Infinite Separation Between General and Chromatic Memory

Alexander Kozachinskiy ✉
IMFD Chile & CENIA Chile

Abstract

In this paper, we construct a winning condition W over a finite set of colors such that, first, every finite arena has a strategy with 2 states of general memory which is optimal w.r.t. W, and second, there exists no k such that every finite arena has a strategy with k states of chromatic memory which is optimal w.r.t. W.

2012 ACM Subject Classification Theory of computation → Logic

Keywords and phrases Games on graphs, finite-memory strategies, chromatic memory

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

1 Introduction

Memory requirements for games on graphs have been studied for decades. Initially, these studies were motivated by applications to automata theory and decidability of logical theories. For example, memoryless determinacy of parity games is a key ingredient for complementation of the tree automata and leads to the decidability of the monadic second-order theory of trees [14]. Recently, games on graphs became an important tool in reactive synthesis [1]. They serve there as a model of the interaction between a reactive system and the environment. Responsibility of the game theory there is to understand, which winning conditions admit "simple" winning strategies (as we interested in implementing our system using them). The prevailing measure of complexity of strategies in the literature is memory. In this note, we study two kinds of memory – *general* (a.k.a. *chaotic*) memory and *chromatic* memory. The relationship between them was first addressed in the Ph.D. thesis of Kopczyński [11], followed by several recent works [3, 5, 6].

We focus on games that are deterministic, infinite-duration and turn-based. We call our players Protagonist and Antagonist. They play over a finite[1] directed graph called an *arena*. Its set of nodes has to be partitioned into ones controlled by Protagonist and ones controlled by Antagonist. Players move a token over the nodes of the graph along its edges. In each turn, the token is moved by the player controlling the current node.

After infinitely many turns, this process produces an infinite path in our graph. A *winning condition* is a set of infinite paths that are winning for Protagonist. In the literature, a standard way of defining winning conditions assumes that arenas are edge-colored by elements of some set of colors C. Then any subset $W \subseteq C^\omega$ is associated with a winning condition, consisting of all infinite paths whose sequence of colors belongs to W. A utility of this approach is that we do not have to define winning conditions individually for each arena.

In this paper, we seek for simple winning strategies of Protagonist, while complexity of Antagonist's strategies is mostly irrelevant for us. Such asymmetry is motivated by reactive synthesis, where Protagonist represents a system and Antagonist represents the environment. Now, the main measure of complexity of Protagonist's strategies for us is

[1] There are papers that study these games over infinite graphs, but in this note we only work with finite graphs.

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

memory. Qualitatively, we distinguish between *finite-memory* strategies and *infinite-memory* strategies. In turn, among finite-memory strategies, we prefer those that have fewer states of memory.

Finite-memory strategies are defined through so-called *memory structures*. Intuitively, a memory structure plays a role of a "hard disk" of a strategy. Formally, a *general* memory structure $\mathcal{M}$ is a deterministic finite automaton whose input alphabet is the set of edges of an arena. During the game, edges over which the token moves are fed to $\mathcal{M}$ one by one. Correspondingly, the state of $\mathcal{M}$ is updated after each move. Now, a strategy *built on top of a memory structure* $\mathcal{M}$ (or simply an $\mathcal{M}$-strategy) is a strategy whose moves at any moment depend solely on two things: first, the current node, and second, the current state of $\mathcal{M}$. A strategy is finite-memory if it can be built on top of some memory structure. More precisely, if this memory structure has k states, then strategies built on top of it are strategies *with k states of general memory*. Of course, there are strategies that cannot be built on top of any memory structure. Such strategies are infinite-memory strategies.

We also consider a special class of general memory structures called *chromatic* memory structures. A memory structure is chromatic if its transition function does not distinguish edges of the same color. In other words, chromatic memory structures only reads colors of edges that are fed into it. Alternatively, a chromatic memory structure can be viewed as a finite automaton whose input alphabet is not the set of edges, but the set of colors. Correspondingly, strategies that are built on top a chromatic memory structure with k states are called strategies with *k states of chromatic memory*.

Around a Kopczyński's question

Complexity of strategies brings us to complexity of winning conditions. For a given winning condition, we want to determine the minimal amount of memory which is sufficient to win whenever it is possible to win. More specifically, the **general memory complexity** of a winning condition W, denoted by $\mathsf{GenMem}(W)$, is the minimal $k \in \mathbb{N}$ such that in every arena there exists a Protagonist's strategy S with k states of general memory which is optimal w.r.t. W. If now such k exists, we set $\mathsf{GenMem}(W) = +\infty$. Now, "$S$ is optimal w.r.t. W" means that there exists no node v such that some Protagonist's strategy is winning from v w.r.t. W and S is not. Substituting "general memory" by "chromatic memory", we obtain a definition of the **chromatic memory complexity** of W, which is denoted by $\mathsf{ChrMem}(W)$.

For any W, we have $\mathsf{GenMem}(W) \le \mathsf{ChrMem}(W)$. This is because any chromatic memory structure is general, and hence every strategy with k states of chromatic memory is also a strategy with k states of general memory. Our paper revolves around a question from the Ph.D. thesis of Kopczyński [11].

▶ **Question 1.** *Is this true that* $\mathsf{GenMem}(W) = \mathsf{ChrMem}(W)$ *for every winning condition* W*?*

To understand Kopczyński's motivation, we first have to go back to 1969, when Büchi and Landweber [4] established that $\mathsf{ChrMem}(W)$ is finite for all ω-regular W. An obvious corollary of this is that $\mathsf{GenMem}(W)$ is also finite for all ω-regular W. Since then, there is an unfinished quest of *exactly characterizing* $\mathsf{ChrMem}(W)$ and $\mathsf{GenMem}(W)$ for ω-regular W. In particular, it is open whether $\mathsf{ChrMem}(W)$ and $\mathsf{GenMem}(W)$ are computable given an ω-regular W as an input (assuming W is given, say, in a form of a non-deterministic Büchi automaton recognizing W).

In his Ph.D. thesis, Kopczyński contributed to this question by giving an algorithm computing $\mathsf{ChrMem}(W)$ for prefix-independent ω-regular W (a winning condition is called prefix-independent if it is invariant under adding and removing finite prefixes). Prior to that, he published a weaker version of this result in [10]. He asked Question 1 to find out, whether his algorithm also computes $\mathsf{GenMem}(W)$ for prefix-independent ω-regular W. His another motivation was that the same chromatic memory structure can be used in different arenas. Indeed, transition functions of chromatic memory structures can be defined over colors so that we do not have to specify them individually for each arena.

Question 1 was recently answered by Casares in [5]. Namely, for every $n \in \mathbb{N}$ he gave a *Muller* condition W over n colors with $\mathsf{GenMem}(W) = 2$ and $\mathsf{ChrMem}(W) = n$.

▶ **Definition 1.** *A winning condition $W \subseteq C^\omega$ is* ***Muller*** *if C is finite and $\alpha \in W \iff \beta \in W$ for any two $\alpha, \beta \in C^\omega$ that have the same sets of colors occurring infinitely often in them.*

Every Muller condition is prefix-independent and ω-regular. Hence, we now know that Kopczyński's algorithm does not always compute $\mathsf{GenMem}(W)$ for prefix-independent ω-regular W. It is still open whether some other algorithm does this job.

