Pushdown games with unboundedness and regular conditions

A. Bouquet[1], O. Serre[2], I. Walukiewicz[3]

Abstract

We consider infinitary two-player perfect information games defined over graphs of configurations of a pushdown automaton. We show how to solve such games when winning conditions are Boolean combinations of a Büchi condition and a new condition that we call unboundedness. An infinite play satisfies the unboundedness condition if there is no bound on the size of the stack during the play. We show that the problem of deciding a winner in such games is EXPTIME-complete.

1 Introduction

Infinite two-player games with perfect information are one of the central notions in verification and in theory of automata on infinite words and trees. The result on existence of finite memory strategies for games with Muller conditions is a necessary ingredient of most automata constructions [13, 15, 17]. The other important results are those describing ways of solving a game, i.e., finding out from which vertices a given player has a winning strategy [16, 20]. The mu-calculus model checking problem is an instance of a game solving problem [8, 7, 9]. The construction of discrete controllers can be also reduced to the problem of solving games [1].

In the most standard setting of verification and synthesis one uses just finite games. Still the model of pushdown games has attracted some attention [12, 2, 10, 11, 4, 5, 14, 18]. In this model a graph of a game is given by a configuration graph of a pushdown automaton. Such games are more

[1]LaBRI, Université Bordeaux I, 351 cours de la Libération, 33405 Talence Cedex, France. bouquet@labri.fr

[2]LIAFA, Université Paris VII, 2 place Jussieu, Paris, France. serre@liafa.jussieu.fr

[3]LaBRI, CNRS, 351 cours de la Libération, 33405 Talence Cedex, France. Corresponding author, igw@labri.fr

suitable to model phenomena like procedure invocation as stack is explicitly present in the model.

Standard, Muller or parity winning conditions, are very useful and natural for the applications mentioned above. Their expressiveness is also satisfactory as any game with S1S definable (i.e. regular) winning conditions can be reduced to a game with Muller or parity conditions. As noted in [6] for pushdown games the situation changes and there exists "natural winning conditions exploiting the infinity of pushdown transition graphs".

We propose a new winning condition for pushdown games that we call unboundedness: an infinite play satisfies the unboundedness condition if there is no bound on the size of the stack during the play. We consider Boolean combinations of this condition and the parity condition, for example, a condition saying that a stack is unbounded and some state appears infinitely often. We characterize conditions for which there is a strategy with finite memory for both players. We show that the problem of deciding a winner in pushdown games with Boolean combinations of Büchi and unboundedness conditions is EXPTIME-complete (in the size of the automaton defining the game graph).

This research reported here was motivated by the paper of Cachat, Duparc and Thomas [6]. They consider the same class of games as we do here, but only a single winning condition: some configuration repeats infinitely often on the play. The negation of this condition is "strict stack unboundedness": every configuration appears only finitely often on the play. While "strict unboundedness" is a more restrictive condition than unboundedness, we show that the two are equivalent if considered in disjunction with a parity condition. In particular, in a pushdown game, a position is winning with unboundedness condition if and only if it is winning with strict unboundedness condition.

As mentioned above, numerous verification and synthesis problems are reducible to the problem of solving games. Hence, the algorithms that we propose here can be used to extend the class of properties that can be model checked or for which synthesis is possible. To give a simple example, our algorithms can be used to solve the problem of checking that on every path of a given pushdown system where the stack is unbounded some LTL property holds.

In summary we show the following results:

- In section 3, for the conditions of the form "parity or unboundedness" we have that from every configuration one of the players has a memoryless winning strategy (Theorem 1).
- In section 4, for every Boolean combination of conditions "states from

a given set appear infinitely often" and "unboundedness", there is an EXPTIME-algorithm deciding who is the winner in a given configuration (Theorem 2). The complete proofs are given in sections 5, 6, 7, and 8.

- For games with the condition "states from a given set appear infinitely often" and "unboundedness" it is shown in subsection 8.1 that player 0 may need infinite memory in order to win. Hence it is a rare case of a game which is algorithmically tractable but does not admit finite memory strategies (Example on page 32).

The proof methods in all the cases are reductions to finite graphs of exponential size (in the size of the pushdown system) but with a linear number of colors [18, 19]. We do not known how to adapt other methods of solving pushdown games to the extension presented here. In particular it is not evident how to use the elegant method from [11] if only because of non-existence of memoryless strategies as demonstrated in the example on page 32.

2 Definitions

The set of words (finite sequences) over an alphabet X is noted X^* and $\varepsilon \in X^*$ is the empty word. The set of infinite words (sequences of type ω) is denoted by X^ω.

Infinite two-player games An infinite two-player game on a finite or infinite graph (V, E) is a tuple $G = \langle V, V_0, V_1, E, Acc \subseteq V^\omega \rangle$ where (V_0, V_1) is a partition of V. The set of vertices V_0 describes the positions for player 0, and V_1 those for player 1. Whereas Acc defines the infinitary winning condition. In figures, we will denote positions of player 0 by ovals and those of player 1 by squares.

Two players, player 0 and player 1, play on G by moving a token between vertices. A play from an initial position $p_0 \in V_0$ proceeds as follows : player 0 moves the token to a new position p_1; then the player to whom p_1 belongs, makes a move reaching p_2 and so on. Similarly, we define a play starting in V_1, where player 1 begins. If one of the players cannot make a move, the other player wins. Otherwise, the play is infinite and results in an infinite path $\vec{p} = p_0 p_1 p_2 \ldots \in V^\omega$ in the game graph. Player 0 wins if $\vec{p} \in Acc$, otherwise player 1 is the winner.

Pushdown systems A *pushdown system* is a tuple $\mathcal{A} = \langle Q, \Gamma, \Delta \rangle$ where Q is a finite set of states, Γ is a finite set of stack symbols, and

$$\Delta : Q \times \Gamma \to \mathcal{P}(\{pop(q), push(q, b) : q \in Q, b \in \Gamma\})$$

is the transition relation. A *configuration* of $\mathcal{A}$ is a pair (q, u) with $q \in Q$ and $u \in \Gamma^*$. It denotes a global state of the pushdown system which consists of a control state and a contents of the stack; the top of the stack is described by the first letter of the word u.

A pushdown system $\mathcal{A}$ defines an infinite graph, $Gr(\mathcal{A})$ called *pushdown graph* whose the nodes are the configurations of $\mathcal{A}$ and the edges are defined by the transitions, i.e., from a node (p, au) we have edges to:

(q, bau) whenever $push(q, b) \in \Delta(p, a)$;

(q, u) whenever $pop(q) \in \Delta(p, a)$.

Observe that any configuration with an empty stack has no successors. A degree of every node is finite and bounded by $|Q|(1 + |\Gamma|)$.

Pushdown games We have mentioned above that a pushdown system $\mathcal{A} = \langle Q, \Gamma, \Delta \rangle$ defines a pushdown graph $Gr(\mathcal{A}) = \langle V, E \rangle$. Now suppose that we have a partition (Q_0, Q_1) of Q and an acceptance condition $Acc \subseteq V^\omega$. These allow to define a pushdown game $\langle V, V_0, V_1, E, Acc \rangle$ where:

$$V_0 = \{(p, u) : p \in Q_0, u \in \Gamma^*\} \qquad V_1 = \{(p, u) : p \in Q_1, u \in \Gamma^*\}$$

Strategies, winning positions and determinacy A *strategy* for player 0 is a function $\sigma : V^*V_0 \to V$ assigning to every partial play $\vec{p} = p_0p_1 \dots p_n$ ending in a vertex p_n from V_0 a vertex $\sigma(\vec{p}) \in V$ such that $E(p_n, \sigma(\vec{p}))$ holds. A strategy for player 0 is called *memoryless* if it depends only on the current position, i.e.: for every $v \in V_0$, and every $\vec{p}, \vec{q} \in V^*$ we have $\sigma(\vec{p}v) = \sigma(\vec{q}v)$. A play $\vec{p} = p_0p_1 \dots$ *respects* a strategy σ for player 0 if whenever $p_i \in V_0$, then $p_{i+1} = \sigma(p_0p_1 \dots p_i)$. In this case player 0 is said to follow the strategy σ, during the play $\vec{p}$. A strategy σ is winning for player 0 from a position p_0 if all the plays beginning in p_0 and respecting σ are winning. A position p_0 is winning for player 0 if he has a winning strategy from p_0. By $W_0(G)$ we denote the set of winning positions for player 0 in the game G. Similarly we define strategies and the set $W_1(G)$ of the winning positions for player 1.

Let σ be a memoryless strategy for player 0 in an infinite two-player game $G = \langle V, V_0, V_1, E, Acc \rangle$. Strategy σ defines a game where only player 1 plays:

$$G(\sigma) = \langle V', V_0', V_1', E', Acc \rangle$$

with V' being the set of positions where σ is winning, $V'_0 = V_0 \cap V'$, $V'_1 = V_1 \cap V'$ and

$$E' = \Big(E \cap (V_1 \times V_0)\Big) \cup \{(v, v') : v \in V_0, v' = \sigma(v)\}$$

Note that all the plays in $G(\sigma)$ are winning for player 0.

Strategies with memory A *strategy with a memory* M is a tuple $\sigma = \langle \varphi, up, m_0 \rangle$ such that :

$$\varphi : M \times V \to V \qquad up : M \times V \to M \qquad m_0 \in M$$

where m_0 is an initial state of the memory, *up* is a function that updates the memory according to the moves played, and φ is a function giving the next move depending on the memory of the play so far and the current position.

The memory update function is extended to sequences of positions:

$$up^*(m, \epsilon) = m \qquad up^*(m, \vec{p}p_i) = up(up^*(m, \vec{p}), p_i)$$

Thus a strategy with memory σ is defined by :

$$\sigma(\vec{p}p_i) = \varphi(up^*(m_0, \vec{p}p_i), p_i)$$

Winning conditions Throughout the paper, the winning conditions will be Boolean combinations of parity and unboundedness conditions. A *parity winning condition* is given by a coloring function $\Omega : Q \to \{0 \dots d\}$ extended to positions of pushdown games by $\Omega((q, u)) = \Omega(q)$. An infinite path $\vec{v}$ is winning for player 0 if in the sequence $\Omega(\vec{v}) = \Omega(v_0)\Omega(v_1)\dots$ the smallest color appearing infinitely often is even. Formally, we have :

$$Acc_\Omega = \{\vec{p} = (p_0, u_0)(p_1, u_1) \dots \in V^\omega : \liminf_{i \to \infty} \Omega(p_i) \text{ is even}\}$$

The *unboundedness winning condition* is

$$Acc_U = \{\vec{p} = (p_0, u_0)(p_1, u_1) \dots \in V^\omega : \limsup_{i \to \infty} |u_i| = \infty\}$$

It says that there is no bound on the size of the stack during the play.

Conditional pushdown games From a pushdown game G we can define for any subset $R \subseteq Q$ a *conditional game* $G(R)$ where the winning plays for player 0 and player 1 are the same as those in G, except for the plays reaching a configuration of the form (q, ε) (i.e. a configuration with the empty stack). A play reaching (q, ε) is declared winning for player 0 in $G(R)$ iff $q \in R$.

Attractors and traps An *attractor* for player 0 of a set of vertices S, $Attr_0(S)$, is the set of vertices from which player 0 has a strategy to reach a vertex from S. The set $Trap_1(S) = V \setminus Attr_0(V \setminus S)$ is called an S *trap* for player 1. This is the set of vertices where player 1 can prevent player 0 from leaving S. It is well known that from every vertex of $Attr_0(S)$ player 0 has a memoryless strategy to reach S. Similarly, player 1 has a memoryless strategy to keep the play in $Trap_1(S)$

3 Memoryless strategies: parity union unboundedness

In this section we consider games with the winning condition that is a union of unboundedness and parity conditions. In other words player 0 wins if he manages to make the stack arbitrary large or to satisfy the parity condition. We show that for conditions of this type both players have a memoryless winning strategy from their respective winning sets of vertices. Moreover for these conditions unboundedness is equivalent to strict unboundedness, i.e., to the requirement that every configuration is visited only finitely often.

Theorem 1
In a game with "unboundedness union parity" winning conditions, from each position one of the players has a memoryless winning strategy. Moreover, the winning memoryless strategy for player 0 guarantees that no configuration appears infinitely often on the play or the parity condition is satisfied.

Proof
Let U_0 be the set of positions from where player 0 can reach arbitrary large stack size:

$$U_0 = \bigcap_{i\geq 0} Attr_0(Q \times \Gamma^{>i})$$

where $\Gamma^{>i}$ stands for the strings over Γ (i.e. stacks) of size bigger than i. The complement of this set is

$$U_1 = \bigcup_{j\geq 0} \overline{Attr_0(Q \times \Gamma^{>j})} = \bigcup_{j\geq 0} Trap_1(Q \times \Gamma^{\leq j}) = \bigcup_{j\geq 0} A_j$$

Observe that $A_j = Trap_1(Q \times \Gamma^{\leq j})$ is the set of configurations from which player 1 can manage to keep the stack's size bounded by j. By $G|_{A_j}$ we denote the game G restricted to A_j. On $G|_{A_j}$ we have a parity winning condition inherited from G. Let A_j^0 be the set of winning vertices for player 0 in $G|_{A_j}$.

Similarly for A^1_j. Finally, we consider the following set of positions:

$$P_0 = \{v \mid \exists k. \forall j \geq k.\ v \in A^0_j\}$$

Intuitively, the set P_0 consists of positions v where player 0 can win the parity game induced in any A_j (provided j is large enough). We claim that $W_0 = U_0 \cup P_0$ is the set of winning positions for player 0 in G.

Let us first show that positions in U_0 are winning for player 0 and that he has a memoryless strategy for them. For any integer i, player 0 has a memoryless winning strategy σ_i to reach the set $Q \times \Gamma^{>i}$ from any position in U_0. From the sequence $(\sigma_i)_{i \geq 0}$, we show how to extract a memoryless strategy σ_U which is winning on U_0 with respect to the unboundedness condition.

The main idea is the following: each node v has a finite number of successors, hence there must be a successor that corresponds to an infinite number of strategies σ_i. It will be the value of $\sigma_U(v)$. Nevertheless one must take care in constructing such a strategy to keep it being winning in an infinite number of subarenas A_j and therefore the construction must go by induction. For this let us take any linear order over vertices of the game. We proceed by induction on this ordering.

Take the first vertex v_0. It has a finite number of possible successors, so it has a successor w_0 such that $\sigma_i(v_0) = w_0$ for an infinite number of indexes $i \in \mathbb{N}$. Let set I_0 to be this (infinite) subset of indexes, and let $\sigma_U(v_0) = w_0$. We thus have that σ_U equals σ_i on v_0 for all index $i \in I_0$. For the induction step suppose we have defined the strategy for vertices $v_0, \ldots, v_i$ and we have defined the sets $I_0, \ldots, I_i$ at the same time. For the vertex v_{i+1} we consider a successor w_{i+1} such that $\sigma_j(v_{i+1}) = w_{i+1}$ for infinitely many $j \in I_i$. We set $\sigma_U(v_{i+1}) = w_{i+1}$ and set $I_{i+1} = \{j \in I_i : \sigma_j(v_{i+1}) = w_{i+1}\}$. We thus have that for all $k \in I_{i+1}$, strategies σ_U and σ_k are the same on $\{v_0, v_1 \ldots, v_{i+1}\}$.

To show that σ_U is winning, assume by contradiction that there is a play Λ in which the stack's size is bounded by some integer n. Let i be the smallest index such that all the positions with a stack's size smaller than $n+1$ appear before i in our fixed linear ordering. Now, consider some $k \in I_i$, such that $k \geq n+1$ (such a k exists as I_i is infinite). As $\sigma_U(v_j) = \sigma_k(v_j)$ for all $0 \leq j \leq i$, and as all the positions visited in Λ belong to $Q \times \Gamma^{\leq n} \subset \{v_j \mid 0 \leq j < i\}$, it follows that Λ can be seen as a play where player 0 follows her memoryless winning strategy σ_k in the reachability game to $Q \times \Gamma^{>k}$. Therefore, in Λ, a node of stack's size $k \geq n+1$ is eventually reached, a contradiction. This proves that σ_U is a memoryless winning strategy for player 0 from positions in U_0.

Using the same techniques, one defines a memoryless strategy on P_0. For this, one considers a sequence of memoryless strategies $(\sigma_i)_{i \geq 1}$ where σ_i

is a winning strategy in $G|_{A_i}$ from A_i^0. (Memoryless strategies exist as the games considered have parity winning conditions). As for the unboundedness condition, we construct a memoryless strategy σ_P and a decreasing sequence of infinite sets $(I_i)_{i\geq 0}$, such that for every integer k and every $i \in I_k$ we have $\sigma_P(v_j) = \sigma_i(v_j)$ for all $j \leq k$. Then we show that on every loop staying in U_1 and consistent with σ_P the smallest priority is even.

Finally, we define a strategy σ for player 0 on W_0 by setting $\sigma(v) = \sigma_U(v)$ if $v \in U_0$ and $\sigma(v) = \sigma_P(v)$ if $v \in P_0$. It is clear that this strategy is winning on U_0. A play from a vertex in P_0 can either stay in P_0 (and be winning for player 0) or player 1 can decide to move to U_0 but then he will lose too.

