Trading Bounds for Memory in Games with Counters*

Nathanaël Fijalkow[1,2], Florian Horn[1], Denis Kuperberg[2,3], and Michał Skrzypczak[1,2]

[1] LIAFA, Université Paris 7
[2] Institute of Informatics, University of Warsaw
[3] Onera/DTIM, Toulouse and IRIT, University of Toulouse

Abstract. We study two-player games with counters, where the objective of the first player is that the counter values remain bounded. We investigate the existence of a trade-off between the size of the memory and the bound achieved on the counters, which has been conjectured by Colcombet and Löding. We show that unfortunately this conjecture does not hold: there is no trade-off between bounds and memory, even for finite arenas. On the positive side, we prove the existence of a trade-off for the special case of thin tree arenas.

1 Introduction

This paper studies finite-memory determinacy for games with counters. The motivation for this investigation comes from the theory of regular cost functions, which we discuss now.

Regular cost functions. The theory of regular cost functions is a *quantitative* extension of the notion of regular languages, over various structures (words and trees). More precisely, it expresses *boundedness questions*. A typical example of a boundedness question is: given a regular language $L \subseteq \{a, b\}^*$, does there exist a bound N such that all words from L contain at most N occurences of a?

This line of work already has a long history: it started in the 80s, when Hashiguchi, and then later Leung, Simon and Kirsten solved the *star-height problem* by reducing it to boundedness questions [10, 15, 13, 11]. Both the logics MSO+$\mathbb{U}$ and later cost MSO (as part of the theory of regular cost functions) emerged in this context [3, 4, 6, 8], as quantitative extensions of the notion of regular languages that can express boundedness questions.

Consequently, developing the theory of regular cost functions comes in two flavours: the first is using it to reduce various problems to boundedness questions, and the second is obtaining decidability results for the boundedness problem for cost MSO over various structures.

* The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement 259454 (GALE).

For the first point, many problems have been reduced to boundedness questions. The first example is the star-height problem over words [11] and over trees [8], followed for instance by the boundedness question for fixed points of monadic formulae over finite and infinite words and trees [2]. The most important problem that has been reduced to a boundedness question is to decide the Mostowski hierarchy for infinite trees [7].

For the second point, it has been shown that over finite words and trees, a significant part of the theory of regular languages can successfully be extended to the theory of regular cost functions, yielding notions of regular expressions, automata, semigroups and logics that all have the same expressive power, and that extend the standard notions. In both cases, algorithms have been constructed to answer boundedness questions.

However, extending the theory of regular cost functions to infinite trees seems to be much harder, and the major open problem there is the decidability of cost MSO over infinite trees.

LoCo conjecture. Colcombet and Löding pointed out that the only missing point to obtain the decidability of cost MSO is a finite-memory determinacy result for games with counters. More precisely, they conjectured that there exists a trade-off between the size of the memory and the bound achieved on the counters [5]. So far, this conjecture resisted both proofs and refutations, and one of the only non-trivial positive case known is due to Vanden Boom [16], which implied the decidability of the weak variant of cost MSO over infinite trees, later generalized to quasi-weak cost MSO in [1]. Unfortunately, Quasi-Weak cost MSO is strictly weaker than cost MSO, and this leaves open the question whether cost MSO is decidable.

Contributions. In this paper, we present two contributions:

- There is no trade-off, even for finite arenas, which disproves the conjecture,
- There is a trade-off for the special case of thin tree arenas.

Our first contribution does not imply the undecidability of cost MSO, it rather shows that proving the decidability will involve subtle combinatorial arguments that are yet to be understood.

Structure of this document. The definitions are given in Section 2. We state the conjecture in Section 3. Section 4 disproves the conjecture. Section 5 proves that the conjecture holds for the special case of thin tree arenas.

Acknowledgments. The unbounded number of fruitful discussions we had with Thomas Colcombet and Mikołaj Bojańczyk made this paper possible.

2 Definitions

Arenas. The games we consider are played by two players, Eve and Adam, over potentially infinite graphs called arenas[4]. Formally, an arena $\mathcal{G}$ consists of a directed graph (V, E) whose vertex set is divided into vertices controlled by Eve

[4] We refer to [9] for an introduction to games.