In a follow-up work, Casares, Colcombet and Lehtinen [6] achieve a larger gap between $\mathsf{GenMem}(W)$ and $\mathsf{ChrMem}(W)$. Namely, they construct a Muller W over n colors such that $\mathsf{GenMem}(W)$ is linear in n and $\mathsf{ChrMem}(W)$ is exponential in n.

It is worth mentioning that Casares, Colcombet and Lehtinen derive these examples from their new automata-theoretic characterizations of $\mathsf{ChrMem}(W)$ and $\mathsf{GenMem}(W)$ for Muller W. First, Casares [5] showed that $\mathsf{ChrMem}(W)$ equals the minimal size of a deterministic Rabin automaton, recognizing W, for every Muller W. Second, Casares, Colcombet and Lehtinen [6] showed that $\mathsf{GenMem}(W)$ equals the minimal size of a good-for-games Rabin automaton, recognizing W, for every Muller W. The latter result complements an earlier work by Dziembowski, Jurdzinski and Walukiewicz [9], who characterized $\mathsf{GenMem}(W)$ for Muller W in terms of their Zielonka's trees [14].

These examples, however, do not answer a natural follow-up question – can the gap between $\mathsf{GenMem}(W)$ and $\mathsf{ChrMem}(W)$ be infinite? To answer it, we have to go beyond Muller and even ω-regular conditions (because $\mathsf{ChrMem}(W)$ is finite for them).

▶ **Question 2.** *Is this true that for every* ***finite*** *set of colors C and for every winning condition $W \subseteq C^\omega$ we have $\mathsf{GenMem}(W) < +\infty \implies \mathsf{ChrMem}(W) < +\infty$?*

▶ Remark 2. If we do not insist on finiteness of C, a negative answer to Question 2 follows from the example of Casares. Namely, for every n he defines a winning condition $W_n \subseteq \{1, 2, \dots n\}^\omega$, consisting of all $\alpha \in \{1, 2, \dots n\}^\omega$ such that there are exactly two numbers from 1 to n that occur infinitely often in α. He then shows that $\mathsf{GenMem}(W_n) = 2$ and $\mathsf{ChrMem}(W_n) = n$ for every n. We can now consider the union of these winning conditions $\cup_{n \geq 2} W_n$, which is a winning condition over $C = \mathbb{N}$. On one hand, $\mathsf{GenMem}(\cup_{n \geq 2} W_n) = 2$ because every arena has only finitely many natural numbers as colors, and hence $\cup_{n \geq 2} W_n$ coincides with W_n for some n there. On the other hand, we have $\mathsf{ChrMem}(\cup_{n \geq 2} W_n) \geq \mathsf{ChrMem}(W_n) = n$ for every n, which means that $\mathsf{ChrMem}(\cup_{n \geq 2} W_n) = +\infty$.

In this paper, we answer negatively to Question 2.

▶ **Theorem 3.** *There exists a finite set of colors C and a winning condition $W \subseteq C^\omega$ such that $\mathsf{GenMem}(W) = 2$ and $\mathsf{ChrMem}(W) = +\infty$.*

Topologically, our W belongs to the Σ^0_2-level of the Borel hierarchy. Next, the size of C in our example is 5, and there is a chance that it can be reduced. In turn, $\mathsf{GenMem}(W)$ is optimal because $\mathsf{GenMem}(W) = 1$ implies $\mathsf{ChrMem}(W) = 1$ (one state of general memory is equally useless as one state of chromatic memory).

We call our W the "Rope Ladder" condition. We define it in Section 3. The upper bound on $\mathsf{GenMem}(W)$ and the lower bound on $\mathsf{ChrMem}(W)$ are given in Section 4 and in Section 5, respectively. Before that, we give Preliminaries in Section 2.

Further open questions

Still, some intriguing variations of Question 2 remain open. For example, it is interesting to obtain Theorem 3 for a closed condition, i.e., for a condition given by a set of prohibited finite prefixes. In the game-theory literature, such conditions are usually called safety conditions. Our W is an infinite union of safety conditions. In [8], Colcombet, Fijalkow and Horn give a tight bound on $\mathsf{GenMem}(W)$ for safety W, but they do not address chromatic memory complexity.

▶ **Problem 1.** *Construct a finite set of colors C and a safety winning condition $W \subseteq C^\omega$ such that* $\mathsf{GenMem}(W) < \infty$ *and* $\mathsf{ChrMem}(W) = +\infty$.

It is equally interesting to obtain Theorem 3 for a prefix-independent W, as our W is not prefix-independent. One motivation is that definition of a winning condition in the Kopczyński's thesis [11] includes prefix-independence. It is unclear, whether he meant his original question for all winning conditions or only for prefix-independent ones.

▶ **Problem 2.** *Construct a finite set of colors C and a prefix-independent winning condition $W \subseteq C^\omega$ such that* $\mathsf{GenMem}(W) < \infty$ *and* $\mathsf{ChrMem}(W) = +\infty$.

There is also a variation of Question 2 related to a paper of Bouyer et al. [3]. In this paper, they introduce and study the class of arena-independent finite-memory determined winning conditions.

▶ **Definition 4.** *A winning condition W is **arena-independent finite-memory determined** if both* $\mathsf{ChrMem}(W)$ *and* $\mathsf{ChrMem}(\neg W)$ *are finite. Here $\neg W = C^\omega \setminus W$ denotes the complement to W.*

(Instead of taking the complement to W, one can swap Protagonist and Antagonist. In other words, in this definition we want both *Protagonist and Antagonist to play optimally w.r.t. W using some constant number of states of chromatic memory.)*

First, Bouyer et al. obtain an automata-theoretic characterization of arena-independent finite-memory determinacy. Second, they deduce a one-to-two-player lifting theorem from it. Namely, they show that as long as both $\mathsf{ChrMem}(W)$ and $\mathsf{ChrMem}(\neg W)$ are finite in arenas without the Antagonist's nodes, the same is true for all arenas.

A natural step forward would be to study W for which both $\mathsf{GenMem}(W)$ and $\mathsf{GenMem}(\neg W)$ are finite. Unfortunately, it is even unknown whether this is a larger class of conditions. This raises the following problem.

▶ **Problem 3.** *Construct a finite set of colors C and a winning condition $W \subseteq C^\omega$ such that* $\mathsf{GenMem}(W)$ *and* $\mathsf{GenMem}(\neg W)$ *are finite, but* $\mathsf{ChrMem}(W)$ *is infinite.*

In fact, it is not clear if our W from Theorem 3 solves this problem. We do not know whether $\mathsf{GenMem}(\neg W)$ is finite for this W.

Question 2 is also open over infinite arenas. There is a relevant result due to Bouyer, Randour and Vandenhove [2], who showed that the class of W for which $\mathsf{ChrMem}(W)$ and $\mathsf{ChrMem}(\neg W)$ are both finite in infinite arenas coincides with the class of ω-regular W. Thus, it would be sufficient to give a non-ω-regular W for which both $\mathsf{GenMem}(W)$ and $\mathsf{GenMem}(\neg W)$ are finite in infinite arenas.

Finally, let us mention a line work which studied the relationship between chromatic and general memory in the non-uniform setting. Namely, fix a single arena $\mathcal{A}$ and some winning condition W, and then consider two quantities: first, the minimal k_{gen} such that $\mathcal{A}$ has an optimal strategy with k_{gen} states of general memory, and second, the minimal k_{chr} such that $\mathcal{A}$ has an optimal strategy with k_{chr} states of chromatic memory. In [13], Le Roux showed that if k_{gen} is finite, then k_{chr} is also finite. There is no contradiction with Theorem 3 because k_{chr} depends not only on k_{gen}, but also on $\mathcal{A}$. A tight bound on k_{chr} in terms of k_{gen} and the number of nodes of $\mathcal{A}$ was obtained in [12].