We note that $W_1 = (Q \times \Gamma^*) \setminus W_0$ is the set of winning positions for player 1. Indeed, for a node $v \in W_1$ there is some j such that v belongs to A_j^1 . Set A_1^j is a trap for player 0, in which player 1 has a memoryless winning strategy. We can merge all these strategies to obtain one memoryless strategy on W_1. Indeed, consider a sequence $(\varphi_i)_{i\geq 0}$ of memoryless strategies for player 1 where each φ_i is winning in $G|_{A_i}$ from A_i^1. Define a memoryless strategy φ on W_1 by setting $\varphi(v) = \varphi_i(v)$ where i is the smallest index such that $v \in A_i^1$.

Finally, as player 0 has a memoryless winning strategy, one concludes that on all the plays respecting this strategy where the parity condition does not hold each configuration appears only finitely often. This follows from the fact that if a memoryless winning strategy makes a loop and returns to an already visited configuration then the smallest priority between two occurrences of this configuration must be even (otherwise the strategy would be loosing.) $\square$

4 Solving games: Boolean combinations of Büchi and unboundedness

Consider a pushdown game G defined by a pushdown system $\langle Q, \Gamma, \Delta\rangle$ and a partition of states (Q_0, Q_1). We assume here that we have a priority function $\Omega : Q \to \{0, 1\}$. The conditions we are interested in are in one of the four forms:

$$Acc_\Omega \cup Acc_U \quad \overline{Acc_\Omega} \cup Acc_U \quad Acc_\Omega \cap Acc_U \quad \overline{Acc_\Omega} \cap Acc_U \tag{1}$$

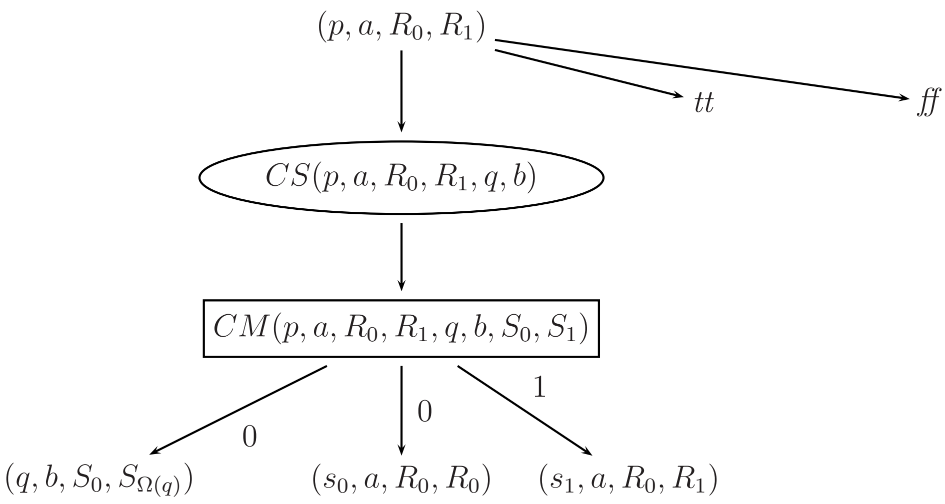
where $\overline{Acc_\Omega}$ stands for the complement of Acc_Ω and similarly for $\overline{Acc_U}$. These four cases cover all possible Boolean combinations of Acc_Ω and Acc_U as, for example, the condition $\overline{Acc_\Omega} \cap \overline{Acc_U} = \overline{Acc_\Omega \cup Acc_U}$ is just the winning condition for player 1 in the game with condition $Acc_\Omega \cup Acc_U$. The main result of the paper is:

Theorem 2
The problem of deciding a winner from a given position in a given pushdown game with conditions as in (1) *is EXPTIME-complete.*

The EXPTIME-hardness follows by the same reduction as for the pushdown games with regular winning conditions (cf. [18, 19]). This is because the EXPTIME lower bound is obtained with reachability conditions. The existence of EXPTIME algorithms for the problem follow from Theorems 30, 42, 43, 3, and 17. Unfortunately, each case requires slightly different reduction. They can be unified but at the expense of substantial complication of the notation. Complication of the notation is also the reason why we restrict to Büchi conditions and do not state the result for all parity conditions.

5 Büchi union unboundedness

From the pushdown game G over priorities $0, 1$ we construct a *finite*-state Büchi game $\widetilde{G}$ with coloring on edges as follows :



The positions of the game are as in the picture for every $p, q \in Q$, $a, b \in G$ and $R_0, R_1, S_0, S_1 \subseteq Q$. From (p, a, R_0, R_1) the edge to tt is when we have $pop(q) \in \Delta(p, a)$ and $q \in R_{\Omega(q)}$. The edge to ff is when we have $pop(q) \in \Delta(p, a)$ and $q \notin R_{\Omega(q)}$. The edge to $CS(p, a, R_0, R_1, q, b)$ is when we have $push(q, b) \in \Delta(p, a)$. From a position $CS(p, a, R_0, R_1, q, b)$ there is an edge to $CM(p, a, R_0, R_1, q, b, S_0, S_1)$ for every $S_0, S_1 \subseteq Q$. From a position $CM(p, a, R_0, R_1, q, b, S_0, S_1)$ there is an edge, called *push edge* to $(q, b, S_0, S_{\Omega(q)})$, and edges to (s_0, a, R_0, R_0) and (s_1, a, R_0, R_1) for every $s_0 \in S_0$ and every $s_1 \in S_1$, respectively.

The positions for player 0 are all the (p, a, R_0, R_1) where $p \in Q_0$, all the positions of the form $CS(\ldots)$ and the position *ff*. The other positions are for player 1.

Theorem 3
For an unboundedness union Büchi winning condition, for every $a \in \Gamma$, $q \in Q$ and $R \subseteq Q$:

$$(q, a) \in W_0(G(R)) \qquad \textit{iff} \qquad (q, a, R, R) \in W_0(\widetilde{G})$$

The proof will follow from Lemmas 9 and 16. We will start with the left to right implication. For it we take a memoryless strategy $\widetilde{\sigma}_1$ for player 1 in $\widetilde{G}$. We assume that $\widetilde{\sigma}_1$ is winning from all the positions winning for player 1. We translate $\widetilde{\sigma}_1$ into a strategy for player 1 in G.

Recall that $\widetilde{G}(\widetilde{\sigma}_1)$ be the game restricted to positions from where $\widetilde{\sigma}_1$ is winning and to edges permitted by $\widetilde{\sigma}_1$.

5.1 Direct implication

Let $\widetilde{\sigma}_1$ be a memoryless strategy for player 1 in $\widetilde{G}$. We assume that $\widetilde{\sigma}_1$ is winning from all the positions winning for player 1. We translate $\widetilde{\sigma}_1$ into a strategy for player 1 in G.

Every path in $\widetilde{G}(\widetilde{\sigma}_1)$ is winning for player 1, hence for every position π in it, there is a finite number of positions of priority 0 that can be reached. Let $h(\pi)$ be this number.

Definition 4 For a position $\pi = (p, a, R_0, R_1)$ in $\widetilde{G}$ we define two sets:

$$\begin{aligned} T_0(\pi) =& \{q : h((q, a, R_0, R_0)) < h(\pi)\} \\ T_1(\pi) =& \{q : h((q, a, R_0, R_1)) \leq h(\pi)\} \end{aligned}$$

Definition 5 We define a strategy σ_1 using $\widetilde{\sigma}_1$ as follows. Suppose that the current position of the play in G is $(q_n, a_n u)$ and the current state of memory is $m = \pi_n \ldots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$ for $i = 1 \ldots n$.

- If the move from π_n to $CS(q_n, a_n, R_0^n, R_1^n, p, c)$ is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then we set
$$up(m, (p, cbu)) = (p, c, S_0, S_{\Omega(p)})\pi_n \ldots \pi_1$$
where $S_0 = Q \setminus T_0(\pi_n)$ and $S_1 = Q \setminus T_1(\pi_n)$. Note that $\Omega(p) \in \{0, 1\}$.

If $q_n \in Q_1$ then additionally we define:

$$\sigma_1(m,(q_n,a_n u)) = (p, ca_n u)$$

- If the move from π_n to *ff* is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then for p such that $p \notin R^n_{\Omega(p)}$ and $pop(p) \in \delta(q_n,a_n)$ we set:

  $$up(m,(p,u)) = (p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2}\dots\pi_1$$

  where $R = R_0^{n-1}$ if $p \in T_0(\pi_{n-1})$, and $R = R_1^{n-1}$ otherwise.

  If $q_n \in Q_1$ then we take p with π'_{n-1} of the first kind if possible, if not then of the second kind and define

  $$\sigma_1(m,(q_n,bu)) = (p,u)$$

For all other cases the update function and the strategy function are not defined.

Definition 6 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$ for $i = 1\dots n$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_1)$ and $R_0^{i+1} = Q \setminus T_0(\pi_i)$ and $R_1^{i+1} = Q \setminus T_0(\pi_i)$ or $R_1^{i+1} = Q \setminus T_1(\pi_i)$ for $i = 1,\dots,n-1$. We say that m is *glued* if $R_0^n = R_1^n$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m. We denote by $tail(m)$ the memory $\pi_{n-1}\dots\pi_1$.

Lemma 7 Suppose that while playing according to strategy σ_1 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$, for $i = 1\dots n$. The next move in $G(\sigma_1)$ is to one of the following positions:

$(p, ca_n \dots a_1)$. The updated memory is $m' = (p, c, S_0, S_{\Omega(p)})\pi_n \dots \pi_1$ and it is consistent. We have that $h(m') < h(m)$. In addition, if $\Omega(p) = 0$ then m' is glued.

$(p, a_{n-1} \dots a_1)$. We have that the updated memory is of the form $m' = (p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2}\dots\pi_1$ and it is consistent. Moreover, if m is glued or $\Omega(p) = 0$, then $h(m') < h(tail(m))$ and m' is glued. Otherwise, we have that $h(m') \le h(tail(m))$.

Proof
Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_1)$ then $push(p, c) \in$

$\delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_1. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_0, S_1) \to^0 (p, c, S_0, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_1)$. Therefore $(p, c, S_0, S_{\Omega(p)})$ is a winning position for player 1. From the definition of the strategy σ_1 we have that $m' = up(m, (p, ca_n \dots a_1)) = (p, c, S_0, S_{\Omega(p)})\pi_n \dots \pi_1$ with $S_0 = Q \setminus T_0(\pi_{n-1})$ and $S_{\Omega(p)} = Q \setminus T_{\Omega(p)}(\pi_{n-1})$. Thus m' is consistent. Moreover, since the edge $CM(\pi_n, p, c, S_0, S_1) \to^0 (p, c, S_0, S_{\Omega(p)})$ is 0-labeled we have $h(m') < h(\pi_n) = h(m)$ and if $\Omega(p) = 0$ then m' is glued by definition.

Consider the second clause. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, a_{n-1} \dots a_1)$ in $G(\sigma_1)$ then $pop(p) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move to *ff* is suggested by the strategy σ_1. In both cases we have by definition of σ_1 that $m' = up(m, (p, a_{n-1} \dots a_1)) = (p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2} \dots \pi_1$ where R is either R_0^{n-1} or R_1^{n-1}, and $p \notin R_{\Omega(p)}^n$. By consistency of m we have that $p \in T_0(\pi_{n-1})$ if $\Omega(p) = 0$, otherwise p belongs either to $T_0(\pi_{n-1})$ or $T_1(\pi_{n-1})$. Therefore $(p, a_{n-1}, R_0^{n-1}, R)$ is a position of $\widetilde{G}(\widetilde{\sigma}_1)$ (since h is defined on it) and $h(m') \leq h(\pi_{n-1}) = h(tail(m))$. By consistency of m we have also that $R_0^{n-1} = Q \setminus T_0(\pi_{n-2})$ and R_1^{n-1} equals either to $Q \setminus T_0(\pi_{n-2})$ or $Q \setminus T_1(\pi_{n-2})$. Thus m' is consistent. Moreover if m is glued or $\Omega(p) = 0$ then $p \in T_0(\pi_{n-1})$. By definition of σ_1 we get $m' = (p, a_{n-1}, R_0^{n-1}, R_0^{n-1})\pi_{n-2} \dots \pi_1$ and it follows that $h(m') < h(\pi_{n-1}) = h(tail(m))$ and m' is glued. $\square$

Lemma 8 Consider two positions $i < j$ such that $|m_i| = |m_j|$ and such that $|m_k| \geq |m_i|$ for all $i \leq k \leq j$. We have that $h(m_i) \geq h(m_j)$. Moreover if a state of priority 0 appears between $i + 1$ and j then $h(m_i) > h(m_j)$ and m_j is glued.

Proof
Let $k_1 = i \leq k_2 \leq \dots \leq k_n = j$ indexes such that $|m_{k_l}| = |m_i|$ for $l = 1 \dots n$. By hypothesis and from Lemma 7 we have for every $l = 2 \dots n$ that $m_{k_l - 1}$ is of the form $\pi m_{k_{(l-1)}}$, for some π. By Lemma 7 we have that $h(m_{k_l}) \leq h(tail(m_{k_l - 1})) = h(m_{k_{(l-1)}})$. By induction we get $h(m_j) \leq h(m_i)$.

To show the second claim, first observe that by easy induction using Lemma 7 we have that if m_k is glued for some $i \leq k \leq j$ then m_j is also glued.

We proceed with the proof of the second claim by induction on the distance between i and j. If $j = i + 2$ and 0 appears between $i + 1$ and $i + 2$ then either $\Omega(q_{i+1}) = 0$ and by Lemma 7 we have $m_{i+1} = \pi m_i$ is glued; or $\Omega(q_j) = 1$. Whatever the case, Lemma 7 provides $h(m_i) > h(m_j)$ and m_j is glued. Suppose now that the distance is bigger. If there is $i < k < j$ with $|m_k| = |m_i|$ then we get the lemma by induction assumption applied to (i, k) and (k, j). If not then we know that all the memories between $i + 1$ and

$j-1$ are of size at least $|m_{i+1}|$. Hence, we can use the induction assumption there. We conclude with the help of Lemma 7. $\square$

Lemma 9 Strategy σ_1 is winning.

Proof
To show that σ_1 is winning consider a play $(q_1, u_1), (q_2, u_2), \ldots$ respecting σ_1 and let $m_1, m_2, \ldots,$ be the associated memories. Using Lemma 7 it follows that for every $i \geq 1$, if there is some $j > i$ such that $|m_i| < |m_j|$ then $h(m_i) < h(m_j)$. As by Lemma 7 the size of the stack at a position is the same as the size of memory at this position we conclude that the size of the stack is bounded; and there is a size of memory that is infinitely repeated. Let N be the smallest size infinitely repeated. Therefore, there is a sequence of positions $i_1, i_2, \ldots$ such that $|m_{i_j}| = N$ for all index i_j and such that $|m_k| \geq N$ for all $k \geq i_1$. From Lemma 8 for every $k \geq 1$, if 0 appears between i_k+1 and $i_{(k+1)}$ then $h(m_{i_k}) > h(m_{i_{(k+1)}})$, and we have $h(m_{i_k}) \geq h(m_{i_{(k+1)}})$ otherwise. Therefore 0 cannot appear infinitely often during the play respecting σ_1.
$\square$

5.2 Reciprocal implication

For the implication in the other direction we take a memoryless winning strategy $\widetilde{\sigma}_0$ for player 0 in $\widetilde{G}$. We assume that the strategy is winning from every winning vertex for player 0. We will construct a winning strategy for player 0 in G.

Let $\widetilde{G}(\widetilde{\sigma}_0)$ be the game restricted to positions from where $\widetilde{\sigma}_0$ is winning and to edges permitted by $\widetilde{\sigma}_0$.

Every path in $\widetilde{G}(\widetilde{\sigma}_0)$ is winning. Hence for a position π there is a finite number of edges of priority 1 that can be reached from π without crossing an edge labeled by 0. Let $h(\pi)$ be this number.

Definition 10 We say that a pair of sets of states (S_0, S_1) is selected for a position $\pi = (p, a, R_0, R_1)$ and $q \in Q$, $b \in \Gamma$, denoted $(S_0, S_1) \in sel(\pi, q, b)$ if there exist $q' \in Q$, $S_1', S_1'' \subseteq Q$ such that in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

$$\pi \to CS(p, a, R_0, R_1, q', b) \to CM(p, a, R_0, R_1, q', b, S_0, S_1') \to (q', b, S_0, S_1'')$$

and (q, b, S_0, S_1) is reachable from (q', b, S_0, S_1'') without passing through a push edge.

Definition 11 We say that a pair of sets of states (S_0, S_1) is 1-selected for a position $\pi = (p, a, R_0, R_1)$ and $q \in Q$, $b \in \Gamma$, denoted $(S_0, S_1) \in sel_1(\pi, q, b)$

if there exist $q' \in Q$, $S'_1 \subseteq Q$ such that $\Omega(q') = 1$ and in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

$$\pi \to CS(p, a, R_0, R_1, q', b) \to CM(p, a, R_0, R_1, q', b, S_0, S'_1) \to (q', b, S_0, S'_1)$$

and (q, b, S_0, S_1) is reachable from (q', b, S_0, S'_1) without passing through a 0-labeled edge.

