\mathcal{P}_1$ that is positional.

Deterministic Automata. A *deterministic Büchi automaton* (DBA) is a tuple $\mathcal{B} = (Q, C, q_{\mathsf{init}}, \delta, \alpha)$ where Q is a finite set of *states*, $q_{\mathsf{init}} \in Q$ is an *initial state*, $\delta\colon Q \times C \to Q$ is an *update function*, and $\alpha \subseteq Q \times C$ is a set of *Büchi transitions*.[1] We denote by δ^* the natural extension of δ to finite words.

A word $c_1c_2\ldots \in C^\omega$ is in the *language of a DBA $\mathcal{B}$* if, when read from q_{init} following δ, it sees infinitely many Büchi transitions. The language of a DBA is denoted $\mathcal{L}(\mathcal{B})$. An objective W is *DBA-recognizable* if there exists a DBA $\mathcal{B}$ such that $W = \mathcal{L}(\mathcal{B})$.

Example 1. *We give two examples of DBA in Figure 2. The language of the one on the left is the set of infinite words seeing a infinitely often; for the one on the right, it is the set of words seeing a infinitely often, or aa at some point.*

Remark 1. *The language of a DBA is always an ω-regular language. However, unlike their nondeterministic counterparts, DBA recognize only a proper subset of the ω-regular languages [Wagner, 1979].*

[1]We use transition-based acceptance conditions, and it is technically important in our approach.

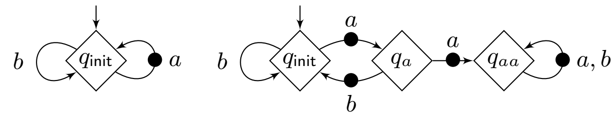


Figure 2: Two DBA using colors $C = \{a, b\}$. The Büchi transitions (transitions in α) are marked with a $\bullet$.

Right Congruence and Prefix Preorder. Let $W \subseteq C^\omega$ be an objective. For a finite word $w \in C^*$, we write $w^{-1}W = \{w' \in C^\omega \mid ww' \in W\}$ for the set of *winning continuations of w*. We define the *right congruence $\sim_W \subseteq C^* \times C^*$ of W* as $w_1 \sim_W w_2$ if $w_1^{-1}W = w_2^{-1}W$ (meaning that w_1 and w_2 have the same winning continuations). It is an equivalence relation. When the context is clear, we simply write $\sim$. For $w \in C^*$, we denote by $[w] \subseteq C^*$ its equivalence class of $\sim$.

When $\sim$ has finitely many equivalence classes, we can associate a natural deterministic "automaton structure" $\mathcal{S}_\sim = (Q_\sim, C, \widetilde{q}_{\mathsf{init}}, \delta_\sim)$ to $\sim$ such that $Q_\sim$ is the set of equivalence classes of $\sim$, $\widetilde{q}_{\mathsf{init}} = [\varepsilon]$, and $\delta_\sim([w], c) = [wc]$ [Staiger, 1983]. The transition function $\delta_\sim$ is well-defined since if $w_1 \sim w_2$, then for all $c \in C$, $w_1c \sim w_2c$. We call $\mathcal{S}_\sim$ the *prefix-classifier of W*.

We define the *prefix preorder $\preceq_W$ of W*: for $w_1, w_2 \in C^*$, we write $w_1 \preceq_W w_2$ if $w_1^{-1}W \subseteq w_2^{-1}W$. Intuitively, $w_1 \preceq_W w_2$ means that a game starting with w_2 is always preferable to a game starting with w_1 for $\mathcal{P}_1$, as there are more ways to win after w_2. When the context is clear, we simply write $\preceq$. It is a (partial) preorder. Notice that $\sim$ is equal to $\preceq \cap \succeq$. We also define the strict preorder $\prec = \preceq \setminus \sim$.

