Playing Pushdown Parity Games in a Hurry (Extended Abstract)

Wladimir Fridman
Chair of Computer Science 7
RWTH Aachen University
Aachen, Germany
fridman@automata.rwth-aachen.de

Martin Zimmermann
Institute of Informatics
University of Warsaw
Warsaw, Poland
zimmermann@mimuw.edu.pl

We continue the investigation of finite-duration variants of infinite-duration games by extending known results for games played on finite graphs to those played on infinite ones. In particular, we establish an equivalence between pushdown parity games and a finite-duration variant. This allows to determine the winner of a pushdown parity game by solving a reachability game on a finite tree.

1 Introduction

Infinite two-player games on graphs are a powerful tool to model, verify, and synthesize open reactive systems and are closely related to fixed-point logics. The winner of a play in such a game typically emerges only after completing the whole (infinite) play. Despite this, McNaughton became interested in playing infinite games in finite time, motivated by his belief that "infinite games might have an interest for casual living room recreation" [3].

As playing infinitely long is impossible for human players, McNaughton introduced scoring functions for Muller games, a certain type of infinite game. Each of these functions is associated to one of the two players, so it makes sense to talk about the scores of a player. The scoring functions are updated after every move and describe the progress a player has made towards winning the play. However, as soon as a scoring function reaches its predefined threshold, the game is stopped and the player whose score reached its threshold first is declared to win this (now finite) play.

By applying finite-state determinacy of Muller games, McNaughton showed that a Muller game and a finite-duration variant with a factorial threshold score have the same winner. Thus, the winner of a Muller game can be determined by solving a finite reachability game, which is much simpler to solve, albeit doubly-exponentially larger than the original Muller game. This result was improved by showing that the finite-duration game with threshold three always has the same winner as the original Muller game [2] and by a (score-based) reduction from a Muller game to a safety game whose solution not only yields the winner of the Muller game, but also a new kind of memory structure implementing a permissive winning strategy [5], i.e., the most general non-deterministic strategy that prevents the losing player from reaching a certain score.

In this work, we begin to extend these results to parity games played on configuration graphs of pushdown systems. For a (min-) parity game on a *finite* game graph there is a straightforward way to define a finite-duration variant: let $|V|_c$ denote the number of vertices colored by c. Then, a positional winning strategy for Player $i \in \{0,1\}$ does not visit $|V|_c + 1$ vertices of color c with parity $1-i$ without visiting a vertex of smaller color in between. This condition can be expressed using scoring functions Sc_c for every color c that count the number of vertices of color c visited since the last visit of a vertex of color $c' < c$. Due to positional determinacy of parity games, the following finite-duration game has the

Submitted to:
AISS 2012

same winner as the original parity game: a play is stopped as soon as some scoring function Sc_c reaches value $|V|_c + 1$ for the first time and Player i is declared to be the winner, if the parity of c is i.

However, this argument is not applicable to infinite game graphs, since some $|V|_c$ is infinite in such a graph. We exploit the intrinsic structure of the game graph induced by the pushdown system to define stair-score functions $\mathsf{StairSc}_c$ for every color c and show the equivalence between a pushdown parity game and the finite-duration version, when played up to an exponential threshold stair-score (in the size of the pushdown system). This result shows how to determine the winner of an infinite game on an infinite game graph by solving a reachability game on a finite tree. We complement this by giving a lower bound on the threshold stair-score that always yields the same winner, which is exponential in the cubic root of the size of the underlying pushdown system.

2 Preliminaries

We consider parity games of the form (G,col), where $G = (V,V_0,V_1,E,v_{\mathrm{in}})$ is a game graph with a (possibly countably infinite) directed graph (V,E), a partition $V_0 \cup V_1$ of V and an initial vertex $v_{\mathrm{in}} \in V$. Furthermore, $\mathrm{col}\colon V \to [n] = \{0,\dots,n-1\}$ for some $n \in \mathbb{N}$ is a coloring function. A play in $\mathscr{G}$ is an infinite sequence $\rho \in V^\omega$ such that $\rho(0) = v_{\mathrm{in}}$ and $(\rho(n),\rho(n+1)) \in E$ for every $n \in \mathbb{N}$. Such a play is winning for Player $i \in \{0,1\}$ if the parity of the minimal color that is visited infinitely often is i.