2 Preliminaries

Notation. For a set A, we let A^* and A^ω stand for the set of all finite and the set of all infinite sequences of elements of A, respectively. For $x \in A^*$, we let $|x|$ denote the length of x. We also set $|x| = +\infty$ for $x \in A^\omega$. We let $\circ$ denote the function composition. The set of positive integral numbers is denoted by $\mathbb{Z}^+$.

2.1 Arenas

▶ **Definition 5.** *Let C be a non-empty set. A tuple $\mathcal{A} = \langle V_P, V_A, E\rangle$ is called an* ***arena over the set of colors*** *C if the following conditions hold:*

- *V_P, V_A, E are finite sets such that $V_P \cap V_A = \varnothing$, $V_P \cup V_A \neq \varnothing$ and $E \subseteq (V_P \cup V_A) \times C \times (V_P \cup V_A)$;*
- *for every $s \in V_P \cup V_A$ there exists $c \in C$ and $t \in V_P \cup V_A$ such that $(s, c, t) \in E$.*

Elements of the set $V = V_P \cup V_A$ will be called nodes of $\mathcal{A}$. Elements of V_P will be called nodes controlled by Protagonist (or simply Protagonist's nodes). Similarly, elements of V_A will be called nodes controlled by Antagonist (or simply Antagonist's nodes). Elements of E will be called edges of $\mathcal{A}$. For an edge $e = (s, c, t) \in E$ we define $\mathsf{source}(e) = s$, $\mathsf{col}(e) = c$ and $\mathsf{target}(e) = t$. We imagine $e \in E$ as an arrow which is drawn from the node $\mathsf{source}(e)$ to the node $\mathsf{target}(e)$ and which is colored into $\mathsf{col}(e)$. Note that the second condition in the definition of an arena means that every node has at least one out-going edge.

We extend the function col to a function $\mathsf{col}\colon E^* \cup E^\omega \to C^* \cup C^\omega$ by setting:

$$\mathsf{col}(e_1 e_2 e_3 \ldots) = \mathsf{col}(e_1)\mathsf{col}(e_2)\mathsf{col}(e_3)\ldots, \qquad e_1, e_2, e_3, \ldots \in E.$$

A non-empty sequence $p = e_1 e_2 e_3 \ldots \in E^* \cup E^\omega$ is called a path if for any $1 \le i < |p|$ we have $\mathsf{target}(e_i) = \mathsf{source}(e_{i+1})$. We set $\mathsf{source}(p) = \mathsf{source}(e_1)$ and, if p is finite, $\mathsf{target}(p) = \mathsf{target}(e_{|p|})$. For technical convenience, every node $v \in V$ is assigned a 0-length path λ_v, for which we set $\mathsf{source}(\lambda_v) = \mathsf{target}(\lambda_v) = v$ and $\mathsf{col}(\lambda_v) =$ empty string.

Paths are sequences of edges, so we will say that some paths are prefixes of the others. However, we have to define this for 0-length paths. Namely, we say that λ_v is a prefix of a path p if and only if $\mathsf{source}(p) = v$.

2.2 Strategies

Let $\mathcal{A} = \langle V_P, V_A, E\rangle$ be an arena over the set of colors C. A Protagonist's strategy in $\mathcal{A}$ is any function

$$S\colon \{p \mid p \text{ is a finite path in } \mathcal{A} \text{ with } \mathsf{target}(p) \in V_P\} \to E,$$

such that for every p from the domain of S we have $\mathsf{source}(S(p)) = \mathsf{target}(p)$. In this paper, we do not mention Antagonist's strategies, but, of course, they can be defined similarly.

The set of finite paths in $\mathcal{A}$ is the set of positions of the game. Possible starting positions are 0-length paths $\lambda_s, s \in V$. When the starting position[2] is λ_s, we say that the game starts at s. Now, consider any finite path p. Protagonist is the one to move after p if and only if $t = \mathsf{target}(p)$ is a Protagonist's node. In this situation, Protagonist must choose some edge starting at t. A Protagonist's strategy fixes this choice for every p with $\mathsf{target}(p) \in V_P$. We then append this edge to p and get the next position in the game. Antagonist acts the same for those p such that $\mathsf{target}(p)$ is an Antagonist's node.

Let us define paths that are consistent with a Protagonist's strategy S. First, any 0-length path λ_v is consistent with S. Now, a non-empty path $p = e_1 e_2 e_3 \dots$ (which may be finite or infinite) is consistent with S if the following holds:

- if $\mathsf{source}(p) \in V_P$, then $e_1 = S(\lambda_{\mathsf{source}(p)})$;
- for every $1 \le i < |p|$, if $\mathsf{target}(e_i) \in V_P$, then $e_{i+1} = S(e_1 e_2 \dots e_i)$.

For brevity, paths that are consistent with S will also be called *plays with S*. For a node v, we let $\mathsf{FinitePlays}(S, v)$ and $\mathsf{InfinitePlays}(S, v)$ denote the set of finite plays with S that start at v and the set of infinite plays with S that start at v, respectively. For $U \subseteq V$, we define $\mathsf{FinitePlays}(S, U) = \bigcup_{v \in U} \mathsf{FinitePlays}(S, v)$ and $\mathsf{InfinitePlays}(S, U) = \bigcup_{v \in U} \mathsf{InfinitePlays}(S, v)$.

2.3 Memory structures

Let $\mathcal{A} = \langle V_P, V_A, E\rangle$ be an arena over the set of colors C. A memory structure in $\mathcal{A}$ is a tuple $\mathcal{M} = \langle M, m_{init}, \delta\rangle$, where M is a finite set, $m_{init} \in M$ and $\delta \colon M \times E \to M$. Elements of M are called states of $\mathcal{M}$, m_{init} is called the initial state of $\mathcal{M}$ and δ is called the transition function of $\mathcal{M}$. Given $m \in M$, we inductively define the function $\delta(m, \cdot)$ over arbitrary finite sequences of edges:

$$\delta(m, \text{empty sequence}) = m,$$
$$\delta(m, se) = \delta(\delta(m, s), e), \qquad s \in E^*, e \in E.$$

In other words, $\delta(m, s)$ is the state of $\mathcal{M}$ after we fed s to it, provided that before that $\mathcal{M}$ was in m.

A memory structure $\mathcal{M} = \langle M, m_{init}, \delta\rangle$ is called chromatic if $\delta(m, e_1) = \delta(m, e_2)$ for every $m \in M$ and for every $e_1, e_2 \in E$ with $\mathsf{col}(e_1) = \mathsf{col}(e_2)$. In this case, there exists $\sigma \colon M \times C \to M$ such that $\delta(m, e) = \sigma(m, \mathsf{col}(e))$. In other words, we can view $\mathcal{M}$ as a deterministic finite automaton over C, with σ being its transition function.