Given a DBA $\mathcal{B} = (Q, C, q_{\mathsf{init}}, \delta, \alpha)$ recognizing the objective W, observe that for $w, w' \in C^*$ such that $\delta^*(q_{\mathsf{init}}, w) = \delta^*(q_{\mathsf{init}}, w')$, we have $w \sim w'$. In this case, equivalence relation $\sim$ has at most $|Q|$ equivalence classes. For $q \in Q$, we write abusively $q^{-1}W$ for the objective recognized by the DBA $(Q, C, q, \delta, \alpha)$. Objective $q^{-1}W$ equals $w^{-1}W$ for any word $w \in C^*$ such that $\delta^*(q_{\mathsf{init}}, w) = q$. We extend the equivalence relation $\sim$ and preorder $\preceq$ to elements of Q.

3 Half-Positionality Characterization

Conditions. We first establish concepts at the core of our upcoming characterization.

Definition 1 (Progress-consistency)**.** *An objective W is* progress-consistent *if for all $w_1 \in C^*$ and $w_2 \in C^+$ such that $w_1 \prec w_1w_2$, we have $w_1(w_2)^\omega \in W$.*

Intuitively, this means that whenever a word w_2 can be used to make progress after seeing a word w_1 (in the sense of getting to a position in which more continuations are winning), then repeating this word has to be winning.

Example 2 (Non-progress-consistent objective)**.** *Let $C = \{a, b\}$. We consider the objective $W = C^*aaC^\omega$ recognized by the DBA with three states in Figure 3. This objective contains the words seeing, at some point, twice the color a in a row. This objective was discussed in the introduction and shown not to be half-positional. In particular, it is not progress-consistent: we have $\varepsilon \prec ba$, but $(ba)^\omega \notin W$.*

Example 3 (Progress-consistent objective)**.** *We go back to a slightly different example by adding two Büchi transitions:*

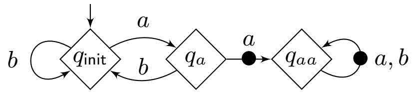


Figure 3: DBA recognizing the set of words seeing aa at some point.

see the DBA in Figure 2 (right). This DBA recognizes the objective W asking to see a infinitely often, or a twice in a row at some point. The equivalence classes of $\sim_W$ are $q_{\mathsf{init}}^{-1}W = W$, $q_a^{-1}W = aC^\omega \cup W$ and $q_{aa}^{-1}W = C^\omega$. This objective is progress-consistent: any word reaching q_{aa} is accepted when repeated infinitely often, and any word w such that $\delta^*(q_{\mathsf{init}}, w) = q_a$ necessarily contains at least one a, and thus is accepted when repeated infinitely often.

Objective W is half-positional; it will be readily shown with Theorem 1. Half-positionality of W cannot be shown using previous half-positionality [Kopczyński, 2006; Bianco et al.*, 2011] or bipositionality criteria (it is not bipositional).*

Definition 2 (Recognizability by the prefix-classifier)**.** *For an objective $W \subseteq C^\omega$ and its prefix-classifier $\mathcal{S}_\sim = (Q_\sim, C, \widetilde{q}_{\mathsf{init}}, \delta_\sim)$,* being recognized by a DBA built on top of the prefix-classifier *requires that there exists $\alpha_\sim \subseteq Q_\sim \times C$ such that W is recognized by DBA $(Q_\sim, C, \widetilde{q}_{\mathsf{init}}, \delta_\sim, \alpha_\sim)$.*

In the case of languages of *finite* words, a straightforward adaptation of the right congruence recovers the known Myhill-Nerode congruence. This equivalence relation characterizes the regular languages (a language is regular if and only if its congruence has finitely many equivalence classes), and the prefix-classifier is exactly the smallest deterministic finite automaton recognizing a language — this is the celebrated Myhill-Nerode theorem [Nerode, 1958].

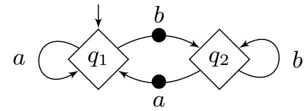
Objectives are languages of *infinite* words, for which the situation is not so clear-cut. In particular, an ω-regular objective may not always be recognized by its prefix-classifier along with a natural acceptance condition [Maler and Staiger, 1997; Angluin and Fisman, 2021]. We show an example below for the Büchi acceptance condition.