A strategy for Player i is a function $\sigma\colon V^*V_i \to V$ such that $(\mathsf{last}(w),\sigma(w)) \in E$ for every $w \in V^*V_i$. A play ρ is consistent with σ if $\rho(n+1) = \sigma(\rho(0)\cdots\rho(n))$ for every $n \in \mathbb{N}$ with $\rho(n) \in V_i$. A strategy σ is a winning strategy for Player i if every play ρ that is consistent with σ is winning for Player i. We say that Player i wins $\mathscr{G}$ if he has a winning strategy. Every parity game is won by one of the players [1, 4].

The game graphs we consider are induced by pushdown systems (PDS) $\mathscr{P} = (Q,\Gamma,\Delta,q_{\mathrm{in}})$, where Q is a finite set of states with initial state $q_{\mathrm{in}} \in Q$, where Γ is a stack alphabet with initial stack symbol $\bot \notin \Gamma$, which can neither be written nor deleted from the stack, and a transition relation $\Delta \subseteq Q \times \Gamma_\bot \times Q \times \Gamma_\bot^{\leq 2}$, where $\Gamma_\bot = \Gamma \cup \{\bot\}$. A stack content is a word from $\Gamma^*\bot$ where the leftmost symbol is assumed to be the top of the stack. A configuration is a pair (q,γ) consisting of a state $q \in Q$ and a stack content $\gamma \in \Gamma^*\bot$. The stack height of a configuration (q,γ) is defined by $\mathrm{sh}(q,\gamma) = |\gamma| - 1$. Furthermore, we write $(q,\gamma) \vdash (q',\gamma')$ if there exists $(q,\gamma(0),q',\alpha) \in \Delta$ and $\gamma' = \alpha\gamma(1)\cdots\gamma(|\gamma|-1)$.

A PDS $\mathscr{P}$ with a partition $Q_0 \cup Q_1$ of Q and a coloring function $\mathrm{col}\colon Q \to [n]$ of Q induce a parity game whose vertices are configurations of $\mathscr{P}$, the edge relation is $\vdash$, the partition of the vertices into V_0 and V_1 is induced by $Q_0 \cup Q_1$, the initial vertex is $v_{\mathrm{in}} = (q_{\mathrm{in}},\bot)$, and the color of a vertex is the color of its state. We refer to such a game as a pushdown game.

3 Finite-Time Pushdown Games

Let (G,col) be a pushdown game. For a path through a pushdown graph, a configuration is said to be a stair configuration if no subsequent configuration of strictly smaller stack height exists in this path. Note that the last configuration of a finite path is always a stair. Let $\mathsf{reset}(v) = \varepsilon$ and $\mathsf{lastBump}(v) = v$ for $v \in V$ and for $w = w(0)\cdots w(r)$ with $r \geq 1$, let $\mathsf{reset}(w) = w(0)\cdots w(l)$ and $\mathsf{lastBump}(w) = w(l+1)\cdots w(r)$ where l is the greatest position such that $\mathrm{sh}(w(l)) \leq \mathrm{sh}(w(r))$ and $l \neq r$, i.e., l is the second largest stair of w. We use the decomposition of w into blocks induced by its stairs to define a scoring function for pushdown games. To this end, let $\mathsf{MinCol}(w) = \min\{\mathrm{col}(w(i)) \mid 0 \leq i < |w|\}$.

Definition 1 (Stair-scoring function)**.** *For every color $c \in [n]$, define the function* $\mathsf{StairSc}_c \colon V^* \to \mathbb{N}$ *by* $\mathsf{StairSc}_c(\varepsilon) = 0$ *and for $w \in V^+$ by*

$$\mathsf{StairSc}_c(w) = \begin{cases} \mathsf{StairSc}_c(\mathsf{reset}(w)) & \textit{if } \mathsf{MinCol}(\mathsf{lastBump}(w)) > c, \\ \mathsf{StairSc}_c(\mathsf{reset}(w)) + 1 & \textit{if } \mathsf{MinCol}(\mathsf{lastBump}(w)) = c, \\ 0 & \textit{if } \mathsf{MinCol}(\mathsf{lastBump}(w)) < c. \end{cases}$$

For $c \in [n]$, the function $\mathsf{MaxStairSc}_c \colon V^* \to \mathbb{N}$ *is defined by* $\mathsf{MaxStairSc}_c(w) = \max_{w' \sqsubseteq w} \mathsf{StairSc}_c(w')$.