(V_E) and vertices controlled by Adam (V_A). A token is initially placed on a given initial vertex v_0, and the player who controls this vertex pushes the token along an edge, reaching a new vertex; the player who controls this new vertex takes over, and this interaction goes on forever, describing an infinite path called a *play*. Finite or infinite plays are paths in the graphs, seen as sequences of edges, typically denoted π. In its most general form, a strategy for Eve is a mapping $\sigma : E^* \cdot V_E \to E$, which given the history played so far and the current vertex picks the next edge. We say that a play $\pi = e_0 e_1 e_2 \dots$ is consistent with σ if $e_{n+1} = \sigma(e_0 \cdots e_n \cdot v_n)$ for every n with $v_n \in V_E$.

Winning conditions. A winning condition for an arena is a set of a plays for Eve, which are called the winning plays for Eve (the other plays are winning for Adam). A game consists of an arena $\mathcal{G}$ and a winning condition W for this arena, it is usually denoted $(\mathcal{G}, W)$.

A strategy for Eve is winning for a condition, or ensures this condition, if all plays consistent with the strategy belong to the condition. For a game $(\mathcal{G}, W)$, we denote $\mathcal{W}_E(\mathcal{G}, W)$ the winning region of Eve, *i.e.* the set of vertices from which Eve has a winning strategy.

Here we will consider the classical parity condition as well as *quantitative* bounding conditions.

The parity condition is specified by a colouring function $\Omega : V \to \{0, \dots, d\}$, requiring that the maximum color seen infinitely often is even. The special case where $\Omega : V \to \{1, 2\}$ corresponds to Büchi conditions, denoted Büchi(F) where $F = \{v \in V \mid \Omega(v) = 2\}$, and $\Omega : V \to \{0, 1\}$ corresponds to CoBüchi conditions, denoted CoBüchi(F) where $F = \{v \in V \mid \Omega(v) = 1\}$. We will also consider the simpler conditions Safe(F) and Reach(F), for $F \subseteq V$: the first requires to avoid F forever, and the second to visit a vertex from F at least once.

The bounding condition B is actually a family of winning conditions with an integer parameter $B = \{B(N)\}_{N \in \mathbb{N}}$. We call it a quantitative condition because it is monotone: if $N < N'$, all the plays in $B(N)$ also belong to $B(N')$.

The counter actions are specified by a function $c : E \to \{\varepsilon, i, r\}^k$, where k is the number of counters: each counter can be incremented (i), reset (r), or left unchanged (ε). The value of a play π, denoted $val(\pi)$, is the supremum of the value of all counters along the play. It can be infinite if one counter is unbounded. The condition $B(N)$ is defined as the set of plays whose value is less than N.

In this paper, we study the condition B-parity, where the winning condition is the intersection of a bounding condition and a parity condition. The value of a play that satisfies the parity condition is its value according to the bounding condition. The value of a play which does not respect the parity condition is ∞. We often consider the special case of B-reachability conditions, denoted B Until F. In such cases, we assume that the game stops when it reaches F.

Given an initial vertex v_0, the value $val(v_0)$ is:

$$\inf_{\sigma} \sup_{\pi} \{val(\pi) \mid \pi \text{ consistent with } \sigma \text{ starting from } v_0\} .$$

Finite-memory strategies. A *memory structure* $\mathcal{M}$ for the arena $\mathcal{G}$ consists of a set M of memory states, an initial memory state $m_0 \in M$ and an update function $\mu : M \times E \to M$. The update function takes as input the current memory state and the chosen edge to compute the next memory state, in a deterministic way. It can be extended to a function $\mu : E^* \cdot V \to M$ by defining $\mu^*(v) = m_0$ and $\mu^*(\pi \cdot (v, v')) = \mu(\mu^*(\pi \cdot v), (v, v'))$.

Given a memory structure $\mathcal{M}$, a strategy is induced by a next-move function $\sigma : V_E \times M \to E$, by $\sigma(\pi \cdot v) = \sigma(v, \mu^*(\pi \cdot v))$. Note that we denote both the next-move function and the induced strategy σ. A strategy with memory structure $\mathcal{M}$ has finite memory if M is a finite set. It is *memoryless*, or *positional* if M is a singleton: it only depends on the current vertex. Hence a memoryless strategy can be described as a function $\sigma : V_E \to E$.