Definition 12 We define a strategy σ_0 using $\widetilde{\sigma}_0$ as follows. Suppose that the current position in the play in G is $(q_n, a_n \dots a_1)$ and the current memory is $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$ for $i = 1 \dots n$.

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

  $$\pi_n \to CS(q_n, a_n, R_0^n, R_1^n, p, c) \to CM(q_n, a_n, R_0^n, R_1^n, p, c, S_0, S_1)$$

  then we define

  $$up(m, (p, ca_n \dots a_1)) = (p, c, S_0, S_{\Omega(p)})\pi_n \dots \pi_1$$

  Moreover if $q_n \in Q_0$ then we put:

  $$\sigma_0(m, (q_n, a_n \dots a_1)) = (p, ca_n \dots a_1)$$

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is edge $\pi_n \to tt$ then, for every $p \in R^n_{\Omega(p)}$ such that $pop(p) \in \delta(q_n, a_n)$ we have

  $$up(m, (p, a_{n-1} \dots a_1)) = \pi'_{n-1} \dots \pi_1$$

  where we set $\pi'_{n-1} = (p, a_{n-1}, R_0^{n-1}, R_1^{n-1})$ if $\Omega(p) = 1$ and $(R_0^n, R_1^n) \in sel_1(\pi_{n-1}, q_n, a_n)$. Otherwise we put $\pi'_{n-1} = (p, a_{n-1}, R_0^{n-1}, R_0^{n-1})$. Moreover if $q_n \in Q_0$ then we take some $p \in R^n_{\Omega(p)}$ with $\Omega(p) = 0$ if possible, and define:

  $$\sigma_0(m, (q_n, a_n \dots a_1)) = (p, a_{n-1} \dots a_1)$$

For all other cases the update function and the strategy function are undefined.

Definition 13 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$, for $i = 1 \dots n$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_0)$ and for all $i = 1, \dots, n-1$ we have $(R_0^{i+1}, R_1^{i+1}) \in sel(\pi_i, q_{i+1}, a_{i+1})$. We say that m is *proper* if $(R_0^n, R_1^n) \in sel_1(\pi_{n-1}, q_n, a_n)$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m. We denote by $tail(m)$ the memory $\pi_{n-1} \dots \pi_1$.

Lemma 14 Suppose that while playing according to strategy σ_0 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_0^i, R_1^i)$ for $i = 1 \dots n$. The next move in $G(\sigma_0)$ is to one of the following positions:

$(p, ca_n \dots a_1)$. The updated memory is $m' = (p, c, S_0, S_{\Omega(p)})\pi_n \dots \pi_1$ and it is consistent. Moreover if $\Omega(p) = 1$ then m' is proper.

$(p, a_{n-1} \dots a_1)$. We have that the updated memory is of the form $m' = (p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2} \dots \pi_1$, and m' is consistent. Moreover we have $h(m') \leq h(tail(m))$. If m is proper and $\Omega(p) = 1$ then $h(m') < h(tail(m))$, in addition if $tail(m)$ is proper then m' is proper.

Proof

Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_0)$ then $push(p, c) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_0. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_0, S_1) \to (p, c, S_0, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_0)$. Therefore $(p, c, S_0, S_{\Omega(p)})$ is a winning position for player 0 and $(S_0, S_{\Omega(p)}) \in sel(\pi_n, p, c)$. In addition, if $\Omega(p) = 1$ then $(S_0, S_{\Omega(p)}) \in sel_1(\pi_n, p, c)$. From the definition of the strategy σ_0 we have that $m' = up(m, (p, ca_n \dots a_1)) = (p, c, S_0, S_{\Omega(p)})\pi_n \dots \pi_1$. Thus m' is consistent. In addition, if $\Omega(p) = 1$ then m' is proper.

Consider the second clause. As m is consistent, the position π_{n-1} is winning, and $(R_0^n, R_1^n) \in sel(\pi_{n-1}, q_n, a_n)$, that is, there is in $\widetilde{G}(\widetilde{s}_0)$ some path $\pi_{n-1} \to CS(\pi_{n-1}, q, a_n) \to CM(\pi_{n-1}, q, a_n, S_0, S_1) \to (q, a_n, S_0, S_1')$ and a path from (q, a_n, S_0, S_1') to (q_n, a_n, R_0^n, R_1^n) without passing through a push edge. By definition of $\widetilde{G}$ we get that $R_0^n = S_0$ and R_1^n equals either S_1 (if m is proper, by definition of sel_1) or S_0. By definition of σ_0 we have $m' = up(m, (p, a_{n-1} \dots a_1)) = (p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2} \dots \pi_1$, and $p \in R_{\Omega(p)}^n$. Thus p belongs either to R_0^n or S_1 and we can deduce that $(p, a_{n-1}, R_0^{n-1}, R)$ is a successor of $CM(\pi_{n-1}, q_n, a_n, R_0^n, S_1)$. Therefore $(p, a_{n-1}, R_0^{n-1}, R)$ is winning for player 0 and since by consistency of m we also have that $(R_0^{n-1}, R_1^{n-1}) \in sel(\pi_{n-2}, q_{n-1}, a_{n-1})$ we deduce that $(R_0^{n-1}, R) \in sel(\pi_{n-2}, q_{n-1}, a_{n-1})$. Thus m' is consistent. Moreover, we deduce that $h(m') \leq h(\pi_{n-1}) = h(tail(m))$. If m is proper and $\Omega(p) = 1$ then $p \in R_1^n = S_1$, $R = R_1^n$, and it follows that the edge $CM(\pi_{n-1}, q_n, a_n, S_0, S_1) \to (p, a_{n-1}, R_1^{n-1}, R_1^{n-1})$ is labeled by 1, thus we have $h(m') < h(\pi_{n-1}) = h(tail(m))$. In addition if $tail(m)$ is proper then by definition of $sel_1(\pi_{n-2}, q_{n-1}, a_{n-1})$ and since $(p, a_{n-1}, R_0^{n-1}, R_1^{n-1})$ is a successor (through a 1-labeled edge) of $CM(\pi_{n-1}, q_n, a_n, R_0^n, R_1^n)$ we have that m' is proper.

□

Lemma 15 Consider two positions $i < j$ such that $|m_i| = |m_j|$ and such that $|m_k| \geq |m_i|$ for all $i \leq k \leq j$. We have that $h(m_i) \geq h(m_j)$. If only states of priority 1 appear between i and j then we have that $h(m_i) > h(m_j)$.

Proof

Let $k_1 = i \leq k_2 \leq \ldots \leq k_n = j$ indexes such that $|m_{k_l}| = |m_i|$ for $l = 1 \ldots n$. By hypothesis and from Lemma 14 we have for every $l = 2 \ldots n$ that $m_{k_l - 1}$ is of the form $\pi m_{k_{(l-1)}}$, for some π. From Lemma 14, we deduce that $h(m_i) \leq h(m_j)$.

If there is only states of priority 1 between i and j then using Lemma 14 we have for every $i < k < j$ that m_k is proper. Therefore, for every $l = 2 \ldots n$ we have that $m_{k_l - 1}$ is proper and $\Omega(q_{k_l}) = 1$, thus by Lemma 14 we get $h(m_{k_l}) < h(tail(m_{k_l - 1})) = h(m_{k_{(l-1)}})$. By induction it follows that $h(m_j) < h(m_i)$. □

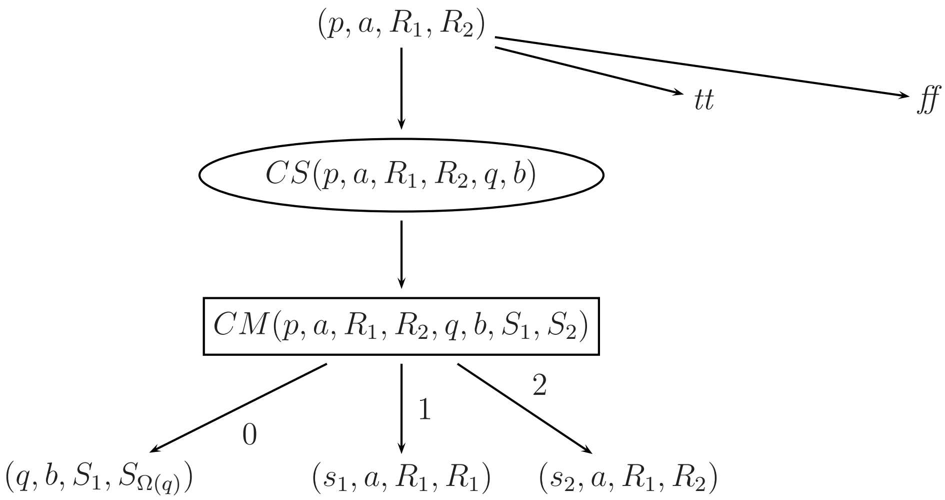
Lemma 16 The strategy σ_0 is winning.

Proof
To show that σ_0 is winning, consider a play $(q_1, u_1), (q_2, u_2), \ldots$ respecting σ_0 and let $m_1, m_2, \ldots,$ be the associated memories.

Assume that the stack is bounded. As, by Lemma 14, the size of memory at a position is the same as the size of stack at this position, we have that there is a size of memory that is infinitely repeated. Let N be the smallest size infinitely repeated. Therefore, there is a sequence of positions $i_1, i_2, \ldots$ such that $|m_{i_j}| = N$ for all index i_j and such that $|m_k| \geq N$ for all $k \geq i_1$. By contradiction assume that only states of priority 1 appear infinitely often. Therefore, there is some k such that after i_k only states colored by 1 are visited. By Lemma 15, we have that $h(m_{i_j}) > h(m_{i_{(j+1)}})$ for all $j \geq k$. As $<$ is well-founded, it is not possible. Therefore the Büchi condition is satisfied. □

6 co-Büchi union unboundedness

From the pushdown game G over priorities $1, 2$ we construct a *finite*-state parity game $\widetilde{G}$ with coloring on edges as follows :



The positions of the game are as in the picture for every $p, q \in Q$, $a, b \in G$ and $R_1, R_2, S_1, S_2 \subseteq Q$. From (p, a, R_1, R_2) the edge to tt is when we have $pop(q) \in \Delta(p, a)$ and $q \in R_{\Omega(q)}$. The edge to $f\!f$ is when we have $pop(q) \in \Delta(p, a)$ and $q \notin R_{\Omega(q)}$. The edge to $CS(p, a, R_1, R_2, q, b)$ is when we have $push(q, b) \in \Delta(p, a)$. From a position $CS(p, a, R_1, R_2, q, b)$ there is an edge to $CM(p, a, R_1, R_2, q, b, S_1, S_2)$ for every $S_1, S_2 \subseteq Q$. From a position $CM(p, a, R_1, R_2, q, b, S_1, S_2)$ there is an edge, called *push edge* to $(q, b, S_1, S_{\Omega(q)})$, and edges to (s_1, a, R_1, R_1) and (s_2, a, R_1, R_2) for every $s_1 \in S_1$ and every $s_2 \in S_2$, respectively.

The positions for player 0 are all the (p, a, R_1, R_2) where $p \in Q_0$, all the positions of the form $CS(\dots)$ and the position $f\!f$. The other positions are for player 1.

Theorem 17
For an unboundedness union co-Büchi winning condition, for every $a \in \Gamma$, $q \in Q$ and $R \subseteq Q$:

$$(q, a) \in W_0(G(R)) \qquad \textit{iff} \qquad (q, a, R, R) \in W_0(\widetilde{G})$$

The proof will follow from Lemmas 23 and 29. We will start with the left to right implication.

Recall that $\widetilde{G}(\widetilde{\sigma})$ is the game restricted to positions from where $\widetilde{\sigma}$ is winning and to edges permitted by $\widetilde{\sigma}$.

6.1 Direct implication

We take a memoryless strategy $\widetilde{\sigma}_1$ for player 1 in $\widetilde{G}$. We assume that $\widetilde{\sigma}_1$ is winning from all the positions winning for player 1. We translate $\widetilde{\sigma}_1$ into a strategy for player 1 in G.

Every path in $\widetilde{G}(\widetilde{\sigma}_1)$ is winning for player 1, hence for every position π in it, there is a finite number of positions of priority 0 that can be reached. Let $\tau_0(\pi)$ be this number. There is also a finite number of positions of priority 2 that can be reached before going through a position of priority 1 or 0. Denote by $\tau_2(\pi)$ this number. Let the height of a position π be a pair $h(\pi) = (\tau_0(\pi), \tau_2(\pi))$. We will compare heights using lexicographic ordering $\preccurlyeq$ on pairs. We also use the relation $(\tau_0, \tau_2) \preccurlyeq_0 (\tau_0', \tau_2')$ which holds if $\tau_0 \leq \tau_0'$.

Definition 18 For a position $\pi = (p, a, R_1, R_2)$ in $\widetilde{G}$ we define two sets:

$$\begin{aligned} T_1(\pi) = &\{s : \exists q \in Q, b \in \Gamma, S_1, S_2 \subseteq Q. \\ &\pi \to CS(\pi, q, b) \to CM(\pi, q, b, S_1, S_2) \to^1 (s, a, R_1, R_1) \\ &\text{is a path in} \widetilde{G}(\widetilde{\sigma}_1)\} \\ T_2(\pi) = &\{s : \exists q \in Q, b \in \Gamma, S_1, S_2 \subseteq Q. \\ &\pi \to CS(\pi, q, b) \to CM(\pi, q, b, S_1, S_2) \to^2 (s, a, R_1, R_2) \\ &\text{is a path in} \widetilde{G}(\widetilde{\sigma}_1)\} \end{aligned}$$

Definition 19 We define a strategy σ_1 using $\widetilde{\sigma}_1$ as follows. Suppose that the current position of the play in G is $(q_n, a_n u)$ and the current state of memory is $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$ for $i = 1 \dots n$.

- If the move from π_n to $CS(q_n, a_n, R_1^n, R_2^n, p, c)$ is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then we set
$$up(m, (p, cbu)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$$
where $S_1 = Q \setminus T_1(\pi_n)$ and $S_2 = Q \setminus T_2(\pi_n)$. Note that $\Omega(p) \in \{1, 2\}$.

  If $q_n \in Q_1$ then additionally we define:
$$\sigma_1(m, (q_n, a_n u)) = (p, c a_n u)$$

- If the move from π_n to *ff* is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then for p such that $p \notin R^n_{\Omega(p)}$ and $pop(p) \in \delta(q_n, a_n)$ we set:
$$up(m, (p, u)) = \pi'_{n-1}\pi_{n-2} \dots \pi_1$$
where $\pi'_{n-1} = (p, a_{n-1}, R_1^{n-1}, R_2^{n-1})$ if $R_2^n = Q \setminus T_2(\pi_{n-1})$ and $\Omega(p) = 2$. Otherwise we have $\pi'_{n-1} = (p, a_{n-1}, R_1^{n-1}, R_1^{n-1})$.

  If $q_n \in Q_1$ then we take p with π'_{n-1} of the first kind if possible, if not then of the second kind and define:
$$\sigma_1(m, (q_n, bu)) = (p, u)$$

For all other cases the update function and the strategy function are not defined.

Definition 20 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_1)$ and $R_1^{i+1} = Q \backslash T_1(\pi_i)$ and $R_2^{i+1} = Q \backslash T_1(\pi_i)$ or $R_2^{i+1} = Q \backslash T_2(\pi_i)$ for all $i = 1, \dots, n-1$. We say that m is *proper* if $R_2^n = Q \setminus T_2(\pi_{n-1})$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m. We denote by $tail(m)$ the memory $\pi_{n-1} \dots \pi_1$.

Lemma 21 Suppose that while playing according to strategy σ_1 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$. The next move in $G(\sigma_1)$ is to one of the following positions:

$(p, ca_n \dots a_1)$**.** The updated memory is $m' = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$ and it is consistent. We have that $h(m') \prec_0 h(m)$. In addition, if $\Omega(p) = 2$, then m' is proper.

$(p, a_{n-1} \dots a_1)$**.** We have that the updated memory is of the form $m' = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$ and it is consistent. Also, we have that $h(m') \preccurlyeq_0 h(\pi_{n-1})$. Moreover, if m is proper and $\Omega(p) = 2$, then $h(m') \prec h(\pi_{n-1})$. In addition, if $tail(m)$ is proper then m' is proper.

Proof
Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_1)$ then $push(p, c) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_1. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_1, S_2) \to^0 (p, c, S_1, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_1)$. Therefore $(p, c, S_1, S_{\Omega(p)})$ is a winning position for player 1. From the definition of the strategy σ_1 we have that $m' = up(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$ with $S_1 = Q \setminus T_1(\pi_{n-1})$ and $S_{\Omega(p)} = Q \setminus T_{\Omega(p)}(\pi_{n-1})$. Thus m' is consistent. Moreover, since the edge $CM(\pi_n, p, c, S_1, S_2) \to^0 (p, c, S_1, S_{\Omega(p)})$ is 0-labeled we have $h(m') \prec_0 h(\pi_n) = h(m)$ and if $\Omega(p) = 2$ then m' is proper by definition.