A strategy S is built on top of a memory structure $\mathcal{M}$ if we have $S(p_1) = S(p_2)$ for any two paths p_1, p_2 with $\mathsf{target}(p_1) = \mathsf{target}(p_2)$ and $\delta(m_{init}, p_1) = \delta(m_{init}, p_2)$. In this case, we sometimes simply say that S is an $\mathcal{M}$-strategy. To define an $\mathcal{M}$-strategy S, it is sufficient to give its *next-move function* $n_S \colon V_P \times M \to E$. For $v \in V_P$ and $m \in M$, the value of $n_S(v, m)$ determines what S does for paths that end at v and bring $\mathcal{M}$ to m from m_{init}.

A strategy S built on top of a memory structure $\mathcal{M}$ with k states is called a strategy with k states of general memory. If $\mathcal{M}$ is chromatic, then S is a strategy with k states of chromatic memory.

For brevity, if S is an $\mathcal{M}$-strategy and p is a finite path, we say that $\delta(m_{init}, p)$ is the state of S after p.

[2] We do not have to redefine S for every starting position. The same S can be played from any of them.

2.4 Winning conditions and their memory complexity

A winning condition is any set $W \subseteq C^\omega$. We say that a Protagonist's strategy S is winning from a node u w.r.t. to W if the image of $\mathsf{InfinitePlays}(S, u)$ under col is a subset of W. In other words, any infinite play from u against S must give a sequence of colors belonging to W. Now, a Protagonist's strategy S is called optimal w.r.t. W if there exists no node u such that some Protagonist's strategy is winning from u w.r.t. W and S is not.

We let $\mathsf{GenMem}(W)$ be the minimal $k \in \mathbb{Z}^+$ such that in every arena $\mathcal{A}$ over C there exists a Protagonist's strategy with k states of general memory which is optimal w.r.t. W. If no such k exists, we set $\mathsf{GenMem}(W) = +\infty$. Likewise, we let $\mathsf{ChrMem}(W)$ be the minimal $k \in \mathbb{Z}^+$ such that in every arena $\mathcal{A}$ over C there exists a Protagonist's strategy with k states of general memory which is optimal w.r.t. W. Again, if no such k exists, we set $\mathsf{ChrMem}(W) = +\infty$.

3 The "Rope Ladder" Condition

Consider a partially ordered set $\Omega = (\mathbb{N} \times \{0, 1\}, \preceq)$, where $\preceq$ is defined by

$$\forall (n, a), (m, b) \in \mathbb{N} \times \{0, 1\} \qquad (n, a) \preceq (m, b) \iff (n, a) = (m, b) \text{ or } n < m.$$

$$\begin{tikzcd}
\vdots & \vdots \\
(2,0) \arrow[d] \arrow[dr] & (2,1) \arrow[d] \arrow[dl] \\
(1,0) \arrow[d] \arrow[dr] & (1,1) \arrow[d] \arrow[dl] \\
(0,0) & (0,1)
\end{tikzcd}$$

Above is its informal depiction, with arrows representing $\preceq$ (they are directed from bigger elements to smaller elements):

We will use an abbreviation $\overline{\mathbf{0}} = (0, 0)$. Next, we let $\mathbb{M}$ be the set of all functions $f : \Omega \to \Omega$ that are monotone w.r.t. $\preceq$. Being monotone w.r.t. $\preceq$ means that $x \preceq y \implies f(x) \preceq f(y)$ for all $x, y \in \Omega$.

▶ **Definition 6.** *The **Rope Ladder** condition is a set* $\mathsf{RL} \subseteq \mathbb{M}^\omega$*, consisting of all infinite sequences* $(f_1, f_2, f_3, \ldots) \in \mathbb{M}^\omega$ *for which there exists* $(N, b) \in \Omega$ *such that* $f_n \circ \ldots \circ f_2 \circ f_1(\overline{\mathbf{0}}) \preceq (N, b)$ *for all* $n \geq 1$.

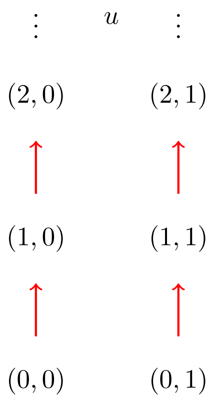
We will use the following informal terminology with regard to RL. Imagine that there is an ant which can move over the elements of Ω. Initially, it sits at $\overline{\mathbf{0}}$. Next, take any sequence $(f_1, f_2, f_3, \ldots) \in \mathbb{M}^\omega$. We start moving the ant by applying functions from the sequence to the position of the ant. Namely, we first move the ant from $\overline{\mathbf{0}}$ to $f_1(\overline{\mathbf{0}})$, then from $f_1(\overline{\mathbf{0}})$ to $f_2 \circ f_1(\overline{\mathbf{0}})$, and so on. Now, $(f_1, f_2, f_3, \ldots) \in \mathsf{RL}$ if and only if there exists a "layer" in Ω which is never exceeded by the ant.

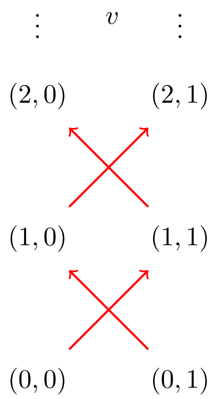
▶ Remark 7. RL is defined over infinitely many colors, but for our lower bound on its chromatic memory complexity we will consider its restriction to some finite subset of $\mathbb{M}$.

To illustrate these definitions, we establish the following fact. It can also be considered as a warm-up for our lower bound.

▶ **Fact 1.** $\mathsf{ChrMem}(\mathsf{RL}) > 1$.

Proof. First, consider $u, v\colon \Omega \to \Omega$, depicted below:



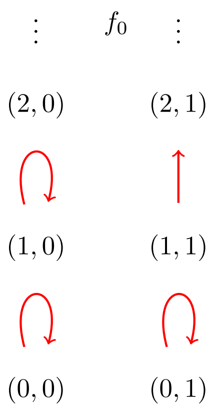


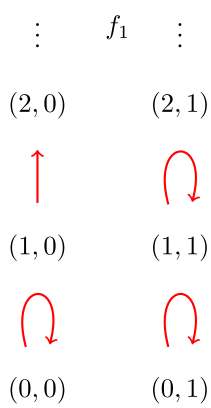
These functions are defined by arrows that direct each element of Ω to the value of the function on this element. Formally, $u((n,a)) = (n+1,a)$ and $v((n,a)) = (n+1,1-a)$ for every $(n,a) \in \Omega$. It holds that $u, v \in \mathbb{M}$ because they both always increase the first coordinate by 1.

We also consider the following two functions $f_0, f_1\colon \Omega \to \Omega$:

$$f_b((n,a)) = \begin{cases} (n,a) & (n,a) = (0,0), (0,1) \text{ or } (1,b), \\ (n+1,a) & \text{otherwise}, \end{cases} \qquad b \in \{0,1\} \tag{1}$$

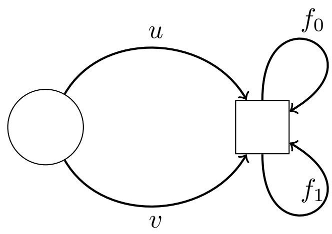
For the reader's convenience, we depict them as well.





Both of these functions have 3 fixed points. At remaining points, they act by increasing the first coordinate by 1. Now, note that the set of fixed points is downwards-closed w.r.t. $\preceq$ for both of them. Hence, $f_0, f_1 \in \mathbb{M}$.

Consider the following arena.



The circle is controlled by Antagonist and the square is controlled by Protagonist. Assume that the game start in the circle. We first show that Protagonist has a wining strategy w.r.t. RL. Then we show that Protagonist does not have a positional strategy which is winning w.r.t. RL. This implies that $\mathsf{ChrMem}(\mathsf{RL}) > 1$.

