Six Types of Pushdown Games

Wladimir Fridman

RWTH Aachen University, Germany

Second GASICS Meeting, October 2009

Joint work with Wolfgang Thomas

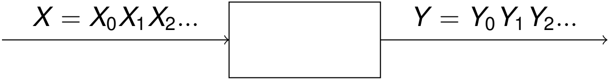
Outline

1. Motivation and Background

2. Six Types of Pushdown Games

3. Main Result

Church's Problem of Controller Synthesis



input $X \in \Sigma_1^\omega$, output $Y \in \Sigma_2^\omega$, $\binom{X}{Y} \in (\Sigma_1 \times \Sigma_2)^\omega = \Sigma^\omega$

given: specification $L \subseteq \Sigma^\omega$

question: is there an automaton (transducer), that transforms every input $X \in \Sigma_1^\omega$ letter-by-letter into an output $Y \in \Sigma_2^\omega$, such that $\binom{X}{Y} \in L$?
if yes - construction!

Formulation in Terms of Infinite 2-Player Game

- Player 1 and Player 2 pick letters in alternation - $X_i \in \Sigma_1$ resp. $Y_i \in \Sigma_2$

- Play $\binom{X}{Y} = \binom{X_0}{Y_0}\binom{X_1}{Y_1}\binom{X_2}{Y_2}\ldots$

- Winning condition given by $L \subseteq \Sigma^\omega$
  - Player 2 wins $\binom{X}{Y}$ if $\binom{X}{Y} \in L$
  - Player 1 wins $\binom{X}{Y}$ if $\binom{X}{Y} \notin L$

- strategy for Player 1 is a function $\binom{X_0}{Y_0}\binom{X_1}{Y_1}\ldots\binom{X_k}{Y_k} \mapsto \Sigma_1$

- strategy for Player 2 is a function $\binom{X_0}{Y_0}\binom{X_1}{Y_1}\ldots\binom{X_k}{*} \mapsto \Sigma_2$

Task: find winner and winning strategy!

Fundamental Result

Theorem (Büchi - Landweber 1969)

- *For each MSO-definable game either Player 1 or Player 2 has a finite-state winning strategy,*
- *one can compute the winner and a finite-state machine realizing a winning strategy.*

winning condition:	winning strategy:
regular ω-language L	finite automaton (with output)

connection between winning conditions and winning strategies

Pushdown Game:

- winning condition: ω-context-free language L
- game on pushdown graph with parity condition

Pushdown Strategy:

- PDA with output
- reads opponent's inputs, outputs player's next letter to pick

Pushdown Automaton

Definition (PDA)

A pushdown automaton $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_{in}, \perp)$

- Q - finite set of states, q_{in} - initial state
- Σ - input alphabet
- Γ - pushdown alphabet, $\perp$ - initial pushdown symbol
- $\delta \colon (Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma) \to \mathcal{P}(Q \times \Gamma^*)$

Definition (Pushdown graph)

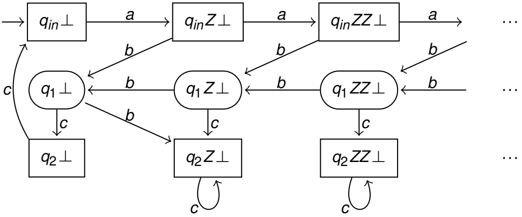
For a PDA $\mathcal{A}$ a pushdown graph is $G(\mathcal{A}) = (V, E)$

- $V = \{(q, \gamma) \mid q \in Q \text{ and } \gamma \in (\Gamma \setminus \{\perp\})^* \perp\}$
- $E \subseteq V \times (\Sigma \cup \{\varepsilon\}) \times V$, $(q, Z\gamma) \xrightarrow{a} (q', \beta\gamma)$ if $(q, \beta) \in \delta(q, a, Z)$

Pushdown Game: Example

Pushdown game

- $Q_1 = \{q_{in}, q_2\}$, $Q_2 = \{q_1\}$
- $\Sigma = \{a, b, c\}$, $\Gamma = \{Z, \perp\}$