An arena $\mathcal{G}$ and a memory structure $\mathcal{M}$ for $\mathcal{G}$ induce the expanded arena $\mathcal{G} \times \mathcal{M}$ where the current memory state is stored explicitly together with the current vertex: the vertex set is $V \times M$, the edge set is $E \times \mu$, defined by: $((v, m), (v', m')) \in E'$ if $(v, v') \in E$ and $\mu(m, (v, v')) = m'$. There is a natural one-to-one correspondence between strategies in $\mathcal{G} \times \mathcal{M}$ using $\mathcal{M}'$ as memory structure and strategies in $\mathcal{G}$ using $\mathcal{M} \times \mathcal{M}'$ as memory structure.

3 The conjecture

In this section, we state the conjecture [5], and explain how positive cases of this conjecture imply the decidability of cost MSO.

3.1 Statement of the conjecture

There exists $\mathrm{mem} : \mathbb{N}^2 \to \mathbb{N}$ and $\alpha : \mathbb{N}^3 \to \mathbb{N}$ such that
for every B-parity game with k counters, $d + 1$ colors and initial vertex v_0,
there exists a strategy σ using $\mathrm{mem}(d, k)$ memory states, ensuring
$B(\alpha(d, k, \mathit{val}(v_0))) \cap \mathrm{Parity}(\Omega)$.

The function α is called a trade-off function: if there exists a strategy ensuring $B(N) \cap \mathrm{Parity}(\Omega)$, then there exists a strategy with *small* memory that ensures $B(\alpha(d, k, N)) \cap \mathrm{Parity}(\Omega)$. So, at the price of increasing the bound from N to $\alpha(d, k, N)$, one can use a strategy using a small memory structure.

To get a better understanding of this conjecture, we show three simple facts:

1. why reducing memory requires to increase the bound,
2. why the memory bound mem depends on the number of counters k,
3. why a weaker version of the conjecture holds, where mem depends on the value.

For the first point, we present a simple game, represented in Figure 1. It involves one counter and the condition B Until F. Starting from v_0, the game moves to v and sets the value of the counter to N. The objective of Eve is to take the edge to the right to F. However, this costs N increments, so if she wants the counter value to remain smaller than N she has to set its value to 0 before taking this edge. She has N options: for $\ell \in \{1, \ldots, N\}$, the ℓ^{th} option consists in going to u_ℓ, involving the following actions:

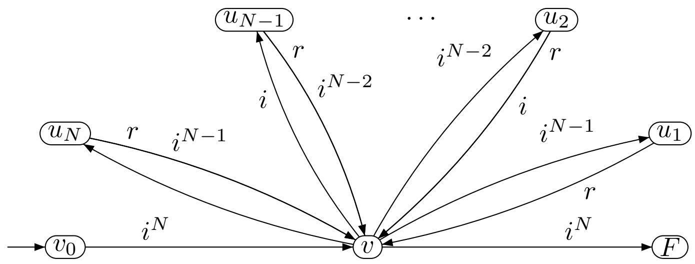

Fig. 1. A trade-off is necessary.

- first, take $N - \ell$ increments,
- then, reset the counter,
- then, take $\ell - 1$ increments, setting the value to $\ell - 1$.

It follows that there is a strategy for Eve to ensure $B(N)$ Until F, which consists in going successively through u_N, u_{N-1}, and so on, until u_1, and finally to F. Hence to ensure that the bound is always smaller than N, Eve needs $N + 1$ memory states.

However, if we consider the bound $2N$ rather than N, then Eve has a very simple strategy, which consists in going directly to F, using no memory at all. This is a simple example of a trade-off: to ensure the bound N, Eve needs $N+1$ memory states, but to ensure the worse bound $2N$, she has a positional strategy.

For the second point, consider the following game with k counters (numbered cyclically) and only one vertex, controlled by Eve. There are k self-loops, each incrementing a counter and resetting the previous one. Eve has a simple strategy to ensure $B(1)$, which consists in cycling through the loops, and uses k memory states. Any strategy using less than k memory states ensures no bound at all, as one counter would be incremented infinitely many times but never reset. It follows that the memory bound mem in the conjecture has to depend on k.