Consider the second clause. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, a_{n-1} \dots a_1)$ in $G(\sigma_1)$ then $pop(p) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move to *ff* is suggested by the strategy σ_1. In both cases we have by definition of σ_1 that $m' = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$ where R is either R_1^{n-1} or R_2^{n-1}, and $p \notin R_{\Omega(p)}^n$. By consistency of m we have that $p \in T_1(\pi_{n-1})$ if $\Omega(p) = 1$, otherwise p belongs either to $T_1(\pi_{n-1})$ or $T_2(\pi_{n-1})$. Therefore $(p, a_{n-1}, R_1^{n-1}, R)$ is a position of $\widetilde{G}(\widetilde{\sigma}_1)$ and $h(m') \preccurlyeq_0 h(\pi_{n-1}) = h(tail(m))$. By consistency of m we have also that $R_1^{n-1} = Q \setminus T_1(\pi_{n-2})$, and R_2^{n-1} equals either to $Q \setminus T_1(\pi_{n-2})$ or $Q \setminus T_2(\pi_{n-2})$. Thus m' is consistent. Moreover if m is proper and $\Omega(p) = 2$ then $p \in T_2(\pi_{n-1})$ and by definition of σ_1 we get $m' = (p, a_{n-1}, R_1^{n-1}, R_2^{n-1})\pi_{n-2} \dots \pi_1$ and it follows that $h(m') \prec h(\pi_{n-1}) = h(tail(m))$. If additionaly $tail(m)$ is proper then $R_2^{n-1} = Q \setminus T_2(\pi_{n-2})$, hence m' is proper. $\square$

Lemma 22 Consider two positions $i < j$ such that $|m_i| = |m_j|$ and such that $|m_k| \geq |m_i|$ for all $i \leq k \leq j$. If only states of priority 2 appear between i and j then we have that $h(m_j) \prec h(m_i)$.

Proof
Let $k_1 = i \leq k_2 \leq \dots \leq k_n = j$ indexes such that $|m_{k_l}| = |m_i|$ for $l = 1 \dots n$. By hypothesis and from Lemma 21 we have for every $l = 2 \dots n$ that $m_{k_l - 1}$ is of the form $\pi m_{k_{(l-1)}}$, for some π.

If there are only states of priority 2 between i and j then using Lemma 21 we have for every $i < k < j$ that m_k is proper. Therefore, for every $l = 2 \dots n$ we have that $m_{k_l - 1}$ is proper and $\Omega(q_{k_l}) = 2$, thus by Lemma 21 we get $h(m_{k_l}) \prec h(tail(m_{k_l - 1})) = h(m_{k_{(l-1)}})$. By induction it follows that $h(m_j) \prec h(m_i)$. $\square$

Lemma 23 Strategy σ_1 is winning.

Proof

To show that σ_1 is winning consider a play $(q_1, u_1), (q_2, u_2), \dots$ respecting σ_1 and let $m_1, m_2, \dots,$ be the associated memories. Using Lemma 21 it follows that for every $i \geq 1$, if there is some $j > i$ such that $|m_i| < |m_j|$ then $h(m_i) \prec_0 h(m_j)$. As by Lemma 21 the size of the stack at a position is the same as the size of memory at this position we conclude that the size of the stack is bounded; and there is a size of memory that is infinitely repeated. Let N be the smallest size infinitely repeated. Therefore, there is a sequence of positions $i_1, i_2, \dots$ such that $|m_{i_j}| = N$ for all index i_j and such that $|m_k| \geq N$ for all $k \geq i_1$.

Now, let us show that a state of priority 1 appears infinitely often on the play. If not, after some position i_k, only states of priority 2 are visited. Using Lemma 22 it follows that $h(m_{i_{j+1}}) \prec h(m_{i_j})$ for all $j \geq k$. But, as $\preccurlyeq$ is well founded, it leads to a contradiction. Hence, an infinite number of states of priority 1 are visited. $\square$

6.2 Reciprocal implication

For the implication in the other direction we take a memoryless winning strategy $\widetilde{\sigma}_0$ for player 0 in $\widetilde{G}$. We assume that the strategy is winning from every winning vertex for player 0. We will construct a winning strategy for player 0 in G.

Every path in $\widetilde{G}(\widetilde{\sigma}_0)$ is winning. Hence for a position π there is a finite number of edges of priority 1 that can be reached from π without crossing an edge labeled by 0. Let $h(\pi)$ be this number.

Definition 24 We say that a pair of sets of states (S_1, S_2) is selected for a position $\pi = (p, a, R_1, R_2)$ and $q \in Q$, $b \in \Gamma$, denoted $(S_1, S_2) \in sel(\pi, q, b)$ if there exist $q' \in Q$, $S_2', S_2'' \subseteq Q$ such that in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

$$\pi \to CS(p, a, R_1, R_2, q', b) \to CM(p, a, R_1, R_2, q', b, S_1, S_2') \to (q', b, S_1, S_2'')$$

and (q, b, S_1, S_2) is reachable from (q', b, S_1, S_2'') without passing through a push edge.

Definition 25 We define a strategy σ_0 using $\widetilde{\sigma}_0$ as follows. Suppose that the current position in the play in G is $(q_n, a_n \dots a_1)$ and the current memory is $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$ for $i = 1 \dots n$.

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

  $$\pi_n \to CS(q_n, a_n, R_1^n, R_2^n, p, c) \to CM(q_n, a_n, R_1^n, R_2^n, p, c, S_1, S_2)$$

  then we define

  $$up(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$$

  Moreover if $q_n \in Q_0$ then we put:

  $$\sigma_0(m, (q_n, a_n \dots a_1)) = (p, ca_n \dots a_1)$$

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is edge $\pi_n \to tt$ then, for every $p \in R^n_{\Omega(p)}$ such that $pop(p) \in \delta(q_n, a_n)$ we have

  $$up(m, (p, a_{n-1} \dots a_1)) = \pi'_{n-1} \dots \pi_1$$

  where $\pi'_{n-1} = (p, a_{n-1}, R_1^{n-1}, R_1^{n-1})$ if $\Omega(p) = 1$ or $R_2^n = R_1^n$. Otherwise we put $\pi'_{n-1} = (p, a_{n-1}, R_1^{n-1}, R_2^{n-1})$.

  Moreover if $q_n \in Q_0$ then we take some $p \in R^n_{\Omega(p)}$ of the first kind if possible, if not of the second kind, and define:

  $$\sigma_0(m, (q_n, a_n \dots a_1)) = (p, a_{n-1} \dots a_1)$$

For all other cases the update function and the strategy function are undefined.

Definition 26 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_0)$ and for all $i = 1, \dots, n-1$ we have $(R_1^{i+1}, R_2^{i+1}) \in \mathit{sel}(\pi_i, q_{i+1}, a_{i+1})$. We say that m is *glued* if $R_1^n = R_2^n$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m. We denote by $\mathit{tail}(m)$ the memory $\pi_{n-1} \dots \pi_1$.

Lemma 27 Suppose that while playing according to strategy σ_0 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$, for $i = 1 \dots n$. The next move in $G(\sigma_1)$ is to one of the following positions:

$(p, ca_n \dots a_1)$. The updated memory is $m' = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$ and it is consistent.

$(p, a_{n-1} \dots a_1)$. We have that the updated memory is of the form $m' = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$ with R equals either R_1^{n-1} or R_2^{n-1}, and it is consistent. Moreover $h(m') \leq h(\mathit{tail}(m))$ and $h(m') < h(\mathit{tail}(m))$ if m is glued or $\Omega(p) = 1$.

Proof

Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_0)$ then $\mathit{push}(p, c) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_0. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_1, S_2) \to (p, c, S_1, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_0)$. Therefore $(p, c, S_1, S_{\Omega(p)})$ is a winning position for player 0 and $(S_1, S_{\Omega(p)}) \in \mathit{sel}(\pi_n, p, c)$. From the definition of the strategy σ_0 we have that $m' = \mathit{up}(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$. Thus m' is consistent.

Consider the second clause. As m is consistent, the position π_{n-1} is winning, and $(R_1^n, R_2^n) \in \mathit{sel}(\pi_{n-1}, q_n, a_n)$, that is, there is in $\widetilde{G}(\widetilde{\sigma}_0)$ some path $\pi_{n-1} \to CS(\pi_{n-1}, q, a_n) \to CM(\pi_{n-1}, q, a_n, S_1, S_2) \to (q, a_n, S_1, S_2')$ and a path from (q, a_n, S_1, S_2') to (q_n, a_n, R_1^n, R_2^n) without passing through a push edge. By definition of $\widetilde{G}$ we get that $R_1^n = S_1$ and R_2^n equals either S_2 or S_1. By definition of σ_0 we have $p \in R_{\Omega(p)}^n$. Thus p belongs either to R_1^n or S_2. Hence $(p, a_{n-1}, R_1^{n-1}, R)$ is a successor of $CM(\pi_{n-1}, q_n, a_n, R_1^n, S_2)$. Therefore $(p, a_{n-1}, R_1^{n-1}, R)$ is winning for player 0. Since by consistency of m we have that $(R_1^{n-1}, R_2^{n-1}) \in \mathit{sel}(\pi_{n-2}, q_{n-1}, a_{n-1})$, we deduce that $(R_1^{n-1}, R) \in$

$sel(\pi_{n-2}, q_{n-1}, a_{n-1})$. Thus m' is consistent. Moreover, we deduce that $h(m') \leq h(\pi_{n-1}) = h(tail(m))$ and if m is glued or $\Omega(p) = 1$ then $p \in S_1 = R_1^n$ and it follows that the edge $CM(\pi_{n-1}, q_n, a_n, S_1, S_2) \to (p, a_{n-1}, R_1^{n-1}, R_1^{n-1})$ is labeled by 1, and we have $h(m') < h(\pi_{n-1}) = h(tail(m))$. □

Lemma 28 Consider two positions $i < j$ such that $|m_i| = |m_j|$ and such that $|m_k| \geq |m_i|$ for all $i \leq k \leq j$. We have that $h(m_i) \geq h(m_j)$. Moreover if a state of priority 1 appears between $i+1$ and j then $h(m_i) > h(m_j)$ and m_j is glued.

Proof

Let $k_1 = i \leq k_2 \leq \ldots \leq k_n = j$ indexes such that $|m_{k_l}| = |m_i|$ for $l = 1 \ldots n$. By hypothesis and from Lemma 27 we have for every $l = 2 \ldots n$ that $m_{k_l - 1}$ is of the form $\pi m_{k_{(l-1)}}$, for some π. From Lemma 27, we deduce that $h(m_i) \leq h(m_j)$.

To show the second claim, first observe that by easy induction using Lemma 27 we have that if m_k is glued for some $i < k < j$ then m_j is also glued.

We proceed with the proof of the second claim by induction on the distance between i and j. If $j = i+2$ and 1 appears between $i+1$ and $i+2$ then either $\Omega(q_{i+1}) = 1$ and by Lemma 27 we have $m_{i+1} = \pi m_i$ is glued; or $\Omega(q_j) = 1$. Whatever the case, Lemma 27 provides $h(m_i) > h(m_j)$ and m_j is glued. Suppose now that the distance is bigger. If there is $i < k < j$ with $|m_k| = |m_i|$ then we get the lemma by induction assumption applied to (i, k) and (k, j). If not then we know that all the memories between $i+1$ and $j-1$ are of size at least $|m_{i+1}|$. Hence, we can use the induction assumption there. We conclude with the help of Lemma 27. □

Lemma 29 The strategy σ_0 is winning.

Proof

To show that σ_1 is winning consider, a play $(q_1, u_1), (q_2, u_2), \ldots$ respecting σ_0 and let $m_1, m_2, \ldots,$ be the associated memories.

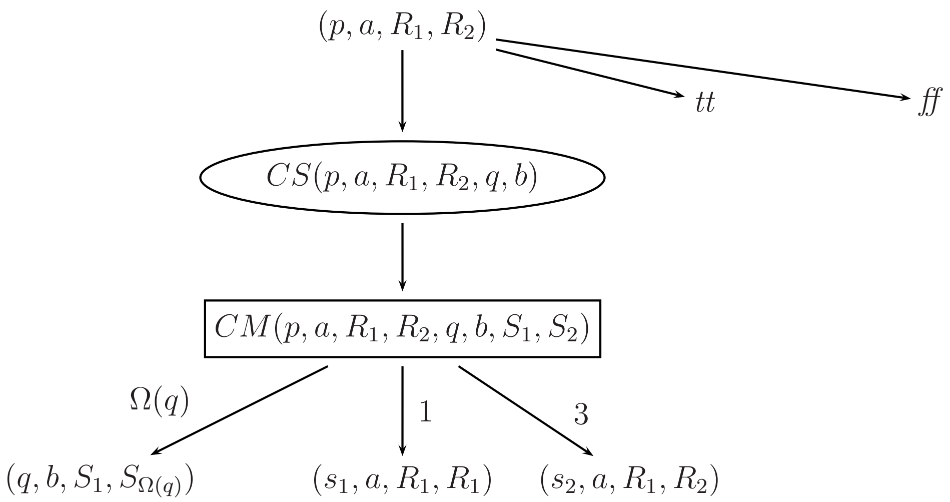
Assume that the stack is bounded. As, by Lemma 27, the size of memory at a position is the same as the size of stack at this position, we have that there is a size of memory that is infinitely repeated. Let N be the smallest size infinitely repeated. Therefore, there is a sequence of positions $i_1, i_2, \ldots$ such that $|m_{i_j}| = N$ for all index i_j and such that $|m_k| \geq N$ for all $k \geq i_1$. By Lemmas 28 we have that $h(m_{i_j}) \leq h(m_{i_{j+1}})$ and the inequality is strict if there is a state of priority 1 between $i_j + 1$ and i_{j+1}. As $<$ is well-founded, it is not possible that there are infinitely many states of priority 1 in the sequence. Therefore the co-Büchi condition is satisfied. □

7 co-Büchi and unboundedness

In this section we consider pushdown games with the condition: "red states appear only finitely often and the stack is unbounded". Formally, we have a priority function Ω assigning to the states of the game priorities 1 or 2. The set of winning paths is:

$$Acc(\Omega \wedge U) = Acc_{\Omega} \cap Acc_{U}$$

Fix a pushdown game G defined by a pushdown system $\langle Q, \Gamma, \Delta \rangle$, a partition of states (Q_0, Q_1) and a priority function $\Omega : Q \to \{1, 2\}$. We will reduce the problem of solving this game to that of solving a finite parity game $\widetilde{G}$. The game $\widetilde{G}$ is presented below. For clarity it has coloring attached to edges not to vertices.



The positions of the game are as in the picture for every $p, q \in Q$, $a, b \in \Gamma$ and $R_1, R_2, S_1, S_2 \subseteq Q$. From (p, a, R_1, R_2) the edge to tt is when we have $pop(q) \in \Delta(p, a)$ and $q \in R_{\Omega(q)}$. The edge to ff is when we have $pop(q) \in \Delta(p, a)$ and $q \notin R_{\Omega(q)}$. The edge to $CS(p, a, R_1, R_2, q, b)$ is when we have $push(q, b) \in \Delta(p, a)$. From a position $CS(p, a, R_1, R_2, q, b)$ there is an edge to $CM(p, a, R_1, R_2, q, b, S_1, S_2)$ for every $S_1, S_2 \subseteq Q$. From a position $CM(p, a, R_1, R_2, q, b, S_1, S_2)$ there is an edge, called *push edge* to $(q, b, S_1, S_{\Omega(q)})$, and edges to (s_1, a, R_1, R_1) and (s_2, a, R_1, R_2) for every $s_1 \in S_1$ and every $s_2 \in S_2$, respectively.

The positions for player 0 are all the (p, a, R_1, R_2) where $p \in Q_0$, all the positions of the form $CS(\ldots)$ and the position ff. The other positions are for player 1.

The following theorem shows the correspondence between G and $\widetilde{G}$.

Theorem 30
For every $a \in \Gamma$, $q \in Q$ and $R \subseteq Q$:

$$(q, a) \in W_0(G(R)) \qquad \textit{iff} \qquad (q, a, R, R) \in W_0(\widetilde{G})$$

The proof will follow from Lemmas 35 and 41.

We will start with the left to right implication.

7.1 Direct

We take a memoryless strategy $\widetilde{\sigma}_1$ for player 1 in $\widetilde{G}$. We assume that $\widetilde{\sigma}_1$ is winning from all the positions winning for player 1. We translate $\widetilde{\sigma}_1$ into a strategy σ_1 for player 1 in G.

Recall that $\widetilde{G}(\widetilde{\sigma}_1)$ is the game restricted to positions from where $\widetilde{\sigma}_1$ is winning and to edges permitted by $\widetilde{\sigma}_1$. Let $\widetilde{E}$ be the set of edges of $\widetilde{G}(\widetilde{\sigma}_1)$.

Every path in $\widetilde{G}(\widetilde{\sigma}_1)$ is winning for player 1, hence for every position π in it there is a finite number of edges of priority 2 that can be reached without going through an edge of priority 1. Let $h(\pi)$ be this number.