Let us start with the first claim. After the first move of Antagonist, the ant moves either to $u(\overline{\mathbf{0}}) = (1, 0)$ or to $v(\overline{\mathbf{0}}) = (1, 1)$. In the first case, Protagonist wins by forever using the f_0-edge (the ant will always stay at $(1, 0)$). In the second case, Protagonist wins by always using the f_1-edge (the ant will always stay at $(1, 1)$).

Now we show that every positional strategy of Protagonist is not winning w.r.t. RL. In fact, there are just 2 Protagonist's positional strategies – one which always uses the f_0-edge and the other which always uses the f_1-edge. The first one looses if Antagonist goes by the v-edge. Then the ant moves to $v(\overline{\mathbf{0}}) = (1, 1)$. If we start applying f_0 to the ant's position, the first coordinate of the ant will get arbitrarily large. Similarly, the second Protagonist's positional strategy looses if Antagonist goes by the u-edge. ◀

4 Upper Bound on the General Memory

In this section, we establish

▶ **Proposition 8.** $\mathsf{GenMem}(\mathsf{RL}) = 2$.

By Fact 1, we only have to show an upper bound $\mathsf{GenMem}(\mathsf{RL}) \leq 2$. For that, for every arena $\mathcal{A}$ over $\mathbb{M}$ and for every Protagonist's strategy S_1 in $\mathcal{A}$ we construct a Protagonist's strategy S_2 with 2 states of general memory for which the following holds: for every node v of $\mathcal{A}$, if S_1 is winning w.r.t. W from v, then so is S_2.

We will use the following notation. Take any finite path $p = e_1 \ldots e_m$ in $\mathcal{A}$. Define $\mathsf{ant}(p) = \mathsf{col}(e_m) \circ \ldots \circ \mathsf{col}(e_2) \circ \mathsf{col}(e_1)(\overline{\mathbf{0}})$. In other words, $\mathsf{ant}(p)$ is the position of the ant after the path p. In case when p is a 0-length path, we set $\mathsf{ant}(p) = \overline{\mathbf{0}}$. We also write $\mathsf{layer}(p)$ for the first coordinate of $\mathsf{ant}(p)$.

Let W be the set of nodes of $\mathcal{A}$ from which S_1 is winning w.r.t. RL. By definition of RL, for every $P \in \mathsf{InfinitePlays}(S_1, W)$ there exists $N \in \mathbb{N}$ such that $\mathsf{layer}(p) \leq N$ for every finite prefix p of P. First step of our argument is to change the quantifiers here. That is, we obtain a strategy S_1' for which there exists some $N \in \mathbb{N}$ such that $\mathsf{layer}(p) \leq N$ for every $p \in \mathsf{FinitePlays}(S_1', W)$.

We use an argument, similar to one which was used in [7] to show finite-memory determinacy of multi-dimensional energy games. We call $p \in \mathsf{FinitePlays}(S_1, W)$ *regular* if there exist two prefixes q_1 and q_2 of p such that, first, q_1 is shorter than q_2, second, $\mathsf{target}(q_1) = \mathsf{target}(q_2)$, and third, $\mathsf{ant}(q_1) = \mathsf{ant}(q_2)$. In other words, q_1 and q_2 must lead to the same node in $\mathcal{A}$ and to the same position of the ant in Ω. We stress that q_2 might coincide with p, but q_1 must be a proper prefix of p. If $p \in \mathsf{FinitePlays}(S_1, W)$ is not regular, then we call it *irregular*.

First, we show that there are only finitely many irregular plays in $\mathsf{FinitePlays}(S_1, W)$. Note that any prefix of an irregular play is irregular. Thus, irregular plays form a collection of trees with finite branching (for each $u \in W$ there is a tree of irregular plays that start at u). Assume for contradiction that there are infinitely many irregular plays. Then, by Kőnig's lemma, there exists an infinite branch in one of our trees. It gives some $P \in \mathsf{InfinitePlays}(S_1, W)$ whose finite prefixes are all irregular. However, P must be winning for Protagonist w.r.t. RL. In other words, there exists $N \in \mathbb{N}$ such that $\mathsf{layer}(p) \leq N$ for every finite prefix of P. So, if p ranges over finite prefixes of P, then $\mathsf{ant}(p)$ takes only finitely many values. Hence, there exist a node v of $\mathcal{A}$ and some $(n, b) \in \Omega$ such that $v = \mathsf{target}(p)$ and $(n, b) = \mathsf{ant}(p)$ for infinitely many prefixes of P. Consider any two such prefixes. A longer one is regular because the shorter one is its prefix and leads to the same node in $\mathcal{A}$ and to the same position of the ant. This is a contradiction.

We now define S_1'. It will maintain the following invariant for plays that start at W: if p_{cur} is the current play, then there exists an irregular $p \in \mathsf{FinitePlays}(S_1, W)$ such that $\mathsf{target}(p_{cur}) = \mathsf{target}(p)$ and $\mathsf{ant}(p_{cur}) = \mathsf{ant}(p)$. Since there are only finitely many irregular plays, this invariant implies that $\mathsf{ant}(p_{cur})$ takes only finitely many values over $p_{cur} \in \mathsf{FinitePlays}(S_1', W)$, as required from S_1'.

To maintain the invariant, S_1' plays as follows. In the beginning, $p_{cur} = \lambda_w$ for some $w \in W$. Hence, we can set $p = \lambda_w$ also. Indeed, $\lambda_w \in \mathsf{FinitePlays}(S_1, W)$ and it is irregular as it has no proper prefixes. Let us now show how to maintain the invariant. Consider any play p_{cur} with S_1' for which there exists an irregular $p \in \mathsf{FinitePlays}(S_1, W)$ such that $\mathsf{target}(p_{cur}) = \mathsf{target}(p)$ and $\mathsf{ant}(p_{cur}) = \mathsf{ant}(p)$. In this position, if its Protagonist's turn to move, S_1' makes the same move as S_1 from p. As a result, some edge e is played. Observe that $pe \in \mathsf{FinitePlays}(S_1, W)$. In turn, our new current play with S_1' is $p_{cur}e$. We have that $\mathsf{target}(p_{cur}e) = \mathsf{target}(pe) = \mathsf{target}(e)$ and $\mathsf{ant}(p_{cur}e) = \mathsf{col}(e)\big(\mathsf{ant}(p_{cur})\big) = \mathsf{col}(e)\big(\mathsf{ant}(p)\big) = \mathsf{ant}(pe)$. So, if pe is irregular, then the invariant is maintained. Now, assume that pe is regular. Then there are two prefixes q_1 and q_2 of pe such that, first, q_1 is shorter than q_2, second, $\mathsf{target}(q_1) = \mathsf{target}(q_2)$, and third, $\mathsf{ant}(q_1) = \mathsf{ant}(q_2)$. Since p is irregular, q_2 cannot be a prefix of p. Hence, $q_2 = pe$. By the same reason, q_1 is irregular. Thus, invariant is maintained if we set the new value of p be q_1. Indeed, $\mathsf{target}(p_{cur}e) = \mathsf{target}(pe) = \mathsf{target}(q_2) = \mathsf{target}(q_1)$ and $\mathsf{ant}(p_{cur}e) = \mathsf{ant}(pe) = \mathsf{ant}(q_2) = \mathsf{ant}(q_1)$.