For the third point, we give an easy result that shows the existence of finite memory strategies whose size depends on the value, even without losing anything on the bound.

Lemma 1. *For every B-parity game with k counters and initial vertex v_0, there exists a strategy σ ensuring $B(val(v_0)) \cap Parity(\Omega)$ with $(val(v_0) + 1)^k$ memory states.*

The proof consists in composing the B-parity game with a memory structure that keeps track of the counter values up to the value, reducing the B-condition to a safety condition.

3.2 The interplay with cost MSO

The aim of the conjecture stated above is the following: if true, it implies the decidability of cost MSO over infinite trees. More precisely, the technical difficulty to develop the theory of regular cost functions over infinite trees is to obtain effective constructions between variants of automata with counters, and this is what this conjecture is about.

In the qualitative case (without counters), to obtain the decidability of MSO over infinite trees, known as Rabin's theorem [14], one transforms MSO formulae into equivalent automata. The complementation construction is the technical cornerstone of this procedure. The *key* ingredient for this is games, and specifically positional determinacy for parity games. Similarly, other classical constructions, to simulate either two-way or alternating automata by non-deterministic ones, make a crucial use of positional determinacy for parity games.

In the quantitative case now, Colcombet and Löding [8] showed that to extend these constructions, one needs a similar result on parity games with counters, which is the conjecture we stated above.

So far, there is only one positive instance of this conjecture, which is the special case of B-Büchi games over chronological arenas[5].

Theorem 1 ([16]). *For every B-Büchi game played over a chronological arena with k counters and initial vertex v_0, Eve has a strategy ensuring $B(k^k \cdot val(v_0)^{2k}) \cap$ $Büchi(F)$ with $2 \cdot k!$ memory states.*

This was the key ingredient in proving the decidability of weak cost MSO over infinite trees.

4 No trade-off over Finite Arenas

In this section, we show that the conjecture does not hold, even for finite arenas.

Theorem 2. *For all K, for all N, there exists a B-reachability game played over a finite arena $G_{K,N}$ with one counter and an initial vertex such that:*

- *there exists a 3^K memory states strategy ensuring $B(K(K+3))$ Until F,*
- *no $K+1$ memory states strategy ensure $B(N)$ Until F.*

We proceed in two steps. The first is an example giving a lower bound of 3, and the second is a nesting of this first example.

4.1 A first lower bound of 3

The game $\mathcal{G}_1$ giving a lower bound of 3 is represented in Figure 2. The condition is B Until F, with only one counter. In this game, Eve is torn between going to the right to reach F, which implies incrementing the counter, and going to the left, to reset the counter. The actions of Eve from the vertex u_n are:

- *increment*, and go one step to the right, to v_{n-1},
- *reset*, and go two steps to the left, to v_{n+2}.

The actions of Adam from the vertex v_n are:

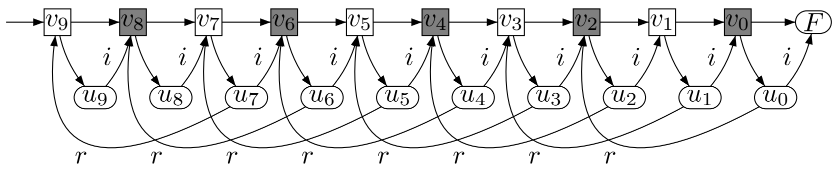

Fig. 2. Part of the game $\mathcal{G}_1$, where Eve needs 3 memory states. The colors on the vertices are used to construct a 3 memory states strategy.

- *play*, and go down to u_n,
- *skip*, and go to v_{n-1}.