Definition 31 For a position $\pi = (p, a, R_1, R_2)$ in $\widetilde{G}$ we define two sets:

$$\begin{aligned} T_1(\pi) = \{ & s : \exists q \in Q, b \in \Gamma, S_1, S_2 \subseteq Q. \\ & \pi \to CS(\pi, q, b) \to (CM(\pi, q, b, S_1, S_2) \to^1 (s, a, R_1, R_1)) \\ & \text{is a path in } \widetilde{G}(\widetilde{\sigma}_1)\} \\ T_2(\pi) = \{ & s : \exists q \in Q, b \in \Gamma, S_1, S_2 \subseteq Q. \\ & \pi \to CS(\pi, q, b) \to (CM(\pi, q, b, S_1, S_2) \to^3 (s, a, R_1, R_2)) \\ & \text{is a path in } \widetilde{G}(\widetilde{\sigma}_1)\} \end{aligned}$$

We denote by *Residues* the set $Q \times \Gamma \times 2^Q \times 2^Q$. Using $\widetilde{\sigma}_1$, we construct a strategy with memory $\sigma_1 = \langle \varphi, up, m_0 \rangle$ for player 1 in $G(R)$. The memory's domain will be $M = (\mathit{Residues})^*$ and the update function will use this memory as a stack. At the beginning of the play the initial state of the memory is $m_0 = (q, a, R, R)$.

Definition 32 We define a strategy $\sigma_1 = \langle \varphi, up, m_0 = (q, a, R, R) \rangle$ for player 1 in $G(R)$ from (q, a). Suppose that the current position of the play in G

is $(q_n, a_n u)$ and the current state of memory is $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$, for $i = 1 \dots n$.

- If the move from π_n to $CS(q_n, a_n, R_1^n, R_2^n, p, c)$ is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then we set
$$up(m, (p, ca_n u)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$$
where $S_1 = Q \setminus T_1(\pi_n)$ and $S_2 = Q \setminus T_2(\pi_n)$. Note that $\Omega(p) \in \{1, 2\}$.
If $q_n \in Q_1$ then additionally we define:
$$\varphi(m, (q_n, a_n u)) = (p, ca_n u)$$

- If the move from π_n to *ff* is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ and if $n > 1$ then for p such that $p \notin R^n_{\Omega(p)}$ and $pop(p) \in \Delta(q_n, a_n)$ we set:
$$up(m, (p, u)) = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$$
where $R = R_2^{n-1}$ if $p \in T_2(\pi_{n-1})$, and $R = R_1^{n-1}$ otherwise.
If $q_n \in Q_1$ then we take some arbitrary $p \notin R^n_{\Omega(p)}$ and define :
$$\varphi(m, (q_n, a_n u)) = (p, u)$$

For all other cases the update function and the strategy function are not defined.

It is not easy to give the intuitions behind this definition. Basically the strategy σ_1 follows the strategy $\widetilde{\sigma}_1$. The problem is that in $\widetilde{G}$ we do not have a stack so when in G the play makes a pop, in $\widetilde{G}$ we expect to arrive at the state *ff* which is winning for player 1. In this case we need to know that we can continue (and win) in G. The role of the sets $T_1(\pi)$ and $T_2(\pi)$ is precisely to guarantee this.

Definition 33 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$, for $i = 1 \dots n$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_1)$ and $R_1^{i+1} = Q \setminus T_1(\pi_i)$ and $R_2^{i+1} = Q \setminus T_2(\pi_i)$ or $R_2^{i+1} = Q \setminus T_1(\pi_i)$ for all $i = 1, \dots, n-1$. We say that m is *proper* if $R_2^n = Q \setminus T_2(\pi_{n-1})$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m.

The following lemma abstracts important properties of the strategy σ_1.

Lemma 34 Suppose that while playing according to strategy σ_1 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$ for $i = 1, \dots, n$. The next move in $G(\sigma_1)$ is to one of the following positions:

$(p, ca_n \dots a_1)$**.** We have that the updated memory is $m' = (p, c, S_1, S_2)\pi_n \dots \pi_1$ and it is consistent. Moreover if $\Omega(p) = 2$ then $h(m') < h(m)$ and m' is proper.

$(p, a_{n-1} \dots a_1)$**.** We have that the updated memory is of the form $m' = (p, a_{n-1}, S_1, S_2)\pi_{n-2} \dots \pi_1$ and it is consistent. In the case when m is proper and $\Omega(p) = 2$ we have $h(m') \leq h(\pi_{n-1})$ and if additionaly $tail(m)$ is proper then m' is proper.

Proof

Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_1)$ then $push(p, c) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_1. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_1, S_2) \to^0 (p, c, S_1, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_1)$. Therefore $(p, c, S_1, S_{\Omega(p)})$ is a winning position for player 1. From the definition of the strategy σ_1 we have that $m' = up(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$ with $S_1 = Q \setminus T_1(\pi_{n-1})$ and $S_2 = Q \setminus T_2(\pi_{n-1})$. Thus m' is consistent. Moreover, if $\Omega(p) = 2$ then m' is proper by definition. In this case there is also a path from π_n to (p, c, S_1, S_2) in $\widetilde{G}(\widetilde{\sigma}_1)$ with an edge labeled by 2 and no edge labeled by 1. Hence $h(m') = h(p, c, S_1, S_2) < h(\pi_n) = h(m)$.

Consider the second clause. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, a_{n-1} \dots a_1)$ in $G(\sigma_1)$ then $pop(p) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move to *ff* is suggested by the strategy σ_1. In both cases we have by definition of σ_1 that $m' = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$ where R is either R_1^{n-1} or R_2^{n-1}, and $p \notin R_{\Omega(p)}^n$. By consistency of m we have that $p \in T_1(\pi_{n-1})$ if $\Omega(p) = 1$, otherwise p belongs either to $T_1(\pi_{n-1})$ or $T_2(\pi_{n-1})$. Therefore $(p, a_{n-1}, R_1^{n-1}, R)$ is a position of $\widetilde{G}(\widetilde{\sigma}_1)$. By consistency of m we have also that $R_1^{n-1} = Q \setminus T_1(\pi_{n-2})$, and R_2^{n-1} equals either to $Q \setminus T_1(\pi_{n-2})$ or $Q \setminus T_2(\pi_{n-2})$. Thus m' is consistent. Moreover if m is proper and $\Omega(p) = 2$ then $p \in T_2(\pi_{n-1})$ and by definition of σ_1 we get $m' = (p, a_{n-1}, R_1^{n-1}, R_2^{n-1})\pi_{n-2} \dots \pi_1$ and it follows that $h(m') \leq h(\pi_{n-1}) = h(tail(m))$. If additionaly $tail(m)$ is proper then $R_2^{n-1} = Q \setminus T_2(\pi_{n-2})$, hence m' is proper. $\square$

Lemma 35 Strategy σ_1 is winning.

Proof
To show that σ_1 is winning consider a play $(q_1, u_1), (q_2, u_2), \dots$ respecting σ_1; let $m_1, m_2, \dots,$ be the associated memories. If a state of priority 1 appears infinitely often on the play then it is winning for player 1.

If not then we will need one observation. For a memory $m = \pi_n \dots \pi_1$ define $\theta(m) = \max\{k + h(\pi_k) : k = 1, \dots, n\}$. From Lemma 34 it follows that for every position i:

$$\text{if } m_i \text{ is proper and } \Omega(q_{i+1}) = 2 \text{ then } \theta(m_i) \geq \theta(m_{i+1}) \tag{2}$$

Indeed, assume that $m_i = \pi_n \dots \pi_1$. From Lemma 34 we have the following inequalities :

if $|m_i| < |m_{i+1}|$ then $m_{i+1} = \pi_{n+1} m_i$ and

$$\begin{aligned} \theta(m_{i+1}) &= max(n + 1 + h(\pi_{n+1}), \theta(m_i)) \\ &\leq max(n + h(m_i), \theta(m_i)) = max(n + h(\pi_n), \theta(m_i)) = \theta(m_i) \end{aligned}$$

if $|m_i| > |m_{i+1}|$ then $m_{i+1} = \pi'_{n-1} \dots \pi_1$ and

$$\begin{aligned} \theta(m_{i+1}) &= max(n - 1 + h(\pi'_{n-1}), \theta(\pi_{n-2} \dots \pi_1)) \\ &\leq max(n - 1 + h(\pi'_{n-1}), \theta(m_i)) \\ &\leq max(n - 1 + h(\pi_{n-1}), \theta(m_i)) = \theta(m_i). \end{aligned}$$

Which proves the observation.

Consider now the position i after which no state of priority 1 appears. Take a position $k > i$ such that $|m_k| \leq |m_j|$ for all $j > k$. By induction, using Lemma 34 we get that after k all the memories of size bigger than $|m_k|$ are proper. By (2) we have that all the memories after k have their size bounded by $\theta(m_k)$ because $|m_j| \leq |m_j| + h(m_j) \leq \theta(m_j) \leq \theta(m_k)$. But, by Lemma 34 the size of the stack at a position is the same as the size of the memory at this position. Hence, the play is winning as the size of the stack is bounded. □

7.2 Reciprocal

For the implication in the other direction we take a memoryless winning strategy $\widetilde{\sigma}_0$ for player 0 in $\widetilde{G}$. We assume that the strategy is winning from every winning vertex for player 0. We will construct a winning strategy for player 0 in G.

Recall that $\widetilde{G}(\widetilde{\sigma}_0)$ denotes the game restricted to positions from where $\widetilde{\sigma}_0$ is winning and to edges permitted by $\widetilde{\sigma}_0$.

Every path in $\widetilde{G}(\widetilde{\sigma}_0)$ is winning. Hence for a position π there is a finite number of edges of priority 1 that can be reached from π; denote this number by $\tau_1(\pi)$. There is also a finite number of edges labeled 3 reachable without crossing an edge labeled 2 or 1. Denote this number by $\tau_3(\pi)$. Let the *height of a position* π be a pair $h(\pi) = (\tau_1(\pi), \tau_3(\pi))$. We will compare heights using lexicographic ordering $\preccurlyeq$ on pairs. We also use the relation $(\tau_1, \tau_3) \preccurlyeq_1 (\tau_1', \tau_3')$ which holds if $\tau_1 \leq \tau_1'$.

Definition 36 We define a strategy σ_0 using $\widetilde{\sigma}_0$ as follows. Suppose that the current position in the play in G is $(q_n, a_n \dots a_1)$ and the current memory is $m = \pi_n \dots \pi_1$ where $\forall i.\ \pi_i = (q_i, a_i, R_1^i, R_2^i)$.

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

  $$\pi_n \to CS(q_n, a_n, R_1^n, R_2^n, p, c) \to CM(q_n, a_n, R_1^n, R_2^n, p, c, S_1, S_2)$$

  then we define

  $$up(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$$

  Moreover if $q_n \in Q_0$ then we put:

  $$\varphi(m, (q_n, a_n \dots a_1)) = (p, ca_n \dots a_1)$$

- If in $\widetilde{G}(\widetilde{\sigma}_0)$ there is edge $\pi_n \to tt$ then for every $p \in R^n_{\Omega(p)}$ such that $pop(p) \in \Delta(q_n, a_n)$ we have

  $$up(m, (p, a_{n-1} \dots a_1)) = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$$

  where $R = R_1^{n-1}$ if $\Omega(p) = 1$ or $R_2^n = R_1^n$, and $R = R_2^{n-1}$ otherwise. Moreover if $q_n \in Q_0$ then we take p with $\Omega(p) = 1$ if possible and define

  $$\varphi(m, (q_n, a_n \dots a_1)) = (p, a_{n-1} \dots a_1)$$

For all other cases the update function and the strategy function are undefined.

Definition 37 We say that a pair of sets of states (S_1, S_2) is selected for a position $\pi = (p, a, R_1, R_2)$ and $q \in Q$, $b \in \Gamma$, denoted $(S_1, S_2) \in sel(\pi, q, b)$ if there exist $q' \in Q$, and $S_2', S_2'' \subseteq Q$ such that in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

$$\pi \to CS(p, a, R_1, R_2, q', b) \to CM(p, a, R_1, R_2, q', b, S_1, S_2') \to (q', b, S_1, S_2'')$$

and (q, b, S_1, S_2) is reachable from (q', b, S_1, S_2'') without passing through a push edge.

Definition 38 Consider a memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$, for $i = 1^l dotsn$. We say that m is *consistent* if all π_i are positions of $\widetilde{G}(\widetilde{\sigma}_0)$ and for all $i = 1, \dots, n-1$ we have $(R_1^{i+1}, R_2^{i+1}) \in sel(\pi_i, q_{i+1}, a_{i+1})$. We say that m is *glued* if $R_1^n = R_2^n$. The *height* of m is $h(m) = h(\pi_n)$. The *size* of m, denoted $|m|$ is n, i.e., the number of elements in m. We denote by $tail(m)$ the memory $\pi_{n-1} \dots \pi_1$.

Lemma 39 Suppose that while playing according to strategy σ_0 a position $(q_n, a_n \dots a_1)$ is reached with a consistent memory $m = \pi_n \dots \pi_1$ where $\pi_i = (q_i, a_i, R_1^i, R_2^i)$. The next move in $G(\sigma_0)$ is to one of the following positions:

$(p, ca_n \dots a_1)$. The updated memory is $m' = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$ and it is consistent. Moreover $h(m') \preccurlyeq_1 h(m)$ and if $\Omega(p) = 1$ then $h(m') \prec_1 h(m)$ and m' is glued.

$(p, a_{n-1} \dots a_1)$. The updated memory is $m' = (p, a_{n-1}, R_1^{n-1}, R)\pi_{n-2} \dots \pi_1$ and it is consistent. Moreover $h(m') \prec h(tail(m))$. If m is glued or $\Omega(p) = 1$ then $h(m') \prec_1 h(tail(m))$ and m' is glued. If $tail(m)$ is glued then m' is glued.

Proof

Consider the first clause. As m is consistent then π_n is winning. If there is a move from $(q_n, a_n \dots a_1)$ to $(p, ca_n \dots a_1)$ in $G(\sigma_0)$ then $push(p, c) \in \delta(q_n, a_n)$ and either $q_n \in Q_0$ or $q_n \in Q_1$ and the move is suggested by the strategy σ_0. In both cases we have some path $\pi_n \to CS(\pi_n, p, c) \to CM(\pi_n, p, c, S_1, S_2) \to^{\Omega(p)} (p, c, S_1, S_{\Omega(p)})$ in $\widetilde{G}(\widetilde{\sigma}_0)$. Therefore $(p, c, S_1, S_{\Omega(p)})$ is a winning position for player 0 and $(S_1, S_{\Omega(p)}) \in sel(\pi_n, p, c)$. From the definition of the strategy σ_0 we have that $m' = up(m, (p, ca_n \dots a_1)) = (p, c, S_1, S_{\Omega(p)})\pi_n \dots \pi_1$. Thus the memory m' is consistent. As the edge $CM(\pi_n, p, c, S_1, S_2) \to (p, c, S_1, S_{\Omega(p)})$ is labeled by $\Omega(p)$ we can deduce that $h(m') \preccurlyeq_1 h(m)$ and if $\Omega(p) = 1$ then $h(m') \prec_1 h(m)$ and m' is glued.

Consider the second clause. As m is consistent, the position π_{n-1} is winning, and $(R_1^n, R_2^n) \in sel(\pi_{n-1}, q_n, a_n)$, that is, there is in $\widetilde{G}(\widetilde{\sigma}_0)$ some path $\pi_{n-1} \to CS(\pi_{n-1}, q, a_n) \to CM(\pi_{n-1}, q, a_n, S_1, S_2) \to (q, a_n, S_1, S_2')$ and a path from (q, a_n, S_1, S_2') to (q_n, a_n, R_1^n, R_2^n) without passing through a push edge. By definition of $\widetilde{G}$ we get that $R_1^n = S_1$ and R_2^n equals either S_2 or S_1. By definition of σ_0 we have $p \in R_{\Omega(p)}^n$. Thus p belongs either to R_1^n or S_2. Hence $(p, a_{n-1}, R_1^{n-1}, R)$ is a successor of $CM(\pi_{n-1}, q_n, a_n, R_1^n, S_2)$ (through an edge labeled by 1 or 3). Therefore $(p, a_{n-1}, R_1^{n-1}, R)$ is winning for player 0 and $h(m') \prec h(\pi_{n-1}) = h(tail(m))$. Since by consistency of m we have that $(R_1^{n-1}, R_2^{n-1}) \in sel(\pi_{n-2}, q_{n-1}, a_{n-1})$, we deduce that $(R_1^{n-1}, R) \in$

$sel(\pi_{n-2}, q_{n-1}, a_{n-1})$. Thus m' is consistent. Moreover if $\Omega(p) = 1$ or m is glued then $p \in R_1^n$ and by definition of σ_0 we have $R = R_1^{n-1}$. Hence we have in this case $h(m') \prec_1 h(\pi_{n-1})$ and m' is glued. If $tail(m)$ is glued then $R = R_1^{n-1} = R_2^{n-1}$ so m' is glued.