$col(q_{in}) = 0$

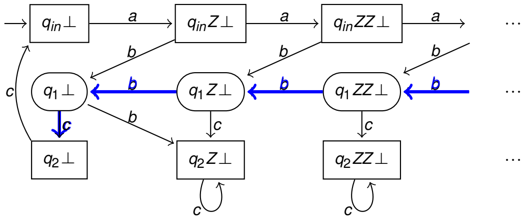
$col(q_1) = 1$

$col(q_2) = 1$

Pushdown Game: Example

Pushdown game

- $Q_1 = \{q_{in}, q_2\}$, $Q_2 = \{q_1\}$
- $\Sigma = \{a, b, c\}$, $\Gamma = \{Z, \perp\}$



$col(q_{in}) = 0$

$col(q_1) = 1$

$col(q_2) = 1$

Winning strategy for Player 2

- pick "*b*" if the prefix contains more *a*'s than *b*'s
- pick "*c*" if the prefix contains equal number of *a*'s and *b*'s,

Pushdown Games

Remark (Finkel 2001)

For nondeterministic context-free languages $L \in CFL_\omega$ it is undecidable to determine which player has a winning strategy in the Gale-Stewart game defined by L.

- proof uses undecidability of UNIVERSALITY(CFL_ω)
- conclusion: Church's Problem for the class of CFL_ω is undecidable.

Theorem (Walukiewicz 1996)

Parity games on deterministic pushdown graphs are determined with deterministic pushdown winning strategies.

- proof idea: reduction to parity game on finite game graph

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
3. deterministic stair pushdown automaton (StDPDA)
4. realtime DPDA
5. deterministic one-counter automaton (D1CA)
6. deterministic blind one-counter automaton (DB1CA)

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
   - stack height controlled by input
3. deterministic stair pushdown automaton (StDPDA)
4. realtime DPDA
5. deterministic one-counter automaton (D1CA)
6. deterministic blind one-counter automaton (DB1CA)

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
   - stack height controlled by input
3. deterministic stair pushdown automaton (StDPDA)
   - evaluation of parity condition only at special positions
4. realtime DPDA
5. deterministic one-counter automaton (D1CA)
6. deterministic blind one-counter automaton (DB1CA)

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
   - stack height controlled by input
3. deterministic stair pushdown automaton (StDPDA)
   - evaluation of parity condition only at special positions
4. realtime DPDA
   - no ε-transitions
5. deterministic one-counter automaton (D1CA)
6. deterministic blind one-counter automaton (DB1CA)

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
   - stack height controlled by input
3. deterministic stair pushdown automaton (StDPDA)
   - evaluation of parity condition only at special positions
4. realtime DPDA
   - no ε-transitions
5. deterministic one-counter automaton (D1CA)
   - only one pushdown symbol Z (except $\perp$)
6. deterministic blind one-counter automaton (DB1CA)

Types of Pushdown Automata

1. deterministic pushdown automaton (DPDA)
2. visibly pushdown automaton (VPA)
   - stack height controlled by input
3. deterministic stair pushdown automaton (StDPDA)
   - evaluation of parity condition only at special positions
4. realtime DPDA
   - no ε-transitions
5. deterministic one-counter automaton (D1CA)
   - only one pushdown symbol Z (except $\perp$)
6. deterministic blind one-counter automaton (DB1CA)
   - if $\delta(q, a, \perp) = (q', Z^n\perp)$ then $\delta(q, a, Z) = (q', Z^n Z)$

Main Result

Theorem

tight connection between winning conditions and winning strategies:

1. *$DCFL_\omega$-games are determined with DCFL w. s.*
2. *VPL_ω-games are determined with VPL w. s.*
3. *$StDCFL_\omega$-games are determined with DCFL w. s.*
4. *realtime-$DCFL_\omega$-games are determined with realtime-DCFL w. s.*
5. *$D1CL_\omega$-games are determined with D1CL w. s.*

no winning strategies of the corresponding format

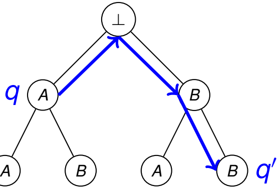
6. *for $DB1CL_\omega$-games DB1CL winning strategies do NOT suffice*