Theorem 3. *In $\mathcal{G}_1$, from v_N:*

- *Eve has a 4 memory states strategy ensuring $B(3)$ Until F,*
- *Eve has a 3 memory states strategy ensuring $B(4)$ Until F,*
- *For all N, no 2 memory states strategy ensures $B(N)$ Until F.*

The first item follows from Lemma 1. However, to illustrate the properties of the game $\mathcal{G}_1$ we will provide a concrete strategy with 4 memory states that ensures $B(3)$ Until F. The memory states are i_1, i_2, i_3 and r, linearly ordered by $i_1 < i_2 < i_3 < r$. With the memory states i_1, i_2 and i_3, the strategy chooses to increment, and updates its memory state to the next memory state. With the memory state r, the strategy chooses to reset, and updates its memory state to i_1. This strategy satisfies a simple invariant: it always resets to the right of the previous reset, if any.

We show how to save one memory state, at the price of increasing the bound by one: we construct a 3 memory states strategy ensuring $B(4)$ Until F. The idea, as represented in Figure 2, is to color every second vertex and to use this information to track progress. The 3 memory states are called i, j and r. The update is as follows: the memory state is unchanged in uncoloured (white) states, and switches from i and j and from j to r on gray states. The strategy is as follows: in the two memory states i and j, Eve chooses to increment, and in r she chooses to reset. As for the previous strategy, this strategy ensures that it always resets to the right of the previous reset, if any.

4.2 General lower bound

We now push the example above further. A first approach is to modify $\mathcal{G}_1$ by increasing the length of the resets, going ℓ steps to the left rather than only 2. However, this does not give a better lower bound: there exists a 3 memory states strategy in this modified game that ensures twice the value, following the same ideas as presented above.

[5] The definition of chronological arenas is given in Section 5.

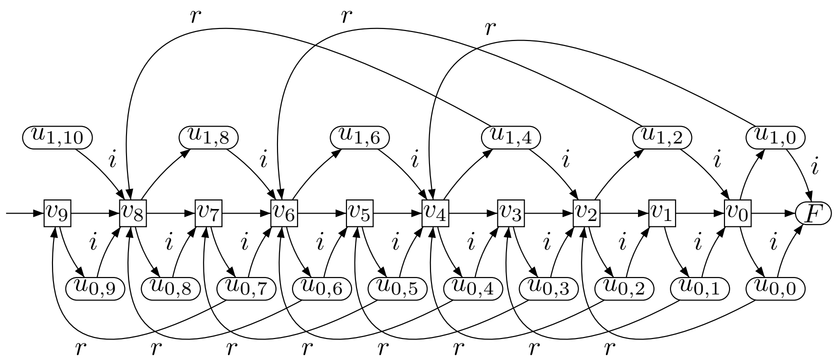

Fig. 3. The game with two levels.

We construct $\mathcal{G}_{K,N}$, a nesting of the game $\mathcal{G}_1$ with K levels. In Figure 3, we represented the interaction between two levels. Roughly speaking, the two levels are independent, so we play both games at the same time. Those two games use different timeline. For instance, in Figure 3, the bottom level is based on $(+1, -2)$ (an increment goes one step to the right, a reset two steps to the left), and the top level is based on $(+2, -4)$. This difference in timeline ensures that a strategy for Eve needs to take care somehow independently of each level, ensuring that the number of memory states depends on the number of levels.

Theorem 4. *In $\mathcal{G}_{K,N}$, from some initial vertex:*

- *Eve has a 3^K memory states strategy ensuring $B(K(K+3))$ Until F,*
- *No $K+1$ memory states strategy ensures $B(N)$ Until F.*

5 Existence of a trade-off for thin tree arenas

In this section, we prove that the conjecture holds for the special case of thin tree arenas[6].

Theorem 5. *There exist two functions $mem : \mathbb{N}^2 \to \mathbb{N}$ and $\alpha : \mathbb{N}^4 \to \mathbb{N}$ such that for every B-parity game played over a thin tree arena of width W with k counters, $d+1$ colors and initial vertex v_0, Eve has a strategy to ensure $B(k^k \cdot val(v_0)^{2k} \cdot \alpha(d, k, W+2, val(v_0))) \cap Parity(\Omega)$, with $W \cdot 3^k \cdot 2k! \cdot mem(d,k)$ memory states.*

The functions α and mem are defined inductively. Since the proof will work by induction on the number of colors, removing the least important color, we define four functions:

[6] The definitions of word and thin tree arenas are given in Subsection 5.1.