$\square$

Lemma 40 Consider two positions $i < j$ such that $|m_i| = |m_j|$ and for every index $i < k < j : |m_k| \geq |m_i|$. We have that $h(m_i) \succ h(m_j)$. Moreover if a state of priority 1 appears between $i+1$ and j then $h(m_i) \succ_1 h(m_j)$.

Proof
Let $k < j$ be the greatest index such that $|m_k| = |m_j|$. If $k \neq i$ then by induction we get corresponding results for i and k. From hypothesis and Lemma 39 it follows that m_{j-1} has the form πm_k, for some π. By Lemma 39 we have : $h(m_j) \prec h(tail(m_{j-1})) = h(m_k) \prec h(m_i)$.

To show the second claim, first observe that by easy induction using Lemma 39 we have that if m_i is glued then m_j is glued. We proceed with the proof of the second claim by induction on the distance between i and j.

If $j = i+2$ and 1 appears between $i+1$ and $i+2$ then either $\Omega(q_{i+1}) = 1$ and by Lemma 39 we have m_{i+1} is glued; or $\Omega(q_j) = 1$. Whatever the case, Lemma 39 provides $h(m_j) \prec_1 h(m_i)$ and m_j is glued.

Suppose now that the distance is bigger. If there is $i < k < j$ with $|m_k| = |m_i|$ then we get the lemma by induction assumption applied to (i, k) and (k, j). If not then we know that all the memories between $i+1$ and $j-1$ are of size at least $|m_{i+1}|$. Hence, we can use the induction assumption there. We conclude with the help of Lemma 39. $\square$

Lemma 41 The strategy σ_0 is winning.

Proof
To show that σ_0 is winning consider a play $(q_1, u_1), (q_2, u_2), \ldots$ respecting σ_0; let $m_1, m_2, \ldots,$ be the associated memories.

By Lemma 40 there can be only finitely many occurrences of memory of a given size, in other words, the size of memory is unbounded. As, by Lemma 39, the size of memory at a position is the same as the size of stack at this position, we have that the size of the stack in the play is unbounded too.

It remains to show that there is only finite number of states of priority 1 in the play. Consider the sequence of positions $i_1, i_2, \ldots$ such that i_j is the last position where the memory of size $|m_{i_j}|$ occurs. By Lemmas 39 and 40 we have that $h(m_{i_j}) \succcurlyeq_1 h(m_{i_{j+1}})$ and the inequality is strict if there is a state of priority 1 between i_j+1 and i_{j+1}. As $\preccurlyeq_1$ is well-founded, it is not possible that there are infinitely many states of priority 1 in the sequence. $\square$

8 Büchi and unboundedness

In this section we consider pushdown games with the condition: "green states appear infinitely often and the stack is unbounded". Formally, we have a priority function Ω assigning to the states of the game priorities 0 or 1. The set of winning paths is:

$$Acc(\Omega \wedge U) = Acc_\Omega \cap Acc_U$$

Fix a pushdown game G defined by a pushdown system $\langle Q, \Gamma, \Delta \rangle$, a partition of states (Q_0, Q_1) and a priority function $\Omega : Q \to \{0, 1\}$. We will solve this game by reducing it to a finite game.

8.1 All the strategies for such a game may need an infinite memory

The following example hints why we do the reduction in two steps. The example shows that in games with $Acc(\Omega \wedge U)$ condition, all the strategies for player 0 may need infinite memory.
Example: Consider the pushdown system $\mathcal{A} = \langle Q, \Gamma, \Delta \rangle$ where : $Q = \{p, q\}$, $\Gamma = \{a, \bot\}$ and

$$\Delta = \{(p, \bot, push(q, a)), (q, a, push(q, a)), (q, a, pop(q)), (q, a, pop(p))\}$$

From state p on letter $\bot$ the automaton pushes a and goes to state q. Then it can push and pop letters a until it decides to change state back to p. If it arrives at p with a stack other than $\bot$, it is blocked.

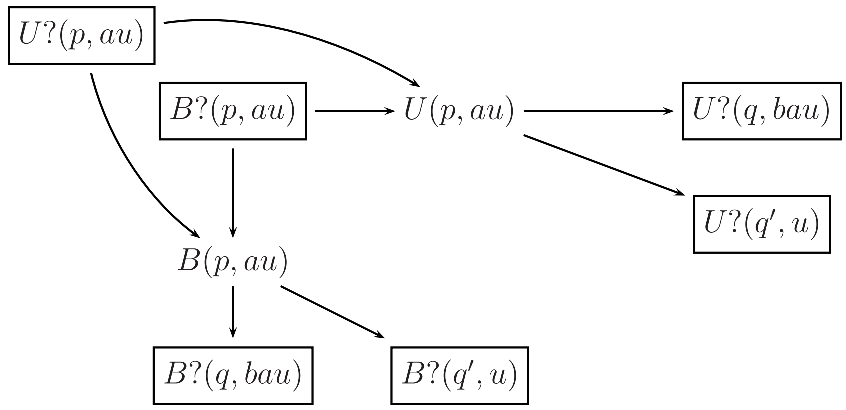
We assume that all the states belong to player 0, i.e., $Q_0 = Q$ and $Q_1 = \emptyset$. We take the Büchi condition $\Omega(p) = 0$ and $\Omega(q) = 1$. Thus $Acc(\Omega \wedge U)$ is the set of plays where the stack is unbounded and state p is visited infinitely often, which in this case means that the stack must contain only $\bot$ infinitely often. Therefore in order to win player 0 needs to go from the configuration $(p, \bot)$ to a configuration $(q, a^i\bot)$ and back to $(p, \bot)$ repeatedly for bigger and bigger i. It is easy to see that any such strategy requires infinite memory as being in the configuration $(p, \bot)$ player 0 needs to memorize which height he wants to reach next.

Finally note that in the same pushdown game but with strict unboundedness condition instead of unboundedness, player 0 has no winning strategy. □

Because of the complications presented in the example, the first step in solving G is to reduce it to another pushdown game G^* as depicted below.

8.2 Reduction

This game has two modes so as to let player 1 to verify that player 0 can win the play in mode B with the Büchi condition or in mode U with the unboundedness condition. Only Player 1 can change the mode and he can do it anytime. However if he changes it infinitely often, he looses.



Formally, for every mode $K \in \{B, U\}$ and for every configuration (p, u) in the pushdown game G, there is in G^* an edge from $K?(p, u)$ to $K(p, u)$ and to $\overline{K}(p, u)$, where $\overline{K}$ denotes the "other letter". For every edge from (p, u) to (q, v) in G, there is an edge in G^* from $K(p, u)$ to $K?(q, v)$.

All the positions with an interrogative key $K?$ are for player 1. A position $K(p, u)$ is for player 0 in G^* if and only if (p, u) is for player 0 in G. The acceptance condition consists of the sequences $K_1?(p_1, u_1)K_2(p_1, u_1)K_2?(p_2, u_2)\ldots$ such that either $(p_1, u_1)(p_2, u_2) \cdots \in Acc(\Omega \wedge U)$ or there are infinitely many i such that $K_i = B$ and $K_{i+1} = U$.

Theorem 42
Player 0 has a winning strategy from a position (p, u) in G iff he has a winning strategy from positions $B?(p, u)$ and $U?(p, u)$ in G^*.

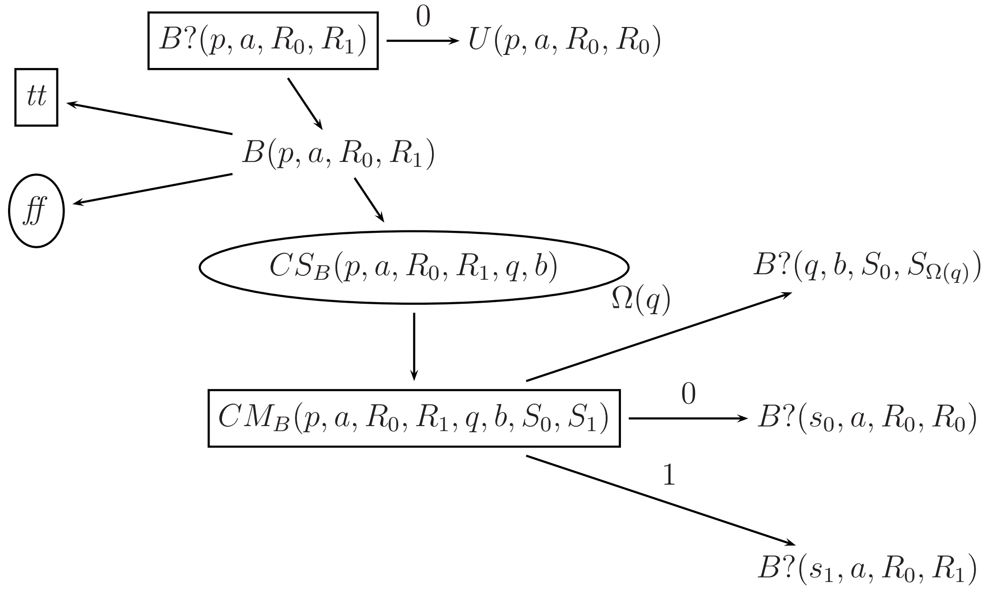
Proof

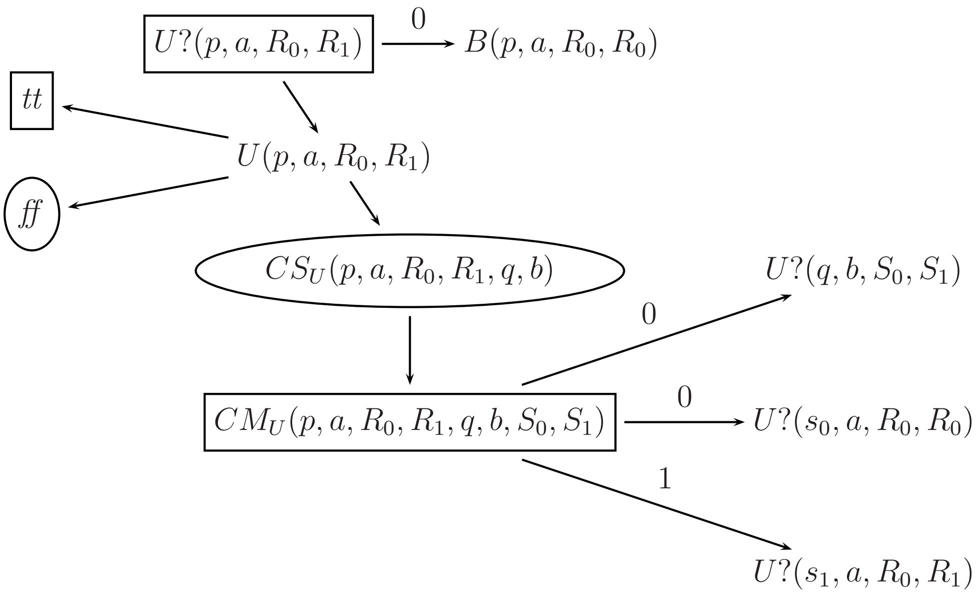
A winning strategy σ_0 for player 0 in G induces a winning strategy for player 0 in G^*. It is enough that player 0 doesn't take account of the mode chosen by player 1 and just plays σ_0. The strategy guarantees that the stack will be unbounded and that a state of priority 0 will appear infinitely often. Hence, no matter in what mode the play will stay, player 0 will win.

From a winning strategy σ_0^* for player 0 in G^*, we construct a strategy σ_0 for player 0 in G, with a memory keeping the greatest height of stack reached so far. First, we use σ_0^* in Büchi mode, until we reach a state of priority 0. Then, we simulate player's 1 switch to unboundedness mode and simulate σ_0^* until the play does reach the stack of height bigger than the maximum so far. Next, once again we switch to Büchi mode etc. This way we visit infinitely often the states of priority 0 and we are sure that the stack is unbounded. $\square$

8.3 Simulation

From the pushdown game G^*, we construct a finite-state Büchi game $\widetilde{G}$ with coloring on edges as follows :





We define sets $Mode = \{B, U\}$ and $Residues = Q \times \Gamma \times 2^Q \times 2^Q$. As depicted, the set of positions $\widetilde{V}$ of $\widetilde{G}$ consists of:

$$K(\eta),\ K?(\eta),\ CS_K(\eta, q, b),\ CM_K(\eta, \eta'),\ tt,\ ff$$

for all $K \in Mode$, $\eta, \eta' \in Residues$, $q \in Q$, $b \in \Gamma$.

Among them positions for player 0 are:

$$CS_K(\eta, q, b),\ ff,\ K(\eta')$$

for all K, η, q, b and η' with the state in η' from Q_0. The remaining positions are for player 1.

From $B(p, a, R_0, R_1)$ the edge to tt is when we have $pop(q) \in \Delta(p, a)$ and $q \in R_{\Omega(q)}$. The edge to ff is when we have $pop(q) \in \Delta(p, a)$ and $q \notin R_{\Omega(q)}$. From $U(p, a, R_0, R_1)$ the edge to tt is when we have $pop(q) \in \Delta(p, a)$ and $q \in R_1$. The edge to ff is when we have $pop(q) \in \Delta(p, a)$ and $q \notin R_1$. From $K(p, a, R_0, R_1)$ the edge to $CS_K(p, a, R_0, R_1, q, b)$ is when we have $push(q, b) \in \Delta(p, a)$. From a position $CS_K(p, a, R_0, R_1, q, b)$ there is an edge to $CM_K(p, a, R_0, R_1, q, b, S_0, S_1)$ for every $S_0, S_1 \subseteq Q$. From a position $CM_K(p, a, R_0, R_1, q, b, S_0, S_1)$ there is an edge, called *push edge* to $(q, b, S_0, S_{\Omega(q)})$, and edges to (s_0, a, R_0, R_0) and (s_1, a, R_0, R_1) for every $s_0 \in S_0$ and every $s_1 \in S_1$, respectively.

The winning condition is a parity condition defined by the priorities on edges as in the figure.

Theorem 43
For every $p \in Q$, $a \in \Gamma$ and $R \subseteq Q$:

$$K(p,a) \in W_0(G^*(R)) \quad \textit{iff} \quad K(p,a,R,R) \in W_0(\widetilde{G})$$

8.4 Direct implication :

We prove for every state $p \in \mathcal{Q}$, every letter $a \in \Gamma$ every subset $R \subseteq \mathcal{Q}$, we have :

$$K(p,a,R,R) \in W_1(\widetilde{G}) \Rightarrow K(p,a) \in W_1(G^*(R))$$

Let $\widetilde{\sigma}_1$ be a memoryless winning strategy for player 1 from every position in $W_1(\widetilde{G})$. Strategy $\widetilde{\sigma}_1$ defines a game where only player 0 plays, denoted by $\widetilde{G}(\widetilde{\sigma}_1)$. This game is a restriction of the game $\widetilde{G}$ to the vertices where $\widetilde{\sigma}_1$ is winning and to the edges suggested by $\widetilde{\sigma}_1$. Hence every position of player 1 has only one successor and every infinite path is winning for player 1.

In the following, it will be useful to consider strongly connected components of $\widetilde{G}(\widetilde{\sigma}_1)$. It's important to remark that two positions connected by an edge colored by 0, belong to different components of $\widetilde{G}(\widetilde{\sigma}_1)$.

Using $\widetilde{\sigma}_1$, we construct a strategy with memory $\sigma = \langle \Phi, up, m_0 \rangle$ in G^*. The memory will be $M = (\widetilde{Pos})^*$ and the update function up will use it as a stack over alphabet $\widetilde{Pos}$. At the beginning of the play the initial state of the memory is $m_0 = (K, (p,a,R,R))$.

A position $K?(\eta)$ for which the strategy is defined is called *switching* if $\widetilde{\sigma}_1(K?(\eta)) = \overline{K}(\eta)$. A position $K(p,a,R_0,R_1)$ is *glued* if $R_0 = R_1$.

Let $h(\pi)$ be the number of strongly connected components of $\widetilde{G}(\widetilde{\sigma}_1)$ reachable from the position π.

We start with an auxiliary definition.