A finite-time pushdown game (G, col, k) consists of a pushdown game graph G, a coloring function col and a threshold $k \in \mathbb{N}$. A play in (G, col, k) is a finite path $w = w(0) \cdots w(r) \in V^*$ with $w(0) = v_{\mathrm{in}}$ such that $\mathsf{MaxStairSc}_c(w) = k$ for some $c \in [n]$, and $\mathsf{MaxStairSc}_c(w(0) \cdots w(r-1)) < k$ for all $c \in [n]$. The play w is winning for Player i if the parity of c is i. The notions of (winning) strategies are defined as usual. Every threshold score k is eventually reached by some stair-score function after at most 2^{k^n} moves and in every move, exactly one stair-score function is increased, i.e., a finite-time pushdown game is a finite zero-sum game. For a pushdown game $\mathscr{G} = (G, \mathrm{col})$ induced by a PDS $\mathscr{P} = (Q, \Gamma, \Delta, q_{\mathrm{in}})$ and a coloring function $\mathrm{col} \colon Q \to [n]$, let $k_{\mathscr{G}} = |Q| \cdot |\Gamma| \cdot 2^{|Q| \cdot n} \cdot n$.

Theorem 2. *Let $\mathscr{G} = (G, \mathrm{col})$ be a pushdown game and let $\mathscr{G}_k = (G, \mathrm{col}, k)$ be the corresponding finite-time pushdown game with threshold k. For every $k > k_{\mathscr{G}}$, Player i wins $\mathscr{G}$ if and only if he wins $\mathscr{G}_k$.*

This is proved by relating stair-scores in $\mathscr{G}$ to scores (as defined in the introduction) in the corresponding finite parity game obtained by Walukiewicz's reduction from pushdown games to finite parity games [6].

4 Lower Bounds

We have established the equivalence between pushdown games and corresponding finite-time pushdown games for an exponential threshold. Here, we present an (almost) matching lower bound on the threshold.

Theorem 3. *There are a family of pushdown games (G_n, col_n) and thresholds k_n exponential in the cubic root of the size of the underlying PDS such that for every $n > 0$, Player 0 wins the pushdown game (G_n, col_n) but for every $k \leq k_n$, Player 1 wins the finite-time pushdown game (G_n, col_n, k).*

Consider the pushdown game (G, col) depicted in Figure 1. The double-lined vertices are colored by 0, all other vertices by 1. A play in the game proceeds as follows. Player 0 picks a natural number $x > 0$ by moving the token to the configuration $(q_\square, A^x \bot)$. If she fails to do so by staying in state q_{in} ad infinitum she loses, since it is colored by 1. At $(q_\square, A^x \bot)$ Player 1 picks a modulo counter $i \in \{2, 3\}$ by moving the token to $(p_0^i, A^x \bot)$. From this configuration, a single path emanates, i.e., there is only one way to continue the play. Player 0 wins if and only if $x \bmod i = 0$. Hence, Player 0 has a winning strategy for this game by moving the token to some non-zero multiple of 6, i.e., Player 0 wins (G, col).

Let $k \leq 6$. If Player 0 reaches $(q_{\mathrm{in}}, A^{k-1} \bot)$, she loses in the finite-time pushdown game (G, col, k), since then Player 1 obtains stair-score k. On the other hand, if she moves the token to a configuration $(q_\square, A^j \bot)$ for some $j \leq k-1$, then there is an $i \in \{2, 3\}$ such that $j \bmod i \neq 0$, as $j < 6$. Hence, assume Player 1 moves the token to $(p_i^0, A^j \bot)$. Then, the play ends in a self-loop at some p_i^m for some $m \neq 0$. The path w from $(q_{\mathrm{in}}, \bot)$ to $(p_i^m, \bot)$ via $(q_\square, A^j)$ satisfies $\mathsf{MaxStairSc}_0(w) \leq j$. Since p_i^m is colored by 1, the scores of Player 0 are never increased while using the self-loop at $(p_i^m, \bot)$. Thus, her scores never reach the threshold k. Hence, Player 1 is the first to reach this threshold, i.e., Player 1 wins (G, col, k).