– α_0 and mem_0 when the set of colors is $\{0, \ldots, d\}$,
– α_1 and mem_1 when the set of colors is $\{1, \ldots, d\}$:

$$\alpha_0(d, k, W, N) = \alpha_1(d-1, k+1, 3W, W \cdot (N+1)^k),$$

$$\alpha_1(d, k, W, N) = \begin{cases} k^k \cdot N^{2k} & \text{if } d = 2, \\ \alpha_0(d, k, 2W, N) & \text{otherwise,} \end{cases}$$

$$\mathrm{mem}_0(d, k) = 2 \cdot \mathrm{mem}_1(d-1, k+1),$$

$$\mathrm{mem}_1(d, k) = \begin{cases} 2 \cdot k! & \text{if } d = 1, \\ 2 \cdot \mathrm{mem}_0(d, k) & \text{otherwise.} \end{cases}$$

Note that the functions α and mem depend on the width parameter; this is sufficient for the intended applications of the LoCo conjecture, as the width corresponds to the size of the automaton.

We focus in this extended abstract on the intermediate result for the special case of word arenas.

Theorem 6. *There exists two functions mem* $: \mathbb{N}^2 \to \mathbb{N}$ *and* $\alpha : \mathbb{N}^4 \to \mathbb{N}$ *such that for every* B*-parity game played over a word arena of width* W *with* k *counters,* $d+1$ *colors and initial vertex* v_0*, Eve has a strategy to ensure* $B(\alpha(d, W, k, \mathit{val}(v_0))) \cap \mathit{Parity}(\Omega)$*, with* $\mathit{mem}(d, k)$ *memory states.*

5.1 Word and thin tree arenas

A (non-labelled binary) *tree* is a subset $T \subseteq \{0,1\}^*$ which is prefix-closed and non-empty. The elements of T are called *nodes*, and we use the natural terminology: for $n \in \{0,1\}^*$ and $\ell \in \{0,1\}$, the node $n \cdot \ell$ is a child of n, and a descendant of n if $\ell \in \{0,1\}^*$.

A (finite or infinite) branch π is a word in $\{0,1\}^*$ or $\{0,1\}^\omega$. We say that π is a branch of the tree T if $\pi \subseteq T$ (or every prefix of π belongs to T when π is infinite) and π is maximal satisfying this property. A tree is called *thin* if it has only countably many branches. For example, the full binary tree $T = \{0,1\}^*$ has uncountably many branches, therefore it is not thin.

Definition 1. *An arena is:*

– chronological *if there exists a function* $r : V \to \mathbb{N}$ *which increases by one on every edge: for all* $(v, v') \in E$*,* $r(v') = r(v) + 1$*.*
– a word arena of width W *if it is chronological, and for all* $i \in \mathbb{N}$*, the set* $\{v \in V \mid r(v) = i\}$ *has cardinal at most* W*.*
– a tree arena of width W *if there exists a function* $R : V \to \{0,1\}^*$ *such that*
  1. *for all* $n \in \{0,1\}^*$*, the set* $\{v \in V \mid R(v) = n\}$ *has cardinal at most* W*.*
  2. *for all* $(v, v') \in E$*, we have* $R(v') = R(v) \cdot \ell$ *for some* $\ell \in \{0,1\}$*.*

  It is a thin tree arena if $R(V)$ *is a thin tree.*

The notions of word and tree arenas naturally appear in the study of automata over infinite words and trees. Indeed, the acceptance games of such automata, which are used to define their semantics, are played on word or tree arenas. Furthermore, the width corresponds to the size of the automaton.

5.2 Existence of a trade-off for word arenas

We prove Theorem 6 by induction on the colors in the parity condition. Consider a B-parity game $\mathcal{G}$ with k counters and $d+1$ colors over a word arena of width W with initial vertex v_0. We examine two cases, depending whether the least important color (*i.e* the smallest) that appears is odd or even:

– if the set of colors is $\{1, \ldots, d\}$, then we construct a B-parity game $\mathcal{G}'$ using $\{2, \ldots, d\}$ as colors,
– if the set of colors is $\{0, \ldots, d\}$, then we construct a B-parity game $\mathcal{G}'$ using $\{1, \ldots, d\}$ as colors.