Definition 44 For every position $\pi = K(q,b,R_0,R_1)$ of $\widetilde{G}(\widetilde{\sigma}_1)$ we define sets $T_i(\pi)$ for $i \in \{0,1\}$:

$$\begin{aligned} T_0(\pi) =& \{q' : h(K?(q',b,R_0,R_0)) < h(\pi)\} \cup \\ & \{q' : h(K?(q',b,R_0,R_1)) \le h(\pi) \text{ and } K?(q',b,R_0,R_1) \text{ switching}\} \\ T_1(\pi) =& \{q' : h(K?(q',b,R_0,R_1)) \le h(\pi)\} \end{aligned}$$

Definition 45 We define σ_1 using $\widetilde{\sigma}_1$ as follows. Suppose that the current position in the play in $G^*(S)$ is $K(q,bu)$ and the current state of the memory is $\vec{m} = \pi_n \dots \pi_1$ where $\forall i.\ \pi_i = K_i(q_i,a_i,R_0^i,R_1^i)$ and $q = q^n$, $b = a^n$:

- If the move from π_n to $CS_K(q_n,a_n,R_0^n,R_1^n,q',c)$ is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then we set :

$$up(m, K(q',cbu)) = K?(q',c,S_0,S_1)\pi_n \dots \pi_1$$

where the sets S_0, S_1 are such that we have :

$$\widetilde{\sigma}_1(CM_K(q_n, a_n, R_0^n, R_1^n, q', c, \mathcal{Q}\backslash T_0(\pi_n), \mathcal{Q}\backslash T_1(\pi_n))) = K?(q', c, S_0, S_1).$$

If $q \in \mathcal{Q}_1$ then additionally we define :

$$\phi(m, K(q, bu)) = K?(q', cbu)$$

- If the move from π_n to *ff* is possible in $\widetilde{G}(\widetilde{\sigma}_1)$ then for q' such that $q' \notin R_1^n$ and $pop(q') \in \Delta(q, b)$ we set:

$$up(m, K?(q', u)) = \pi'_{n-1}\pi_{n-2}\ldots\pi_1$$

  where $\pi'_{n-1} = K_{n-1}?(q', a_{n-1}, R_0^{n-1}, R_0^{n-1})$ if the following inequality $h(K_{n-1}?(q', a_{n-1}, R_0^{n-1}, R_0^{n-1})) < h(\pi_{n-1})$ holds. Otherwise we have $\pi'_{n-1} = K_{n-1}?(q', a_{n-1}, R_0^{n-1}, R_1^{n-1})$.
  If $q \in \mathcal{Q}_1$ then we take q' with π'_{n-1} of the first kind if possible, if not then of the second kind and define.

$$\phi(m, K(q, bu)) = K?(q', u)$$

When the current position in the play in $G^*(R)$ is $L?(q, bu)$ and the current state of the memory $m = \pi_{n+1}\pi_n\ldots\pi_1$ with $\pi_{n+1} = K?(q, b, R_0, R_1)$ we have:

- if $\widetilde{\sigma}(\pi_{n+1}) = K(q, b, R_0, R_1)$, then :

$$\begin{aligned} up(m, K(q, bu)) &= K(q, b, R_0, R_1)\pi_n\ldots\pi_1 \\ \phi(m, L?(q, bu)) &= K(q, bu) \qquad \text{if } q \in \mathcal{Q}_1 \end{aligned}$$

- if $\widetilde{\sigma}(\pi_{n+1}) = \overline{K}(q, b, R_0, R_0)$, then :

$$\begin{aligned} up(m, K?(q, bu)) &= \overline{K}(q, b, R_0, R_0)\pi_n\ldots\pi_1 \\ \phi(m, L?(q, bu)) &= \overline{K}(q, bu) \qquad \text{if } q \in \mathcal{Q}_1 \end{aligned}$$

Definition 46 From a position $K(q, u)$, the associated memory $m = \pi_n\ldots\pi_1$ where $\pi_i = K_i(q_i, a_i, R_0^i, R_1^i)$ is *consistent* if $q_n = q$, $u = a_n\ldots a_1$ and all π_i are winning positions, $R_0^{i+1} = \mathcal{Q} \setminus T_0(\pi_i)$ and $R_1^{i+1} = \mathcal{Q} \setminus T_0(\pi_i)$ or $R_1^{i+1} = \mathcal{Q} \setminus T_1(\pi_i)$. Moreover $R_0^i = R_1^i$ if $K_i \neq K_{i+1}$.

Lemma 47 Suppose that while using strategy σ_1 a position $K_n(q_n, a_n \dots a_1)$ is reached with consistent memory $m = \pi_n \dots \pi_1$, where $\pi_i = K_i(q_i, a_i, R_0^i, R_1^i)$. The next two moves in $G^*(\sigma_1)$ are in one of the following patterns:

1. $\rightarrow K_n?(p, ca_n \dots a_1) \rightarrow K_n(p, ca_n \dots a_1)$.

   We have that $m'' = K_n(p, c, S_0, S_1)\pi_n \dots \pi_1$ and it is consistent. Moreover $h(m'') \leq h(m)$ and the inequality is strict if $K_n = U$ or $K_n = B$ and $\Omega(p) = 0$. In the last case $S_0 = S_1$.

2. $\rightarrow K_n?(p, ca_n \dots a_1) \rightarrow \overline{K}_n(p, ca_n \dots a_1)$.

   We have that $m'' = \overline{K}_n(p, c, S_0, S_0)\pi_n \dots \pi_1$ and it is consistent. Moreover $h(m'') < h(m)$.

3. $\rightarrow K_n?(p, a_{n-1} \dots a_1) \rightarrow K_n(p, a_{n-1} \dots a_1)$.

   We have that $m'' = K_n(p, a_{n-1}, S_0, S_1)\pi_{n-2} \dots \pi_1$ and it is consistent. Moreover, $h(m'') \leq h(\pi_{n-1})$ and if π_n is glued, or $K_n = B$ and $\Omega(p) = 0$ then the inequality is strict and $S_0 = S_1$.

4. $\rightarrow K_n?(p, a_{n-1} \dots a_1) \rightarrow \overline{K}_n(p, a_{n-1} \dots a_1)$.

   We have that $m'' = \overline{K}_n(p, a_{n-1}, S_0, S_0)\pi_{n-2} \dots \pi_1$ is consistent and $h(m'') < h(\pi_{n-1})$.

Proof

Consider the first two clauses. If we have a move $K_n(q_n, a_n \dots, a_1) \rightarrow K_n?(p, ca_n \dots a_1)$ in $G^*(\sigma_1)$ then we have a move from π_n to the position $CS_{K_n}(q_n, a_n, R_0^n, R_1^n, p, c)$ in $\widetilde{G}(\widetilde{\sigma}_1)$. By definition

$$m' = up(m, K_n?(p, ca_n \dots a_1)) = K_n?(p, c, S_0, S_1)\pi_n \dots \pi_1$$

where $S_0 = \mathcal{Q} \setminus T_0(\pi_n)$ and $S_1 = \mathcal{Q} \setminus T_1(\pi_n)$. By the definition of S_0, S_1 we have that $CM_{K_n}(q_n, a_n, R_0^n, R_1^n, p, c, S_0, S_1) \rightarrow K_n?(p, c, S_0, S_1)$ is an edge in $\widetilde{G}(\widetilde{\sigma}_1)$. Hence $h(K_n?(p, c, S_0, S_1)) \leq h(\pi_n)$ and the inequality is strict if $K_n = U$ or $K_n = B$ and $\Omega(p) = 0$. In the last case $S_0 = S_1$.

For the first clause. If the next move from $K_n?(p, ca_n \dots a_1)$ reaches the position $K_n(p, ca_n \dots a_1)$ then $K_n?(p, c, S_0, S_1) \rightarrow K_n(p, c, S_0, S_1)$ is an edge in $\widetilde{G}(\widetilde{\sigma}_1)$. So $h(K_n(p, c, S_0, S_1)) \leq h(K_n?(p, c, S_0, S_1))$. The memory $m'' = K_n(p, c, S_0, S_1)\pi_n \dots \pi_1$ is of the form as required and it is consistent.

For the second clause. If the next move from $K_n?(p, ca_n \dots a_1)$ reaches the position $\overline{K}_n(p, ca_n \dots a_1)$ then $K_n?(p, c, S_0, S_1) \rightarrow \overline{K}_n(p, c, S_0, S_0)$ is an edge in $\widetilde{G}(\widetilde{\sigma}_1)$. We have that $h(\overline{K}_n(p, c, S_0, S_0)) < h(\pi_n)$. The memory $m'' = \overline{K}_n(p, c, S_0, S_0)\pi_n \dots \pi_1$ is of the form as required and it is consistent.

Consider the last two clauses. If we have a move $K_n(q_n, a_n \ldots, a_1) \to K_n?(p, a_{n-1} \ldots a_1)$ then $\pi_n \to \mathit{ff}$ is possible in $\widetilde{G}(\widetilde{\sigma}_1)$. By definition of $\widetilde{G}(\widetilde{\sigma}_1)$ we have $p \notin R_1^n$ with $pop(p) \in \Delta(q_n, a_n)$. The updated memory is:

$$m' = up(m, K?(p, a_{n-1} \ldots a_1)) = K_{n-1}?(p, a_{n-1}, S_0, S_1)\pi_{n-2} \ldots \pi_1$$

for $S_0 = R_0^{n-1}$ and $S_1 = R_1^{n-1}$ or $S_1 = R_0^{n-1}$. By consistency of memory m we have $R_1^n = T_1(\pi_{n-1})$, hence $h(K_{n-1}?(p, a_{n-1}, S_0, S_1)) \leq h(\pi_{n-1})$. If π_n is glued then $p \in R_0^n = T_0(\pi_{n-1})$. By definition of $T_0(\pi_{n-1})$, position $K_{n-1}?(p, a_{n-1}, S_0, S_1)$ is either glued or switching. If it is glued then $h(K_{n-1}?(p, a_{n-1}, S_0, S_1)) < h(\pi_{n-1})$.

For the third clause. If the next move from $K_n?(p, a_{n-1} \ldots a_1)$ reaches the position $K_n(p, a_{n-1} \ldots a_1)$ then it means that in $\widetilde{G}(\widetilde{\sigma}_1)$ there is a move from $K_{n-1}?(p, a_{n-1}, S_0, S_1)$ either to $K_n(p, a_{n-1}, S_0, S_1)$ or to $K_n(p, a_{n-1}, S_0, S_0)$. The first case is when $K_{n-1} = K_n$ the other when the equality does not hold.

If $K_{n-1} \neq K_n$ then $h(K_{n-1}(p, a_{n-1}, S_0, S_0)) < h(K_{n-1}?(p, a_{n-1}, S_0, S_1))$ and we are done. Observe that this case happens if $K_{n-1}?(p, a_{n-1}, S_0, S_1)$ is switching.

If $K_{n-1} = K_n$ then we have that

$$h(K_n(p, a_{n-1}, S_0, S_1)) \leq h(K_{n-1}?(p, a_{n-1}, S_0, S_1)) \leq h(\pi_{n-1})$$

If $K_n?(q_n, a_n, R_0^n, R_1^n)$ is glued, or $K_n = B$ and $\Omega(p) = 0$ then $p \in T_0(\pi_{n-1})$, therefore the second inequality is strict and $S_0 = S_1$.

In both cases the memory $m'' = K_n(p, a_{n-1}, S_0, S_1)\pi_{n-2} \ldots \pi_1$ is of required form and consistent.

For the fourth clause. If the next move from $K_n?(p, a_{n-1} \ldots a_1)$ reaches $\overline{K}_n(p, a_{n-1} \ldots a_1)$ then in $\widetilde{G}(\widetilde{\sigma}_1)$ there is a move from $K_{n-1}?(p, a_{n-1}, S_0, S_1)$ either to $\overline{K}_n(p, a_{n-1}, S_0, S_1)$ or to $\overline{K}_n(p, a_{n-1}, S_0, S_0)$.

The first case is when $K_{n-1} \neq K_n$. By memory consistency we have then $p \notin R_0^n = R_1^n$. Hence,$p \in T_0(\pi_{n-1})$ and as $K_{n-1}?(p, a_{n-1}, S_0, S_1)$ is not switching, we have:

$$h(\overline{K}_n(p, a_{n-1}, S_0, S_1)) \leq h(K_{n-1}?(p, a_{n-1}, S_0, S_1)) < h(\pi_{n-1})$$

If $K_{n-1} = K_n$ then

$$h(\overline{K}_n(p, a_{n-1}, S_0, S_0)) < h(K_{n-1}?(p, a_{n-1}, S_0, S_1)) \leq h(\pi_{n-1})$$

In both cases the memory $m'' = \overline{K}_n(p, a_{n-1}, S_0, S_1)\pi_{n-2} \ldots \pi_1$ is of required form and consistent. $\square$

To show that σ_1 is a winning strategy for player 1, consider any play $K_1?(q_1, u_1), K_2(q_2, u_2), K_3?(q_3, u_3) \ldots$ respecting σ_1. Let $m_1, m_2, m_3, \ldots$ be the associated memories.

Definition 48 A size $|m|$ of memory $m = \pi_n \dots \pi_1$ is the number of elements in m. The height for memory m is $h(m) = h(\pi_n)$ where π_n is the top element of m. We denote by $tail(m)$ the memory $\pi_{n-1} \dots \pi_1$.

Lemma 49 If m_i is glued and k is such that $|m_k| < |m_i|$ and there are no memories of size smaller than m_k between i and k then m_k is glued.

Proof
Using Lemma 47. By induction on the distance between i and k. $\square$

Lemma 50 If there is either a change of mode between m_{2i-1} and m_{2i}; or $K_{2i-1} = B$ and $\Omega(q_{2i-1}) = 0$, whereas $|m_{2i}| < |m_{2i-2}|$ then m_{2i} is glued and $h(m_{2i}) < h(tail(m_{2i-2}))$.

Proof
From Lemma 47. Observe that m_{2i-1} has K_{2i-1}? but m_{2i-2} and m_{2i} have keys without question marks. $\square$

Lemma 51 If there is either a change of mode between m_{2i-1} and m_{2i}; or $K_{2i-1} = B$ and $\Omega(q_{2i-1}) = 0$, whereas $|m_{2i-2}| < |m_{2i}|$ then m_{2i} is glued and $h(m_{2i-2}) > h(m_{2i})$.

Proof
From Lemma 47. $\square$

Lemma 52 Suppose that $i < k$ are two even indices of two memories m_i, m_k of the same size such that all the memories in between are *strictly* of bigger size. We have $h(m_i) \geq h(m_k)$ and the inequality is strict if m_{k-2} is glued.

Proof
From Lemmas 51, 50 and Lemma 49. $\square$

Lemma 53 Suppose that $i < k$ are two even indices of two memories m_i, m_k of the same size such that all the memories in between are of bigger or same size. We have $h(m_k) \leq h(m_i)$ and the inequality is strict if there is either a change of mode or a visit in Büchi of a state of priority 0 between associated positions.

Proof
From Lemma 52 and by induction on the distance between indices of memories of same size than $|m_i|$. $\square$

Lemma 54 There are finitely many changes of modes and states of priority 0 in Büchi mode in the play.

Proof
Suppose the contrary. First assume that there were infinitely many memories of the same size. Take the smallest size repeating infinitely often. Let $i_1, i_2, \dots$ be the even positions of memories of the chosen size. We have by Lemma 53 $h(m_{i_1}) \geq h(m_{i_2}) \geq \dots$ and the inequality is strict each time when between $m_{i_{j-1}}$ and m_{i_j} there is a change of mode or a state of priority 0 in Büchi mode.

If no size of memory repeats infinitely often then consider the sequence $i_0, i_0+2, i_1 \dots$ of even indices such that m_{i_k} is the last occurrence of a memory of size k ; and therefore where m_{i_k+2} is the first occurrence of a memory of size $k+1$ from which no memory of size less or equal than k is met. We have by Lemma 51 and Lemma 53 that $h(m_{i_0}) \geq h(m_{i_0+2}) \geq h(m_{i_1}) \dots$ and the inequality is strict each time there is a change of mode between two positions or there is a state of priority 0 in Büchi mode. This is impossible. □

Lemma 55 If $K_{2i-2} = K_{2i-1} = K_{2i} = U$ and $\Omega(q_{2i-1}) = 0$, whereas $|m_{2i-2}| < |m_{2i}|$ then $h(m_{2i-2}) > h(m_{2i})$.

Proof
From Lemma 47. □

Lemma 56 If the play remains in mode U from $U(p, u)$ with a memory m_i such that every next memories m_j is of bigger or same size than m_i, then $h(m_j) \leq h(m_i) - (|m_j| - |m_i|)$. Therefore we have that $|m_j| \leq h(m_i) + |m_i|$ and by consistency the height of the stack is bounded.

Proof
From Lemma 53 and Lemma 55. □

Corollary 57 Strategy σ_1 is winning.

Proof
From Lemma 54 and Lemma 56. □

8.5 Reciprocal implication :

We prove for every state $p \in \mathcal{Q}$, every letter $a \in \Gamma$ every subset $R \subseteq \mathcal{Q}$, we have :

$$K(p, a, R, R) \in W_0(\widetilde{G}) \Rightarrow K(p, a) \in W_0(G^*(R))$$

Let $\widetilde{\sigma}_0$ be a memoryless winning strategy for player 0 from every position in $W_0(\widetilde{G})$. Strategy $\widetilde{\sigma}_0$ defines a game where only player 1 plays, denoted by $\widetilde{G}(\widetilde{\sigma}_0)$. This game is a restriction of the game $\widetilde{G}$ to the vertices where $\widetilde{\sigma}_0$ is winning and to the edges suggested by $\widetilde{\sigma}_0$. Hence every position of player 0 has only one successor and every infinite path is winning for player 0.

Using $\widetilde{\sigma}_0$, we construct a strategy with memory $\sigma_0 = \langle \Phi, up, m_0 \rangle$ in $G^*(R)$. The memory will be $M = (\widetilde{Pos})^*$ and the update function up will use it as a stack over alphabet $\widetilde{Pos}$. At the beginning of the play the initial state of the memory is $m_0 = (K(p, a, R, R))$.