We now turn S_1' into a strategy S_2 with 2 states of general memory which is winning w.r.t. RL from every node of W.

Preliminary definitions. Let X be the set of nodes reachable from W by plays with S_1'. Next, for $v \in X$, define $\Omega_v \subseteq \Omega$ as the set of all $(n, b) \in \Omega$ such that $(n, b) = \mathsf{ant}(p)$ for some $p \in \mathsf{FinitePlays}(S_1', W)$ with $v = \mathsf{target}(p)$. In other words, Ω_v is the set of all possible positions of the ant that can arise at v if we play according to S_1' from a node of W.

Now, take any $v \in X$. The set Ω_v is non-empty and, by our requirements on S_1', finite. Hence, it has 1 or 2 maximal elements w.r.t. $\preceq$. We will denote them by M_0^v and M_1^v. If Ω_v has just a single maximal element, then $M_0^v = M_1^v$. If Ω_v has two different maxima, then M_0^v is the one having 0 as the second coordinate. Finally, for every $v \in X$ and for every $b \in \{0, 1\}$ fix some $p_b^v \in \mathsf{FinitePlays}(S_1', W)$ such that $\mathsf{target}(p_b^v) = v$ and $\mathsf{ant}(p_b^v) = M_b^v$.

Description of S_2. Two states of S_2 will simply be denoted by 0 and 1. The initial state of S_2 is 0. The next-move function of S_2 is defined as follows. Assume that the state of S_2 is $I \in \{0, 1\}$ and it has to make a move from a node v. If $v \notin X$, it makes an arbitrary move (this case does not matter for the argument below). Now, assume that $v \in X$. Then S_2 make the same move as S_1' after p_I^v.

We now describe the memory structure of S_2. Assume that it receives an edge e when its state is $I \in \{0,1\}$. The new state $J \in \{0,1\}$ is computed as follows. Denote $u = \mathsf{source}(e)$ and $v = \mathsf{target}(e)$. If $u \notin X$ or $v \notin X$, then $J = 0$ (again, this case is irrelevant for the rest of the argument). Assume now that $u, v \in X$. If $\mathsf{col}(e)\big(M_I^u\big) \in \Omega_v$, then we find some $b \in \{0,1\}$ such that $\mathsf{col}(e)\big(M_I^u\big) \preceq M_b^v$ and set $J = b$. Otherwise, we set $J = 0$.

Showing that S_2 is winning from W. First, we observe that $\mathsf{target}(p) \in X$ for every $p \in \mathsf{FinitePlays}(S_2, W)$ (in other words, S_2 cannot leave X if we start somewhere in W). Indeed, assume for contradiction that some play with S_2 leaves X from some node $v \in X$. Let I be the state of S_2 at the moment just before leaving X. If it is a Protagonist's turn to move, then it moves as S_1' after p_I^v. Recall that $p_I^v \in \mathsf{FinitePlays}(S_1', W)$. Thus, we obtain a continuation of p_I^v which is consistent with S_1' and leads outside X. This contradicts the definition of X. Now, if it is an Antagonist's turn to move from v, then any continuation of p_I^v by one edge is consistent with S_1', so we obtain the same contradiction.

Next, we show that for any play $p \in \mathsf{FinitePlays}(S_2, W)$ we have $\mathsf{ant}(p) \preceq M_I^{\mathsf{target}(p)}$, where I is the state of S_2 after p. This statement implies that S_2 is winning w.r.t. RL from every node of W. Note that $M_I^{\mathsf{target}(p)}$ is well-defined thanks to the previous paragraph.

We prove this statement by induction on the length of p. Let us start with the induction base. Assume that $|p| = 0$ (then $p = \lambda_w$ for some $w \in W$). The state of S_2 after p is the initial state, that is, 0. Thus, we have to show that $\mathsf{ant}(p) \preceq M_0^{\mathsf{target}(p)}$. Note that p has length 0 and hence is consistent with any strategy. In particular, $p \in \mathsf{FinitePlays}(S_1', W)$. Hence, $\mathsf{ant}(p) \in \Omega_{\mathsf{target}(p)}$. If $\Omega_{\mathsf{target}(p)}$ has just a single maximum, then $\mathsf{ant}(p)$ does not exceed this maximum, as required. Now, if $M_0^{\mathsf{target}(p)} \neq M_1^{\mathsf{target}(p)}$, then the second coordinate of $M_0^{\mathsf{target}(p)}$ is 0, so we have $\mathsf{ant}(p) \preceq M_0^{\mathsf{target}(p)}$ just because $\mathsf{ant}(p) = \overline{\mathbf{0}}$.

Next, we establish the induction step. Consider any $p \in \mathsf{FinitePlays}(S_2, W)$ of positive length and assume that for all paths from $\mathsf{FinitePlays}(S_2, W)$ of smaller length the statement is already proved. We prove our statement for p. Let e be the last edge of p. Correspondingly, let q be the part of p preceding e. Denote $u = \mathsf{target}(q) = \mathsf{source}(e)$ and $v = \mathsf{target}(p) = \mathsf{target}(e)$

Any prefix of p is also in $\mathsf{FinitePlays}(S_2, W)$, so $q \in \mathsf{FinitePlays}(S_2, W)$. Therefore, our statement holds for q. Namely, if I is the state of S_2 after q, then $\mathsf{ant}(q) \preceq M_I^u$.

Let J be the state of S_2 after p. Our goal is to show that $\mathsf{ant}(p) \preceq M_J^v$. Note that $\mathsf{ant}(p) = \mathsf{col}(e)\big(\mathsf{ant}(q)\big)$ by definition of ant. Since $\mathsf{col}(e) \in \mathbb{M}$ is monotone and $\mathsf{ant}(q) \preceq M_I^u$, we have that $\mathsf{ant}(p) = \mathsf{col}(e)\big(\mathsf{ant}(q)\big) \preceq \mathsf{col}(e)\big(M_I^u\big)$. It remains to show that $\mathsf{col}(e)\big(M_I^u\big) \preceq M_J^v$. Note that J is the state into which S_2 transits from the state I after receiving e. By definition of the memory structure of S_2, it is sufficient to show that $\mathsf{col}(e)\big(M_I^u\big) \in \Omega_v$.

By definition of p_I^u, we have that $M_I^u = \mathsf{ant}(p_I^u)$. Hence, $\mathsf{col}(e)\big(M_I^u\big) = \mathsf{ant}(p_I^u e)$. The path $p_I^u e$ starts at some node of W and ends in $\mathsf{target}(e) = v$. Thus, to establish $\mathsf{ant}(p_I^u e) \in \Omega_v$, it remains to show consistency of $p_I^u e$ with S_1'. We have $p_I^u \in \mathsf{FinitePlays}(S_1', W)$ by definition of p_I^u. In turn, if Protagonist is the one to move from $u = \mathsf{target}(p_I^u)$, then $e = S_1'(p_I^u)$. Indeed, e is the edge played by S_2 from u when its state is I. Hence, $e = S_1'(p_I^u)$, by the definition of the next-move function of S_2.

5 Lower Bound on the Chromatic Memory

In this section, we establish the following proposition.

▶ **Proposition 9.** *There exists a finite set $C \subseteq \mathbb{M}$ such that* $\mathsf{ChrMem}(\mathsf{RL} \cap C^\omega) = +\infty$.