Example 4 (Not recognizable by the prefix-classifier)**.** *Let $C = \{a, b\}$. Consider the objective W recognized by the DBA in Figure 4: it asks to see both a and b infinitely often. There is only one equivalence class for $\sim$: the winning continuations of all finite words coincide (and are actually equal to W). Therefore, its prefix-classifier $\mathcal{S}_\sim$ has only one state; however, any DBA recognizing this objective needs at least two states. This objective is not half-positional, as witnessed by the arena in Figure 4 (right): $\mathcal{P}_1$ has a winning strategy from v, but it needs to take infinitely often both a and b.*

Characterization. Our characterization consists of the conjunction of three conditions. The first one requires that the prefix preorder is total, and the other two correspond to the two definitions above.

Theorem 1. *Let $W \subseteq C^\omega$ be a DBA-recognizable objective. Objective W is half-positional if and only if*

- *its prefix preorder $\preceq$ is a total preorder,*
- *it is progress-consistent, and*
- *it is recognized by a DBA built on top of its prefix-classifier.*



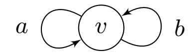


Figure 4: Left: DBA recognizing the objective of Example 4. Right: arena in which positional strategies do not suffice for $\mathcal{P}_1$ to play optimally for this objective.

This characterization is valuable to prove (and disprove) half-positionality of DBA-recognizable objectives. Examples 2 and 4 are not half-positional, and each of them falsifies exactly one of the three conditions from the statement. On the other hand, Example 3 is half-positional. We have discussed its progress-consistency, and it is also straightforward to verify that its prefix preorder is total ($[\varepsilon] \prec [a] \prec [aa]$) and that it is recognizable by a DBA built on top of its prefix-classifier (as shown with the DBA in Figure 2, right).

The first two conditions are necessary for half-positionality of *all* objectives. Being recognized by a DBA built on top of the prefix-classifier is necessary for half-positionality of *DBA-recognizable* objectives, but not for arbitrary objectives in general, including objectives recognized by other standard classes of automata over infinite words. The first condition turns out to be equivalent to earlier properties used to study bipositionality and half-positionality [Gimbert and Zielonka, 2005; Bianco *et al.*, 2011]. The third condition has been studied multiple times in the language-theoretic literature, both for itself and for minimization and learning algorithms [Staiger, 1983; Le Saëc, 1990; Maler and Staiger, 1997; Angluin and Fisman, 2021]. As an example, all deterministic *weak* automata (a restriction on DBA) satisfy it [Staiger, 1983; Angluin and Fisman, 2021].

We state two notable consequences of Theorem 1.

Lifting Result. We showed that half-positionality of DBA-recognizable objectives can be reduced to half-positionality over the restricted class of *finite, one-player arenas*. A one-player arena of $\mathcal{P}_1$ is an arena in which $\mathcal{P}_1$ controls all vertices (i.e., $V_2 = \emptyset$). Results reducing strategy complexity in two-player arenas to the easier question of strategy complexity in one-player arenas are sometimes called *one-to-two-player lifts* and appear in multiple places in the literature [Gimbert and Zielonka, 2005; Bouyer *et al.*, 2022b; Kozachinskiy, 2022; Bouyer *et al.*, 2023b].

Proposition 1 (One-to-two-player and finite-to-infinite lift)**.** *Let $W \subseteq C^\omega$ be a DBA-recognizable objective. If objective W is half-positional over the class of finite one-player arenas, then it is half-positional (over all arenas of any cardinality).*

Decidability of Half-Positionality. Given a DBA $\mathcal{B}$, deciding if $\mathcal{L}(\mathcal{B})$ is half-positional can be done in polynomial time.