Let $dist_0(\pi)$ be the maximal distance from π up to a 0-colored edge or tt in $\widetilde{G}(\widetilde{\sigma}_0)$. It is well defined since every position in $\widetilde{G}(\widetilde{\sigma}_0)$ is winning.

Definition 58 We say that a pair of sets of states (S_0, S_1) is selected for a position $\pi = K(p, a, R_0, R_1)$, $q \in Q$, $b \in \Gamma$ and $K' \in Mode$, denoted $(S_0, S_1) \in sel_{K'}(\pi, q, b)$ if there exists $q' \in Q$, $S_1', S_1'' \subseteq \mathcal{Q}$ such that in $\widetilde{G}(\widetilde{\sigma}_0)$ there is a sequence:

$$\begin{aligned} \pi \to\ & CS_K((p, a, R_0, R_1, q', b) \to CM_K(p, a, R_0, R_1, q', b, S_0, S_1') \\ & \to K?(q', b, S_0, S_1'') \end{aligned}$$

and $K'(q, b, S_0, S_1)$ is reachable from $K?(q', b, S_0, S_1'')$ without passing through a push edge.

Definition 59 We define σ_0 using $\widetilde{\sigma}_0$ as follows. Suppose that the current position in the play in $G^*(R)$ is $K(q, bu)$ and the current state of the memory is $\vec{m} = \pi_n \dots \pi_1$ where $\forall i.\ \pi_i = K_i(q_i, a_i, R_0^i, R_1^i)$ and $q = q^n$, $b = a^n$ and $K = K_n$:

- If the move from π_n to $CS_K(q_n, a_n, R_0^n, R_1^n, q', c)$ is possible in $\widetilde{G}(\widetilde{\sigma}_0)$ then we set :

  $$up(m, K(q', cbu)) = K?(q', c, S_0, S)\pi_n \dots \pi_1$$

  where $CS_K(q_n, a_n, R_0^n, R_1^n, q', c) \to CM_K(q_n, a_n, R_0^n, R_1^n, q', c, S_0, S_1) \to (q', c, S_0, S)$ is a path in $\widetilde{G}(\widetilde{\sigma}_0)$, and S equals either S_1 if $K = U$; or $S_{\Omega(q)}$ if $K = B$.

  If $q \in \mathcal{Q}_0$ then additionally we define:

  $$\phi(m, K(q, bu)) = K?(q', cbu)$$

- If the move from π_n to tt is possible in $\widetilde{G}(\widetilde{\sigma}_0)$ then for all q' such that $q' \in R_1^n$ if $K_n = U$ or $q' \in R_{\Omega(q')}^n$ if $K_n = B$, and $pop(q') \in \Delta(q, b)$ we set:
$$up(m, K?(q', u)) = \pi'_{n-1}\pi_{n-2}\ldots\pi_1$$
where $\pi'_{n-1} = K_{n-1}?(q', a_{n-1}, R_0^{n-1}, R_1^{n-1})$ if $K_n = U$ or $\Omega(q') = 1$. Otherwise, $\pi'_{n-1} = K_{n-1}?(q', a_{n-1}, R_0^{n-1}, R_0^{n-1})$

  If $q \in \mathcal{Q}_0$ then we take q' with π'_{n-1} of the second kind if possible, if not then of the first kind and define.
$$\phi(m, K(q, bu)) = K?(q', u)$$

When the current position in the play in $G^*(R)$ is $L?(q, bu)$ and the current state of the memory $m = \pi_{n+1}\pi_n \ldots \pi_1$ with $\pi_{n+1} = K?(q, b, R_0, R_1)$ we have:

if the next move chosen by player 1 is $K'(q, bu)$, then we set :
$$up(m, K'(q, bu)) = K'(q, b, R_0, S_1)\pi_n \ldots \pi_1$$
where $S_1 = R_0$ if $K' = K_n$, and $S_1 = R_1$ otherwise.

Definition 60 From a position $K(q, u)$, the associated memory $m = \pi_n \ldots \pi_1$ where $\pi_i = K_i(q_i, a_i, R_0^i, R_1^i)$ is *consistent* if $q_n = q$, $u = a_n \ldots a_1$, all π_i are winning positions and for all $i = 1, \ldots, n-1$ we have $(R_0^{i+1}, R_1^{i+1}) \in sel_{K_{i+1}}(\pi_i, q_{i+1}, a_{i+1})$.

Lemma 61 Suppose that while playing according to strategy σ_0 a position $K_n(q_n, a_n \ldots a_1)$ is reached with consistent memory $m = \pi_n \ldots \pi_1$, where $\pi_i = K_i(q_i, a_i, R_0^i, R_1^i)$. Assume that $K_{n-1} = K_n$. Then the next two moves where player 1 does not change the mode, in $G^*(\sigma_0)$ are in one of the following patterns:

- If $K_n = B$:
  - $\rightarrow B?(p, ca_n \ldots a_1) \rightarrow B(p, ca_n \ldots a_1)$. We have that the updated memory $m'' = B(p, c, S_0, S_1)\pi_n \ldots \pi_1$ and it is consistent. Moreover $dist_0(m'') < dist_0(m)$ if $\Omega(p) = 1$.
  - $\rightarrow B?(p, a_{n-1} \ldots a_1) \rightarrow B(p, a_{n-1} \ldots a_1)$. We have that $m'' = B(p, a_{n-1}, R_0^{n-1}, R)\pi_{n-2} \ldots \pi_1$ and it is consistent. Moreover, $R = R_1^{n-1}$ and $dist_0(m'') < dist_0(tail(m))$ if $\Omega(p) = 1$.

- If $K_n = U$:
  - $\rightarrow U?(p, ca_n \dots a_1) \rightarrow U(p, ca_n \dots a_1)$. We have that the updated memory $m'' = U(p, c, S_0, S_1)\pi_n \dots \pi_1$ and it is consistent.
  - $\rightarrow U?(p, a_{n-1} \dots a_1) \rightarrow U(p, a_{n-1} \dots a_1)$. We have that $m'' = U(p, a_{n-1}, R_0^{n-1}, R_1^{n-1})\pi_{n-2} \dots \pi_1$ and it is consistent. Moreover, $dist_0(m'') < dist_0(tail(m))$.

Proof

For the first clause : As m is consistent π_n is winning. By definition of σ_0 we have that $\pi_n = B(\eta_n) \rightarrow CS_B(\eta_n, p, c) \rightarrow CM_B(\eta_n, p, c, S_0, S_1) \rightarrow^{\Omega(p)} B?(p, c, S_0, S_{\Omega(q)})$ is a path in $\widetilde{G}(\sigma_0)$. Therefore m'' is consistent and if $\Omega(p) = 1$ then between π_n and $B(p, c, S_0, S_1)$ there is an edge labeled by 1, thus $dist_0(m'') < dist_0(m)$.

For the second clause : As m is consistent we have that π_{n-1} is winning, $(R_0^{n-1}, R_1^{n-1}) \in sel_{K_n}(\pi_{n-2}, q_{n-1}, a_{n-1})$, and there is both in $\widetilde{G}(\sigma_0)$ the path $\pi_{n-1} = B(\eta_{n-1}) \rightarrow CS_B(\eta_{n-1}, q, a_n) \rightarrow CM_B(\eta_{n-1}, q, a_n, R_0^n, S) \rightarrow^{\Omega(p)} B?(q, a_n, R_0^n, S')$ and a path from $B?(q, a_n, R_0^n, S')$ to $B(q_n, a_n, R_0^n, R_1^n)$ without passing through a push edge. By definition of $\widetilde{G}$, R_1^n equals either R_0^n or S. By definition of σ_0 we have $p \in R_{\Omega(p)}^n$. Thus p belongs either to R_0^n or S. Hence $B(p, a_{n-1}, R_0^{n-1}, R)$ is a successor of $CM_B(\eta_{n-1}, q, a_n, R_0^n, S)$. Therefore m'' is consistent. Moreover if $\Omega(p) = 1$ then by definition of σ_0 we have $R = R_1^{n-1}$. Hence $p \in S$ and the edge $CM_B(\eta_{n-1}, q, a_n, R_0^n, S) \rightarrow B(p, c, R_0^{n-1}, R_1^{n-1})$ is labeled by 1, thus we have $dist_0(m'') < dist_0(tail(m))$.

For the third clause : As m is consistent π_n is winning. By definition of σ_0 we have that $\pi_n = U(\eta_n) \rightarrow CS_U(\eta_n, p, c) \rightarrow CM_U(\eta_n, p, c, S_0, S_1) \rightarrow U?(p, c, S_0, S_1)$ is a path in $\widetilde{G}(\sigma_0)$. Therefore m'' is consistent.

For the fourth clause : As m is consistent we have that π_{n-1} is winning, $(R_0^{n-1}, R_1^{n-1}) \in sel_{K_n}(\pi_{n-2}, q_{n-1}, a_{n-1})$, and there is in $\widetilde{G}(\widetilde{\sigma}_0)$ the path $\pi_{n-1} = U(\eta_{n-1}) \rightarrow CS_U(\eta_{n-1}, q, a_n) \rightarrow CM_U(\eta_{n-1}, q, a_n, R_0^n, S) \rightarrow U?(q, a_n, R_0^n, S')$ and a path from $U?(q, a_n, R_0^n, S')$ to $U(q_n, a_n, R_0^n, R_1^n)$ without passing through a push edge. By definition of $\widetilde{G}$, R_1^n equals either R_0^n or S. By definition of σ_0 we have $p \in R_1^n$. Thus p belongs either to R_0^n or S. Hence $U(p, a_{n-1}, R_1^{n-1}, R)$ is a successor of $CM_U(\eta_{n-1}, q, a_n, R_0^n, S)$. Therefore m'' is consistent. Moreover as in mode U we have by definition of σ_0 that $R = R_1^{n-1}$. It follows that $p \in S$ and the edge $CM_U(\eta_{n-1}, q, a_n, R_0^n, S) \rightarrow U(p, c, R_0^{n-1}, R_1^{n-1})$ is labeled by 1, thus we have $dist_0(m'') < dist_0(tail(m))$.

$\square$

Lemma 62 While playing according to the strategy σ_0 all the memories are consistent.

Proof
From Lemma 61 $\square$

Lemma 63 Suppose that $i < k$ are two even indices of two memories m_i, m_k of the same size such that all the memories in between are of *strictly* bigger size and of same mode. We have always $dist_0(m_k) < dist_0(m_i)$ in mode U. In mode B, if there is no state of priority 0 between positions i and k, and $m_i = (q_i, a, R_0, R_1)\pi_n \dots \pi_1$ then we have $m_k = (q_k, a, R_0, R_1)\pi_n \dots \pi_1$ and $dist_0(m_k) < dist_0(m_i)$. if there is not a visit of a state of priority 0 between associated positions.

Proof
If the final mode is U :

By hypothesis and from Lemma 61 we get that m_{k-2} has the form πm_i for some π. By Lemma 62 m_{k-2} is consistent. Actually by Lemma 61 we have : $dist_0(m_k) < dist_0(tail(m_{k-2})) = dist_0(m_i)$

If the final mode is B :

By hypothesis and Lemma 61 we have $m_{i+2} = B(q_{i+2}, b, S_0, S_1)m_i$. By induction we get that $m_{k-2} = (q_{k-2}, b, S_0, S_1)m_i$ By Lemma 62 m_{k-2} is consistent. From hypothesis and by Lemma 61 we get that $q_k \in S_1$ and by definition of σ_0 we have $m_k = (q_k, a, R_0, R_1)\pi_n \dots \pi_1$ and $dist_0(m_k) < dist_0(m_i)$.

$\square$

Lemma 64 Strategy σ_0 is winning.

Proof
If the play is finite then by Lemma 62 and by definition of σ_0, a state from R is reached. If player 1 changes infinitely often the mode, he looses. Else, every play respecting σ_0 remains from some position in some mode. If the final mode is U then by Lemma 63 the play cannot visit infinitely often configurations of same height of stack. If the final mode is B then by Lemma 63 the play cannot avoid infinitely often positions whose the state is 0-colored.
$\square$

9 Conclusions

In the framework of pushdown games we have introduced a new condition expressing the fact that the stack is unbounded. Although this condition is related to the strict unboundedness condition from the paper of Cachat, Duparc and Thomas [6], we believe it more natural. We have shown nevertheless that player 0 has a winning strategy with one condition iff he has one with the other. This property extends to the case when these conditions are considered in union with a parity condition. It stops being true for conditions of the form $Acc_\Omega \cap Acc_U$.

We have proved that the problem of solving a pushdown game with boolean combinations of Büchi and unboundedness conditions is EXPTIME-complete. Unfortunately we were able to fit just parts of proofs of two cases into the page limit. The omitted constructions are given in the appendix. The complete proofs can be found in [3].

The proofs give strategies that are implementable using a stack. This is useful as such strategies are finitely described, and could be used for instance to define a controller.

We have given methods to decide the winner from a given position, but only from one that has just one letter on the stack. One may be interested in having a uniform version, that is a full description of the set of winning positions. Using techniques of [14] one deduces from our algorithm an alternating automaton recognizing the set of winning positions, which is in fact a regular language. One may also note that this alternating automaton gives a method to define strategies using a stack for any winning position.

Finally, let us comment on the restriction to Büchi and co-Büchi winning conditions. We believe that the method presented here works for all parity conditions and we hope to include the proof in the journal version of this paper. At present we do not have the right notation to describe the general construction in a readable way. Nevertheless we think that the Büchi/co-Büchi case is sufficiently interesting as these kinds of conditions are enough to encode LTL and CTL properties.

References

[1] A. Arnold, A. Vincent, and I. Walukiewicz. Games for synthesis of controlers with partial observation. *Theoretical Computer Science*, 303(1):7–34, 2003.

[2] A. Bouajjani, J. Esparza, and O. Maler. Reachability analysis of pushdown automata: Applications to model checking. In *CONCUR'97*, volume 1243 of *Lecture Notes in Computer Science*, pages 135–150, 1997.

[3] A. Bouquet, O. Serre, and I. Walukiewicz. Pushdown games with the unboundedness and regular conditions: full version with complete proofs. http://www.labri.fr/∼igw/publications.html.

[4] T. Cachat. Symbolic strategy synthesis for games on pushdown graphs. In *Proceedings of the 29th International Colloquium on Automata, Languages, and Programming*, volume 2380 of *Lecture Notes in Computer Science*, pages 704–715. Springer Verlag, 2002.

[5] T. Cachat. Uniform solution of parity games on prefix-recognizable graphs. In Antonin Kucera and Richard Mayr, editors, *Proceedings of the 4th International Workshop on Verification of Infinite-State Systems*, volume 68 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, 2002.

[6] T. Cachat, J. Duparc, and W. Thomas. Solving pushdown games with a sigma-3 winning condition. In *Proceedings of the 11th Annual Conference of the European Association for Computer Science Logic, CSL 2002*, volume 2471 of *Lecture Notes in Computer Science*, pages 322–336, 2002.

[7] E. A. Emerson, C.S. Jutla, and A.P. Sistla. On model-checking for fragments of μ-calculus. In *CAV'93*, volume 697 of *LNCS*, pages 385–396, 1993.

[8] E. Allen Emerson and Charanjit S. Jutla. Tree automata, mu-calculus and determinacy. In *Proc. FOCS 91*, pages 368–377, 1991.

[9] E.A. Emerson, C.S. Jutla, and A.P. Sistla. On model-checking for the mu-calculus and its fragments. *Theoretical Computer Science*, 258(1-2):491–522, 2001.

[10] J. Esparza and A. Podelski. Efficient algorithms for pre star and post star on interprocedural parallel flow graphs. In *POPL'00: Principles of Programming Languages*, 2000.

[11] O. Kupferman and M.Y. Vardi. An automata-theoretic approach to reasoning about infinite-state systems. In *Proceedings of CAV'00*, volume 1855 of *Lecture Notes in Computer Science*, pages 36–52. Springer Verlag, 2000.

[12] D.E. Muller and P.E. Schupp. The theory of ends, pushdown automata and second-order logic. *Theoretical Computer Science*, 37:51–75, 1985.

[13] D.E. Muller and P.E. Schupp. Alternating automata on infinite trees. *Theoretical Computer Science*, 54:267–276, 1987.

[14] O. Serre. Note on winning positions on pushdown games with ω-regular conditions. *Information Processing Letters*, 85:285–291, 2003.

[15] Wolfgang Thomas. Automata on infinite objects. In J. van Leeuven, editor, *Handbook of Theoretical Computer Science Vol.B*, pages 133–192. Elsevier, 1990.

[16] Wolfgang Thomas. On the synthesis of strategies in infinite games. In *STACS '95*, volume 900 of *Lecture Notes in Computer Science*, pages 1–13, 1995.

[17] Wolfgang Thomas. Languages, automata, and logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3. Springer-Verlag, 1997.

[18] Igor Walukiewicz. Pushdown processes: Games and model checking. In *CAV'96*, volume 1102 of *Lecture Notes in Computer Science*, pages 62–74, 1996. Full version to appear in Information and Computation.

[19] Igor Walukiewicz. Pushdown processes: Games and model checking. *Information and Computation*, 164(2):234–263, 2001.

[20] Wieslaw Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200:135–183, 1998.