We start by describing C. First, we put there f_0, f_1 that are defined in (1). Next, put there a function $h\colon \Omega \to \Omega$, defined by

$$h((n,a)) = \begin{cases} (n-1,a) & n > 1 \\ (0,0) & n = 0,1. \end{cases}$$

For the reader's convenience, it is depicted below:

$$\begin{tikzcd}
\vdots & h & \vdots \\
(2,0) \arrow[d] & & (2,1) \arrow[d] \\
(1,0) \arrow[d] & & (1,1) \arrow[dll] \\
(0,0) \arrow[loop left] & & (0,1) \arrow[ll]
\end{tikzcd}$$

Let us establish that $h \in \mathbb{M}$. Take any $(n,a),(m,b) \in \Omega$ such that $(n,a) \preceq (m,b)$. We show that $h((n,a)) \preceq h((m,b))$. If $(n,a) = (m,b)$, then $h((n,a)) = h((m,b))$. Now, if $(n,a) \neq (m,b)$, then $n < m$. The first coordinates of $h((n,a))$ and $h((m,b))$ are $\max\{0, n-1\}$ and $\max\{0, m-1\}$, respectively. If $m > 1$, then $\max\{0, m-1\} = m-1 > \max\{0, n-1\}$, which implies that $h((n,a)) \preceq h((m,b))$. Now, if $m \leq 1$, then $n \leq 1$ also, which means that $h((n,a)) = h((m,b)) = (0,0)$.

We will also put into C two more functions $p^0, p^1\colon \Omega \to \Omega$, but to define them, we need some axillary work.

▶ **Definition 10.** *A function $f\colon \Omega \to \Omega$ is called* ***incremental*** *if for every $(n,b) \in \Omega$ there exists $c \in \{0,1\}$ such that $f((n,b)) = (n+1,c)$.*

Let us first observe that every incremental $f\colon \Omega \to \Omega$ belongs to $\mathbb{M}$. Indeed, take any $(n,a),(m,b) \in \Omega$ such that $(n,a) \preceq (m,b)$. We have to show that $f((n,a)) \preceq f((m,b))$. for every incremental f. If $(n,a) = (m,b)$, we have $f((n,a)) = f((m,b))$. Otherwise, $n < m$. Then $f((n,a)) = (n+1,c)$ and $f((m,b)) = (m+1,d)$ for some $c,d \in \{0,1\}$. Since $n+1 < m+1$, we have $f((n,a)) \preceq f((m,b))$.

We say that two binary words $x,y \in \{0,1\}^*$ are *Q-indistinguishable* if there exists no deterministic finite automaton over $\{0,1\}$ with at most Q states which comes to different states on x and on y.

▶ **Lemma 11.** *There exist two infinite sequences of bits $\{I_n^0\}_{n=0}^\infty \in \{0,1\}^\omega$ and $\{I_n^1\}_{n=0}^\infty \in \{0,1\}^\omega$ such that for every $Q \in \mathbb{N}$ there exist $t \in \mathbb{N}$ and two Q-indistinguishable binary words $x = x_0 \ldots x_{t-1}$ and $y = y_0 \ldots y_{t-1}$ of length t such that:*

$$I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}} \neq I_0^{y_0} \oplus \ldots \oplus I_{t-1}^{y_{t-1}}$$

($\oplus$ denotes XOR).

Proof. Let A_Q be the number of deterministic finite automata over $\{0,1\}$ with at most Q states. For $Q \in \mathbb{N}$, let l_Q be any number such that $2^{l_Q} > Q^{A_Q}$.

We split natural numbers in consecutive blocks $B_1, B_2, B_3, \ldots$, where

$$B_Q = \{l_1 + \ldots + l_{Q-1}, \ldots, l_1 + \ldots + l_{Q-1} + l_Q - 1\}$$

(so that $|B_Q| = l_Q$). We first define I_n^0, I_n^1 for $n \in B_1$, then for $n \in B_2$, and so on. The requirement of the lemma for Q will be guaranteed after we define our sequences in the first Q blocks.

More specifically, assume that our sequences are already defined in the first $Q-1$ blocks. We have to define them in B_Q in some way that satisfies the requirement of the lemma for Q. Since $2^{l_Q} > Q^{A_Q}$, there exist two different Q-indistinguishable binary words $a, b \in \{0,1\}^{t_Q}$. Indeed, to every binary word w we can assign a tuple, where for all deterministic finite automaton $\mathcal{A}$ with at most Q states we have a coordinate, indicating the state of $\mathcal{A}$ after reading w. The number of such tuples is bounded by Q^{A_Q}. Hence, in $\{0,1\}^{t_Q}$ there are two different binary words a, b with the same tuple assigned to them. This means that a and b are Q-indistinguishable.

Let $t = l_1 + \ldots + l_Q$ and

$$x = \underbrace{00\ldots0}_{l_1+\ldots+l_{Q-1}} a \in \{0,1\}^t, \qquad y = \underbrace{00\ldots0}_{l_1+\ldots+l_{Q-1}} b \in \{0,1\}^t.$$

Note that x and y are obtained from a and b by attaching the same prefix. Hence, x and y are also Q-indistinguishable. We claim that we can define I_n^0, I_n^1 for $n \in B_Q$ in such a way that:

$$I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}} \neq I_0^{y_0} \oplus \ldots \oplus I_{t-1}^{y_{t-1}}. \tag{2}$$

Since a and b are different, we have that x and y are also different. But both x and y start with $l_1 + \ldots + l_{Q-1}$ zeros. Hence, all the indices where they differ belong to B_Q. Take any $m \in B_Q$ such that $x_m \neq y_m$. Define $I_n^0 = I_n^1 = 0$ for all $n \in B_Q \setminus \{m\}$. It remains to define I_m^0 and I_m^1 in such a way that (2) holds. Note that all the summands except $I_m^{x_m}$ and $I_m^{y_m}$ are already defined. Since $x_m \neq y_m$, one of these summands is I_m^0 and the other is I_m^1. One of them is in the left-hand side, and the other one is in the right-hand side. Hence, we can define them in such a way that the inequality is true. ◀

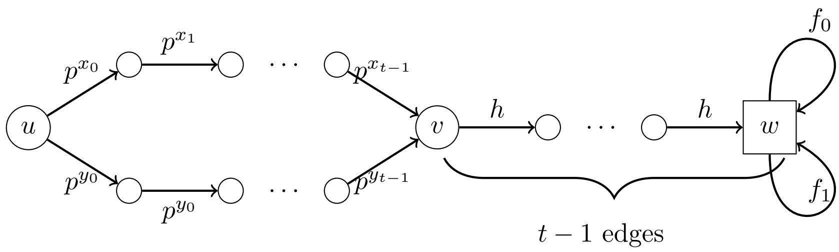
We now take sequences $\{I_n^0\}_{n=0}^\infty \in \{0,1\}^\omega$ and $\{I_n^1\}_{n=0}^\infty \in \{0,1\}^\omega$, satisfying Lemma 11, and define $p^0, p^1 \colon \Omega \to \Omega$ as follows:

$$p^0((n,b)) = \Big(n+1, b \oplus I_n^0\Big), \qquad p^1((n,b)) = \Big(n+1, b \oplus I_n^1\Big).$$