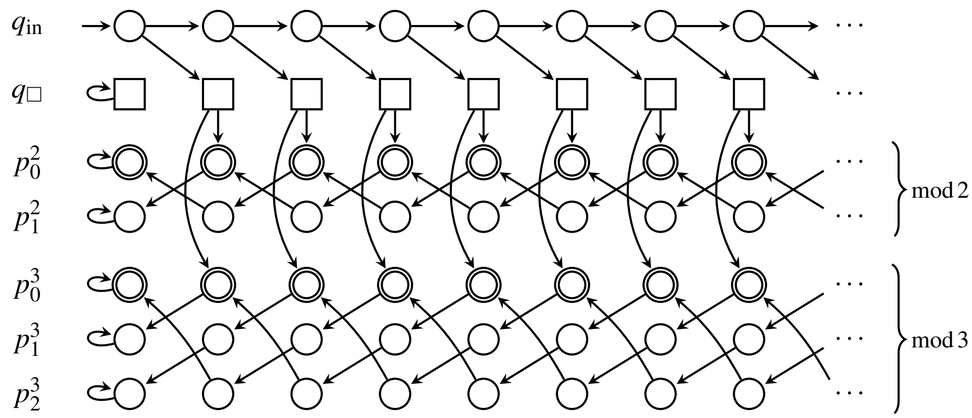

Figure 1: A Pushdown Game (G, col) for Theorem 3.

To prove Theorem 3, we define (G_n, col_n) to contain modulo-counters for the first n prime numbers, i.e., the example game in Figure 1 is (G_2, col_2). The size of the PDS underlying (G_n, col_n) is cubic in n, while in order to win, Player 0 has to reach a stack height k_n, which is defined to be the product of the first n prime numbers. But thereby, Player 1 is the first to reach a stair-score of k_n, which is exponential in the cubic root of the size of the underlying PDS.

5 Conclusion

This work transfers results on playing infinite games in finite time obtained for games on finite game graphs to infinite graphs. In ongoing work, we investigate if and how a winning strategy for the finite safety game, in which Player 0 wins if and only if he prevents his opponent from reaching an exponential stair-score can be turned into a (permissive) winning strategy for the original pushdown game. Note that the winner of these two games is equal. On the other hand, our results could be extended by considering other classes of infinite graphs having an intrinsic structure, e.g., configuration graphs of higher-order pushdown systems. Finally, there is a small gap between the upper and lower bound on the threshold score that always yields an equivalent finite-duration pushdown game, which remains to be closed.

References

[1] E. Allen Emerson & Charanjit S. Jutla (1991): *Tree Automata, Mu-Calculus and Determinacy (Extended Abstract)*. In: *FOCS*, IEEE Computer Society, pp. 368–377.

[2] John Fearnley & Martin Zimmermann (2010): *Playing Muller Games in a Hurry*. In Angelo Montanari, Margherita Napoli & Mimmo Parente, editors: *GANDALF*, *EPTCS* 25, pp. 146–161.

[3] Robert McNaughton (2000): *Playing Infinite Games in Finite Time*. In Arto Salomaa, Derick Wood & Sheng Yu, editors: *A Half-Century of Automata Theory*, World Scientific, pp. 73–91.

[4] Andrzej W. Mostowski (1991): *Games with forbidden positions*. Technical Report 78, University of Gdańsk.

[5] Daniel Neider, Roman Rabinovich & Martin Zimmermann (2011): *Solving Muller Games via Safety Games*. Technical Report AIB-2011-14, RWTH Aachen University.

[6] Igor Walukiewicz (1996): *Pushdown Processes: Games and Model Checking*. In Rajeev Alur & Thomas A. Henzinger, editors: *CAV*, *Lecture Notes in Computer Science* 1102, Springer, pp. 62–74.