Proposition 2. *Given a DBA $\mathcal{B} = (Q, C, q_{\mathsf{init}}, \delta, \alpha)$, the half-positionality of $\mathcal{L}(\mathcal{B})$ can be decided in time $\mathcal{O}(|C| \cdot |Q|^4)$.*

The algorithm checks every condition separately. It reduces each one to the inclusion of multiple pairs of languages recognized by DBA (*language containment queries*). Such a problem is standard: given two DBA $\mathcal{B}$ (with states Q) and $\mathcal{B}'$ (with states Q') on the same set C of colors, the inclusion $\mathcal{L}(\mathcal{B}) \subseteq \mathcal{L}(\mathcal{B}')$ can be decided in time $\mathcal{O}(|C| \cdot |Q| \cdot |Q'|)$.

References

[Aminof and Rubin, 2017] Benjamin Aminof and Sasha Rubin. First-cycle games. *Inf. Comput.*, 254:195–216, 2017.

[Angluin and Fisman, 2021] Dana Angluin and Dana Fisman. Regular ω-languages with an informative right congruence. *Inf. Comput.*, 278:104598, 2021.

[Bianco *et al.*, 2011] Alessandro Bianco, Marco Faella, Fabio Mogavero, and Aniello Murano. Exploring the boundary of half-positionality. *Ann. Math. Artif. Intell.*, 62(1-2):55–77, 2011.

[Bloem *et al.*, 2018] Roderick Bloem, Krishnendu Chatterjee, and Barbara Jobstmann. Graph games and reactive synthesis. In Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem, editors, *Handbook of Model Checking*, pages 921–962. Springer, 2018.

[Bouyer *et al.*, 2008] Patricia Bouyer, Ulrich Fahrenberg, Kim G. Larsen, Nicolas Markey, and Jirí Srba. Infinite runs in weighted timed automata with energy constraints. In Franck Cassez and Claude Jard, editors, *FORMATS 2008*, volume 5215 of *Lect. Notes Comput. Sci.*, pages 33–47. Springer, 2008.

[Bouyer *et al.*, 2018] Patricia Bouyer, Nicolas Markey, Mickael Randour, Kim G. Larsen, and Simon Laursen. Average-energy games. *Acta Inform.*, 55(2):91–127, 2018.

[Bouyer *et al.*, 2022a] Patricia Bouyer, Antonio Casares, Mickael Randour, and Pierre Vandenhove. Half-positional objectives recognized by deterministic Büchi automata. In Bartek Klin, Sławomir Lasota, and Anca Muscholl, editors, *CONCUR 2022*, volume 243 of *LIPIcs*, pages 20:1–20:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Bouyer *et al.*, 2022b] Patricia Bouyer, Stéphane Le Roux, Youssouf Oualhadj, Mickael Randour, and Pierre Vandenhove. Games where you can play optimally with arena-independent finite memory. *Log. Methods Comput. Sci.*, 18(1), 2022.

[Bouyer *et al.*, 2022c] Patricia Bouyer, Mickael Randour, and Pierre Vandenhove. The true colors of memory: A tour of chromatic-memory strategies in zero-sum games on graphs (invited talk). In Anuj Dawar and Venkatesan Guruswami, editors, *FSTTCS 2022*, volume 250 of *LIPIcs*, pages 3:1–3:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Bouyer *et al.*, 2023a] Patricia Bouyer, Nathanaël Fijalkow, Mickael Randour, and Pierre Vandenhove. How to play optimally for regular objectives? In *ICALP 2023 (to appear)*, volume 261 of *LIPIcs*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.

[Bouyer *et al.*, 2023b] Patricia Bouyer, Mickael Randour, and Pierre Vandenhove. Characterizing omega-regularity through finite-memory determinacy of games on infinite graphs. *TheoretiCS*, 2:1–48, 2023.

[Büchi and Landweber, 1969] J. Richard Büchi and Lawrence H. Landweber. Definability in the monadic second-order theory of successor. *J. Symb. Log.*, 34(2):166–170, 1969.

[Büchi, 1962] J. Richard Büchi. On a decision method in restricted second order arithmetic. *International Congress on Logic, Methodology and Philosophy of Science*, pages 1–11, 1962.