Cases 1 - 5

Proof idea: refinement of the following approach (M. Y. Vardi)

1. construction of an alternating two-way tree automaton $\mathcal{A}$, which simulates the pushdown transitions on the full $(\Gamma \setminus \{\bot\})$-tree

$$\delta(q, a, A) = (q', BB)$$



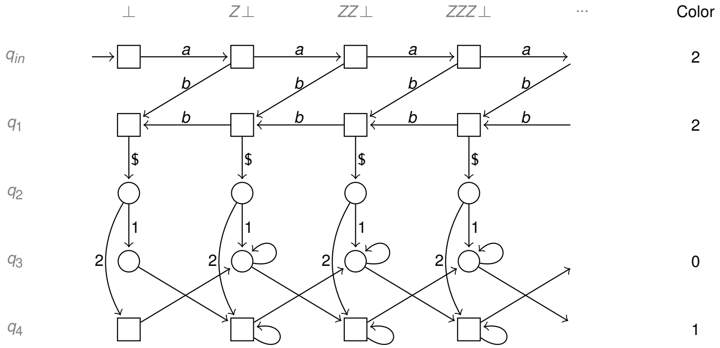
2. construction of an equivalent one-way parity tree automaton $\mathcal{A}'$, Player 2 has a winning strategy iff $L(\mathcal{A}') \neq \emptyset$,
3. deduction of a strategy in the respective format from the deterministic word automaton generating a regular tree in $L(\mathcal{A}')$

Case 6 (Blind 1-counter)

Claim: for $DB1CL_\omega$-games DB1CL winning strategies do NOT suffice.

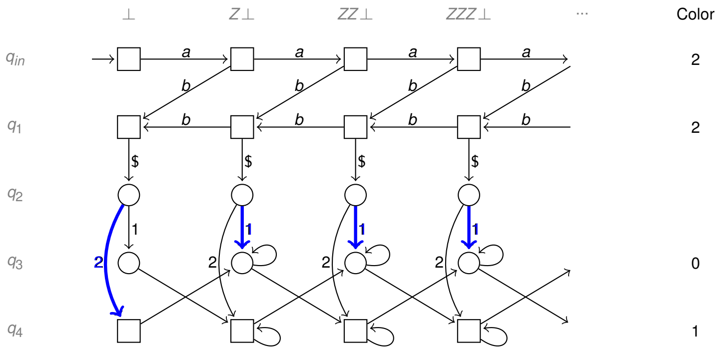
- define a $DB1CL_\omega$-game $\mathcal{G}$
- there is a pushdown winning strategy for Player 2 in $\mathcal{G}$
- there is no DB1CL winning strategy for Player 2 in $\mathcal{G}$

Blind 1-counter game $\mathcal{G}$



Case 6 (Blind 1-counter)

Blind 1-counter game $\mathcal{G}$



Pushdown winning strategy for Player 2:

- pick "1" if the prefix constructed by Player 1 is $a^n b^m \$$, $m < n$
- pick "2" if the prefix constructed by Player 1 is $a^n b^n \$$,

Case 6 (Blind 1-counter)

Lemma

$\{a^n b^n \$ | n > 0\}$ *is not accepted by any DB1CA.*

Proof: simple exercise.

There is no DB1CL winning strategy for $\mathcal{G}$

Every strategy has to decide if the prefix chosen by Player 1 is

- in $\{a^n b^n \$ | n > 0\}$ or
- in $\{a^n b^m \$ | n, m > 0 \text{ and } m < n\}$

Summary

- connection between context-free winning conditions and winning strategies in six special cases
- in 5 cases (DPDA, VPA, StDPDA, realtime-DPDA, D1CA) strategies are implementable by the corresponding type of pushdown machine
- for blind 1-counter games blind 1-counter strategies do not suffice

Outlook

- open question: general analysis
  What are the abstract reasons for the distinction between cases 1-5 and case 6?