Note that p^0, p^1 are incremental. Hence, $p^0, p^1 \in \mathbb{M}$. We set $C = \{f_0, f_1, h, p^0, q^1\}$ and show that $\mathsf{ChrMem}(\mathsf{RL} \cap C^\omega) = +\infty$. For that, for every $Q \in \mathbb{N}$, we show that $\mathsf{ChrMem}(\mathsf{RL} \cap C^\omega) > Q$. Fix any $Q \in \mathbb{N}$ and let $t \in \mathbb{N}$ and $x = x_0 \ldots x_{t-1} \in \{0,1\}^t, y = y_0 \ldots y_{t-1} \in \{0,1\}^t$ be such that x and y are Q-indistinguishable and

$$I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}} \neq I_0^{y_0} \oplus \ldots \oplus I_{t-1}^{y_{t-1}} \tag{3}$$

(existence of such t, x and y is guaranteed by Lemma 11). Consider the following arena:



All circles are controlled by Antagonist and the square is controlled by Protagonist. The game starts at the node u. We claim that Protagonist has a winning strategy w.r.t. RL. Indeed, in the beginning, Antagonist has two choices – to go through $p^{x_0}, p^{x_1}, \ldots, p^{x_{t-1}}$ or to go through $p^{y_0}, p^{y_1}, \ldots, p^{y_{t-1}}$. In any case, upon reaching v, the first coordinate of the ant will be t (both p^0 and p^1 always increase the first coordinate of the ant by 1). Then we go through $t-1$ edges colored by h. As a result, the position of the ant at w will be either $(1,0)$ or $(1,1)$. If it is $(1,0)$, then Protagonist wins by always using f_0. If it is $(1,1)$, then Protagonist wins by always using f_1.

It remains to show that Protagonist has no winning strategy with at most Q states of chromatic memory. Indeed, consider any Protagonist's strategy S with at most Q states of chromatic memory. Our goal is to show that S is not winning. It is built on top of some chromatic memory structure with at most Q states. This memory structure, by definition, only reads colors of edges. Hence, when we go from u to v, we either feed $p^{x_0}p^{x_1}\ldots p^{x_{t-1}} \in \{p^0, p^1\}^t$ or $p^{y_0}p^{y_1}\ldots p^{y_{t-1}} \in \{p^0, p^1\}^t$ to it. We claim that S comes into the same state on these two sequences. Indeed, up to renaming letters of the alphabet, we may assume that we feed $x = x^0 \ldots x^{t-1}$ and $y = y^0 \ldots y^{t-1}$ to the memory structure of S. By definition, x and y are Q-indistinguishable. Since the memory structure of S has at most Q states, it must come into the same state on x and y. We conclude the state of S at v, and hence at w, will be the same in both possible plays. Thus, S acts identically at w in these two plays.

At the same time, there are two different possible positions of the ant at w. More specifically, if the Antagonist goes through $p^{x_0}p^{x_1}\ldots p^{x_{t-1}}$, the position of the ant will be

$$\begin{aligned}\underbrace{h \circ \ldots \circ h}_{t-1} \circ p^{x_{t-1}} \circ \ldots \circ p^{x_1} \circ p^{x_0}(\overline{\mathbf{0}}) &= \underbrace{h \circ \ldots \circ h}_{t-1} \circ p^{x_{t-1}} \circ \ldots \circ p^{x_1}((1, I_0^{x_0})) \\ &= \underbrace{h \circ \ldots \circ h}_{t-1}((t, I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}}) \\ &= (1, I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}}).\end{aligned}$$

Likewise, if the Antagonist goes through $p^{y_0}p^{y_1}\ldots p^{y_{t-1}}$, the position of the ant will be

$$\underbrace{h \circ \ldots \circ h}_{t-1} \circ p^{x_{t-1}} \circ \ldots \circ p^{x_1} \circ p^{x_0}(\overline{\mathbf{0}}) = (1, I_0^{y_0} \oplus \ldots \oplus I_{t-1}^{y_{t-1}}).$$

Since $I_0^{x_0} \oplus \ldots \oplus I_{t-1}^{x_{t-1}} \neq I_0^{y_0} \oplus \ldots \oplus I_{t-1}^{y_{t-1}}$ by (3), we conclude that both $(1,0)$ and $(1,1)$ are possible positions of the ant at w. But once again, S acts in the same way at w in both cases. Assume first that S plays the f_0-edge when it first reaches w. Then S looses if the ant reached w being in $(1,1)$. Indeed, after S plays its first move at w, the position of the ant becomes $f_0((1,1)) = (2,1)$. If the first coordinate of the ant is 2 or more, both f_0 and f_1 increase it by 1. Hence, no matter what Protagonist does afterwards, the ant will get infinitely high in Ω. Likewise, if the first move of S at w is the f_1-edge, then it looses if the ant reaches w being in $(1,0)$.

References

1 Bloem, R., Chatterjee, K., and Jobstmann, B. Graph games and reactive synthesis. In *Handbook of Model Checking*. Springer, 2018, pp. 921–962.

2 Bouyer, P., Randour, M., and Vandenhove, P. Characterizing Omega-Regularity Through Finite-Memory Determinacy of Games on Infinite Graphs. In *39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022)* (Dagstuhl, Germany, 2022), P. Berenbrink and B. Monmege, Eds., vol. 219 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, pp. 16:1–16:16.

3 Bouyer, P., Roux, S. L., Oualhadj, Y., Randour, M., and Vandenhove, P. Games Where You Can Play Optimally with Arena-Independent Finite Memory. In *31st International Conference on Concurrency Theory (CONCUR 2020)* (Dagstuhl, Germany, 2020), I. Konnov and L. Kovács, Eds., vol. 171 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, pp. 24:1–24:22.

4 Büchi, J. R., and Landweber, L. H. Solving sequential conditions by finite-state strategies. *Transactions of the American Mathematical Society 138* (1969), 295–311.

5 Casares, A. On the Minimisation of Transition-Based Rabin Automata and the Chromatic Memory Requirements of Muller Conditions. In *30th EACSL Annual Conference on Computer Science Logic (CSL 2022)* (Dagstuhl, Germany, 2022), F. Manea and A. Simpson, Eds., vol. 216 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, pp. 12:1–12:17.

6 Casares, A., Colcombet, T., and Lehtinen, K. On the size of good-for-games rabin automata and its link with the memory in muller games. *arXiv preprint arXiv:2204.11333* (2022).

7 Chatterjee, K., Doyen, L., Henzinger, T. A., and Raskin, J.-F. Generalized mean-payoff and energy games. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010)* (2010), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

8 Colcombet, T., Fijalkow, N., and Horn, F. Playing safe. In *34th International Conference on Foundation of Software Technology and Theoretical Computer Science (FSTTCS 2014)* (2014), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

9 Dziembowski, S., Jurdzinski, M., and Walukiewicz, I. How much memory is needed to win infinite games? In *Proceedings of Twelfth Annual IEEE Symposium on Logic in Computer Science* (1997), IEEE, pp. 99–110.

10 Kopczyński, E. Omega-regular half-positional winning conditions. In *International Workshop on Computer Science Logic* (2007), Springer, pp. 41–53.

11 Kopczyński, E. *Half-positional determinacy of infite games.* PhD thesis, Warsaw University, 2008.

12 Kozachinskiy, A. State complexity of chromatic memory in infinite-duration games. *arXiv preprint arXiv:2201.09297* (2022).

13 Le Roux, S. Time-aware uniformization of winning strategies. In *Conference on Computability in Europe* (2020), Springer, pp. 193–204.

14 Zielonka, W. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science 200*, 1-2 (1998), 135–183.