[Casares *et al.*, 2022] Antonio Casares, Thomas Colcombet, and Karoliina Lehtinen. On the size of good-for-games Rabin automata and its link with the memory in Muller games. In Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff, editors, *ICALP 2022*, volume 229 of *LIPIcs*, pages 117:1–117:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Casares, 2022] Antonio Casares. On the minimisation of transition-based Rabin automata and the chromatic memory requirements of Muller conditions. In Florin Manea and Alex Simpson, editors, *CSL 2022*, volume 216 of *LIPIcs*, pages 12:1–12:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Chatterjee and Doyen, 2012] Krishnendu Chatterjee and Laurent Doyen. Energy parity games. *Theor. Comput. Sci.*, 458:49–60, 2012.

[Chatterjee *et al.*, 2005] Krishnendu Chatterjee, Thomas A. Henzinger, and Marcin Jurdziński. Mean-payoff parity games. In *LICS 2005*, pages 178–187. IEEE Computer Society, 2005.

[Colcombet and Niwiński, 2006] Thomas Colcombet and Damian Niwiński. On the positional determinacy of edge-labeled games. *Theor. Comput. Sci.*, 352(1-3):190–196, 2006.

[Colcombet *et al.*, 2014] Thomas Colcombet, Nathanaël Fijalkow, and Florian Horn. Playing safe. In Venkatesh Raman and S. P. Suresh, editors, *FSTTCS 2014*, volume 29 of *LIPIcs*, pages 379–390. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2014.

[Dziembowski *et al.*, 1997] Stefan Dziembowski, Marcin Jurdziński, and Igor Walukiewicz. How much memory is needed to win infinite games? In *LICS 1997*, pages 99–110. IEEE Computer Society, 1997.

[Ehrenfeucht and Mycielski, 1979] Andrzej Ehrenfeucht and Jan Mycielski. Positional strategies for mean payoff games. *Int. J. Game Theory*, 8(2):109–113, 1979.

[Emerson and Jutla, 1991] E. Allen Emerson and Charanjit S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *FOCS 1991*, pages 368–377. IEEE Computer Society, 1991.

[Fijalkow *et al.*, 2023] Nathanaël Fijalkow, Nathalie Bertrand, Patricia Bouyer-Decitre, Romain Brenguier, Arnaud Carayol, John Fearnley, Hugo Gimbert, Florian Horn, Rasmus Ibsen-Jensen, Nicolas Markey, Benjamin Monmege, Petr Novotný, Mickael Randour, Ocan Sankur, Sylvain Schmitz, Olivier Serre, and Mateusz Skomra. *Games on Graphs*. Online, 2023.

[Gimbert and Zielonka, 2004] Hugo Gimbert and Wiesław Zielonka. When can you play positionally? In Jiří Fiala, Václav Koubek, and Jan Kratochvíl, editors, *MFCS 2004*, volume 3153 of *Lect. Notes Comput. Sci.*, pages 686–697. Springer, 2004.

[Gimbert and Zielonka, 2005] Hugo Gimbert and Wiesław Zielonka. Games where you can play optimally without any memory. In Martín Abadi and Luca de Alfaro, editors, *CONCUR 2005*, volume 3653 of *Lect. Notes Comput. Sci.*, pages 428–442. Springer, 2005.

[Gurevich and Harrington, 1982] Yuri Gurevich and Leo Harrington. Trees, automata, and games. In Harry R. Lewis, Barbara B. Simons, Walter A. Burkhard, and Lawrence H. Landweber, editors, *STOC 1982*, pages 60–65. ACM, 1982.

[Hahn *et al.*, 2022a] Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, and Dominik Wojtczak. Alternating good-for-MDPs automata. In Ahmed Bouajjani, Lukás Holík, and Zhilin Wu, editors, *ATVA 2022*, volume 13505 of *Lect. Notes Comput. Sci.*, pages 303–319. Springer, 2022.

[Hahn *et al.*, 2022b] Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, and Dominik Wojtczak. An impossibility result in automata-theoretic reinforcement learning. In Ahmed Bouajjani, Lukás Holík, and Zhilin Wu, editors, *ATVA 2022*, volume 13505 of *Lect. Notes Comput. Sci.*, pages 42–57. Springer, 2022.