In both cases, we obtain from the induction hypothesis a winning strategy using few memory states in $\mathcal{G}'$, which we use to construct a winning strategy using few memory states in $\mathcal{G}$. The base case is given by Büchi conditions, and follows from Theorem 1.

Removing the least important color: the odd case The first case we consider is when the least important color is 1. The technical core of the construction is motivated by the technique used in [16]. It consists in slicing the game horizontally, such that in each slice the strategy σ ensures to see a vertex of color greater than 1. This way, the combination of a memory structure of size 2 and a safety condition expresses that some color greater than 1 should be seen infinitely often, allowing us to remove the color 1.

Removing the least important color: the even case The second case we consider is when the least important color is 0.

We explain the intuition for the case of CoBüchi conditions, *i.e* if there are only colors 0 and 1. Let $F = \{v \mid \Omega(v) = 1\}$. Define $X_0 = Y_0 = \emptyset$, and for $i \geq 1$:

$$\begin{cases} X_{i+1} = \mathcal{W}_E(\text{Safe}(F) \text{ WeakUntil } Y_i) \\ Y_{i+1} = \mathcal{W}_E(\text{Reach}(X_{i+1})) \end{cases}$$

The condition Safe(F) WeakUntil Y_i is satisfied by plays that do not visit F before Y_i: they may never reach Y_i, in which case they never reach F, or they reach Y_i, in which case they did not reach F before that.

We have $\bigcup_i Y_i = \mathcal{W}_E(\text{CoBüchi}(F))$. A winning strategy based on these sets has two aims: in X_i it avoids F ("Safe" mode) and in Y_i it attracts to the next X_i ("Attractor" mode). The key property is that since the arena is a word arena of width W where Eve can bound the counters by N, she only needs to alternate between modes a number of times bounded by a function of N and W. In other words, the sequence $(Y_i)_{i \in \mathbb{N}}$ stabilizes after a number of steps bounded by a function of N and W. A remote variant of this bounded-alternation fact can be found in [12]. Hence the CoBüchi condition can be checked using a new counter and a Büchi condition, as follows.

There are two modes: "Safe" and "Attractor". The Büchi condition ensures that the "Safe" mode is visited infinitely often. In the "Safe" mode, only vertices of color 0 are accepted; visiting a vertex of color 1 leads to the "Attractor" mode and increments the new counter. At any time, she can reset the mode to "Safe". The counter is never reset, so to ensure that it is bounded, Eve must change modes finitely often. Furthermore, the Büchi condition ensures that the final mode is "Safe", implying that the CoBüchi condition is satisfied.

For the more general case of parity conditions, the same idea is used, but as soon as a vertex of color greater than 1 is visited, then the counter is reset.

Define $\mathcal{G}'$:

$$V' = \begin{cases} V'_E = V_E \times \{A, S\} \ \cup \ \overline{V} \\ V'_A = V_A \times \{A, S\} \ . \end{cases}$$

After each edge followed, Eve is left the choice to set the flag to S. The set of choice vertices is denoted $\overline{V}$. We define E' and the counter actions.

$$E' = \begin{cases} (v, A) \xrightarrow{c(v,v'),\varepsilon} \overline{v'} & \text{if } (v, v') \in E, \\ (v, S) \xrightarrow{c(v,v'),\varepsilon} (v', S) & \text{if } (v, v') \in E \text{ and } \Omega(v') = 0, \\ (v, S) \xrightarrow{c(v,v'),i} (v', A) & \text{if } (v, v') \in E \text{ and } \Omega(v') = 1, \\ (v, S) \xrightarrow{c(v,v'),r} (v', S) & \text{if } (v, v') \in E \text{ and } \Omega(v') > 1, \\ \overline{v} \xrightarrow{\varepsilon} (v, A) \text{ and } \overline{v} \xrightarrow{\varepsilon} (v, S) \end{cases}$$

Equip the arena $\mathcal{G}'$ with the colouring function Ω' defined by

$$\Omega'(v, m) = \begin{cases} 1 & \text{if } m = A, \\ 2 & \text{if } \Omega(v) = 0 \text{ and } m = S, \\ \Omega(v) & \text{otherwise.} \end{cases}$$

Remark that Ω' uses one less color than Ω, since no vertices have color 0 for Ω'.

Before stating the equivalence between $\mathcal{G}$ and $\mathcal{G}'$, we formalise the property mentioned above, that in word arenas Eve does not need to alternate an unbounded number of times between the modes "Safe" and "Attractor".

Lemma 2. *Let G be a word arena of width W, and a subset F of vertices such that every path in G contains finitely many vertices in F. Define the following sequence of subsets of vertices $X_0 = \emptyset$, and for $i \geq 0$*

$$\begin{cases} X_{2i+1} = \left\{ v \;\middle|\; \begin{array}{c} \textit{all paths from } v \textit{ contain no vertices in } F \\ \textit{before the first vertex in } X_{2i}, \textit{ if any} \end{array} \right\}, \\ X_{2i+2} = \left\{ v \;\middle|\; \textit{all paths from } v \textit{ are finite or lead to } X_{2i+1} \right\}. \end{cases}$$

We have $X_0 \subseteq X_1 \subseteq X_2 \subseteq \cdots$, and X_{2W} covers the whole arena.

Denote $N = val(v_0)$.

Lemma 3.

1. *There exists a strategy σ' in $\mathcal{G}'$ that ensures $B(W \cdot (N+1)^k) \cap Parity(\Omega')$.*
2. *Let σ' be a strategy in $\mathcal{G}'$ ensuring $B(N') \cap Parity(\Omega')$ with K memory states, then there exists σ a strategy in $\mathcal{G}$ that ensures $B(N') \cap Parity(\Omega)$ with $2K$ memory states.*

Conclusion

We studied the existence of a trade-off between bounds and memory in games with counters, such as conjectured by Colcombet and Löding. We proved that there is no such trade-off in general, but that under some structural restrictions, as thin tree arenas, the conjecture holds.

We believe that the conjecture holds for all tree arenas, which would imply the decidability of cost MSO over infinite trees. A proof of this result would probably involve advanced combinatorial arguments, and require a deep understanding of the structure of tree arenas.

References

1. Achim Blumensath, Thomas Colcombet, Denis Kuperberg, Paweł Parys, and Michael Vanden Boom. Two-way cost automata and cost logics over infinite trees. In *CSL-LICS*, pages 16–26, 2014.
2. Achim Blumensath, Martin Otto, and Mark Weyer. Decidability results for the boundedness problem. *Logical Methods in Computer Science*, 10(3), 2014.
3. Mikołaj Bojańczyk. A bounding quantifier. In *CSL*, pages 41–55, 2004.
4. Mikołaj Bojańczyk and Thomas Colcombet. Bounds in ω-regularity. In *LICS*, pages 285–296, 2006.
5. Thomas Colcombet. Fonctions régulières de coût. Habilitation Thesis, 2013.
6. Thomas Colcombet. Regular cost functions, part I: logic and algebra over words. *Logical Methods in Computer Science*, 9(3), 2013.
7. Thomas Colcombet and Christof Löding. The non-deterministic Mostowski hierarchy and distance-parity automata. In *ICALP (2)*, pages 398–409, 2008.
8. Thomas Colcombet and Christof Löding. Regular cost functions over finite trees. In *LICS*, pages 70–79, 2010.
9. Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.
10. Kosaburo Hashiguchi. Improved limitedness theorems on finite automata with distance functions. *Theoretical Computer Science*, 72(1):27–38, 1990.
11. Daniel Kirsten. Distance desert automata and the star height problem. *ITA*, 39(3):455–509, 2005.
12. Orna Kupferman and Moshe Y. Vardi. Weak alternating automata are not that weak. In *5th Israeli Symposium on Theory of Computing and Systems*, pages 147–158. IEEE Computer Society Press, 1997.
13. Hing Leung. Limitedness theorem on finite automata with distance functions: An algebraic proof. *Theoretical Computuer Science*, 81(1):137–145, 1991.
14. Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the AMS*, 141:1–23, 1969.
15. Imre Simon. On semigroups of matrices over the tropical semiring. *ITA*, 28(3-4):277–294, 1994.
16. Michael Vanden Boom. Weak cost monadic logic over infinite trees. In *MFCS*, pages 580–591, 2011.