[Klarlund and Kozen, 1991] Nils Klarlund and Dexter Kozen. Rabin measures and their applications to fairness and automata theory. In *LICS 1991*, pages 256–265. IEEE Computer Society, 1991.

[Klarlund, 1994] Nils Klarlund. Progress measures, immediate determinacy, and a subset construction for tree automata. *Ann. Pure Appl. Log.*, 69(2-3):243–268, 1994.

[Kopczyński, 2006] Eryk Kopczyński. Half-positional determinacy of infinite games. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP 2006*, volume 4052 of *Lect. Notes Comput. Sci.*, pages 336–347. Springer, 2006.

[Kopczyński, 2007] Eryk Kopczyński. Omega-regular half-positional winning conditions. In Jacques Duparc and Thomas A. Henzinger, editors, *CSL 2007*, volume 4646 of *Lect. Notes Comput. Sci.*, pages 41–53. Springer, 2007.

[Kopczyński, 2008] Eryk Kopczyński. *Half-positional Determinacy of Infinite Games*. PhD thesis, Warsaw University, 2008.

[Kozachinskiy, 2022] Alexander Kozachinskiy. One-to-two-player lifting for mildly growing memory. In Petra Berenbrink and Benjamin Monmege, editors, *STACS 2022*, volume 219 of *LIPIcs*, pages 43:1–43:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Le Roux *et al.*, 2018] Stéphane Le Roux, Arno Pauly, and Mickael Randour. Extending finite-memory determinacy by Boolean combination of winning conditions. In Sumit Ganguly and Paritosh K. Pandya, editors, *FSTTCS 2018*, volume 122 of *LIPIcs*, pages 38:1–38:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018.

[Le Saëc, 1990] Bertrand Le Saëc. Saturating right congruences. *RAIRO – Theoretical Informatics and Applications*, 24:545–559, 1990.

[Maler and Staiger, 1997] Oded Maler and Ludwig Staiger. On syntactic congruences for omega-languages. *Theor. Comput. Sci.*, 183(1):93–112, 1997.

[McNaughton, 1966] Robert McNaughton. Testing and generating infinite sequences by a finite automaton. *Inf. Control*, 9(5):521–530, 1966.

[Mostowski, 1984] Andrzej W. Mostowski. Regular expressions for infinite trees and a standard form of automata. In Andrzej Skowron, editor, *SCT 1984*, volume 208 of *Lect. Notes Comput. Sci.*, pages 157–168. Springer, 1984.

[Nerode, 1958] Anil Nerode. Linear automaton transformations. *Proc. Am. Math. Soc.*, 9(4):541–544, 1958.

[Ohlmann, 2023] Pierre Ohlmann. Characterizing positionality in games of infinite duration over infinite graphs. *TheoretiCS*, 2:1–51, 2023.

[Pnueli, 1977] Amir Pnueli. The temporal logic of programs. In *FOCS 1977*, pages 46–57. IEEE Computer Society, 1977.

[Shapley, 1953] Lloyd S. Shapley. Stochastic games. *Proc. Natl. Acad. Sci.*, 39(10):1095–1100, 1953.

[Staiger, 1983] Ludwig Staiger. Finite-state ω-languages. *J. Comput. Syst. Sci.*, 27(3):434–448, 1983.

[Sutton and Barto, 2018] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[Thomas, 2008] Wolfgang Thomas. Church's problem and a tour through automata theory. In Arnon Avron, Nachum Dershowitz, and Alexander Rabinovich, editors, *Pillars of Computer Science, Essays Dedicated to Boris (Boaz) Trakhtenbrot on the Occasion of His 85th Birthday*, volume 4800 of *Lect. Notes Comput. Sci.*, pages 635–655. Springer, 2008.

[Wagner, 1979] Klaus W. Wagner. On ω-regular sets. *Inf. Control*, 43(2):123–177, 1979.

[Zielonka, 1998] Wiesław Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theor. Comput. Sci.*, 200(1-2):135–183, 1998.