Determinacy strength of infinite games in ω-languages recognized by variations of pushdown automata

Wenjuan Li
jointly with Prof. Kazuyuki Tanaka

Mathematical Institute, Tohoku University

Sep. 16, 2016

Workshop on Mathematical Logic and its Applications
JSPS Core-to-Core Program
Kyoto University, Kyoto

- Infinite games
  - Gale-Stewart game $\mathbb{G}(X)$, where $X(\subseteq A^{\omega})$ is a winning set for player I
- Determinacy
  - With the usual convention, $\mathcal{C}$-Det denotes that "A Gale-Stewart game $\mathbb{G}(X)$ is determined (one of the two players has a winning strategy), if X is contained in the class $\mathcal{C}$".
- ω-languages accepted by automata
  - $L(\mathcal{M})$, where $\mathcal{M}$ is some kind of automata

Question

If the winning sets are effectively given, i.e., winning sets are accepted by some kind of automata, how is the determinacy strength of such games?

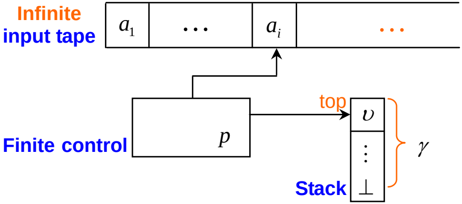
Outline

1. Introduction
   - Pushdown automata, visibly pushdown automata, etc.

2. Determinacy strength and ω-languages
   - $\mathbf{2DVPL}_\omega$, r-$\mathbf{PDL}_\omega$, $\mathbf{PDL}_\omega$, etc.

3. Ongoing and future works

Pushdown automata on infinite words (ω-PDA)



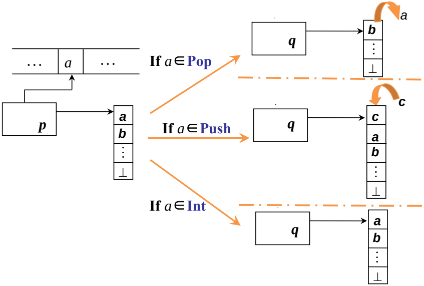
A run on $a_1 \ldots a_n \ldots$ is an infinite sequence of configurations:

$$(q_{in}, \perp) \xrightarrow{a_1 \text{ or } \varepsilon} (q_1, \gamma_1) \ldots \xrightarrow{a_n \text{ or } \varepsilon} (q_s, \gamma_s) \xrightarrow{a_{n+1} \text{ or } \varepsilon} \ldots$$

An infinite word $a_1 \ldots a_n \ldots \in A^\omega$ is accepted by a Büchi pushdown automaton if there exists a run visiting a state in F infinitely many times.

Visibly Pushdown Automata

- For (1-stack) visibly pushdown automata (2VPA), the *alphabet* A is partitioned into **Push**, **Pop**, **Int**. The *transitions* are as follows.



- For 2-stack visibly pushdown automata (2VPA), the *alphabet* A is partitioned into $\mathbf{Push}_1$, $\mathbf{Pop}_1$, $\mathbf{Push}_2$, $\mathbf{Pop}_2$, **Int**.

Example

Given $A = (\{a\}, \{\overline{a}\}, \{b\}, \{\overline{b}\}, \emptyset)$, the language $\{(ab)^n \overline{a}^n \overline{b}^n \mid n \in \mathbb{N}\}$, is recognized by a deterministic 2-stack visibly pushdown automaton (2DVPA).

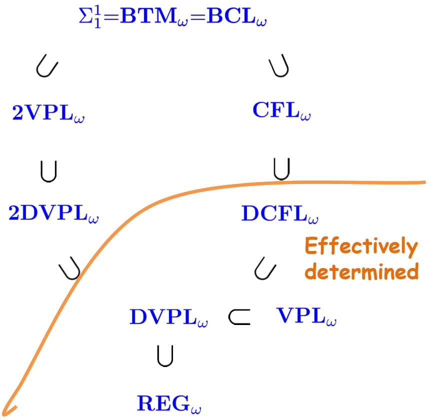
Outline

1. Introduction
   - Pushdown automata, visibly pushdown automata, etc.

2. Determinacy strength and ω-languages
   - $\mathbf{2DVPL}_\omega$, r-$\mathbf{PDL}_\omega$, $\mathbf{PDL}_\omega$, etc.

3. Ongoing and future works

Determinacy strength of infinite games in deterministic 2-stack visibly ω-languages



REG$_\omega$: ω-regular lang. (FA)

CFL$_\omega$: context free ω-lang. (PDA)

DCFL$_\omega$: deterministic **CFL**$_\omega$(DPDA)

VPL$_\omega$: visibly pushd. ω-lang. (VPA)

BTM$_\omega$: ω-lang. by Büchi Turing machine

Note that these languages are defined with Büchi or Muller condition.

Question

How about other acceptance conditions of lower levels?

In this talk

we concentrates on determinacy strength of infinite games specified by nondeterministic pushdown automata and variants of it with various acceptance conditions, e.g., safety, reachability, co-Büchi conditions.

Acceptance conditions of infinite words

Safety (or Π_1) accepance condition

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r = (q_i, \gamma_i)_{i \geq 1} \text{ of } \mathcal{M} \text{ on } \alpha \text{ such that } \forall i, q_i \in F\}.$$

Reachability (or Σ_1) acceptance condition

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r = (q_i, \gamma_i)_{i \geq 1} \text{ of } \mathcal{M} \text{ on } \alpha \text{ such that } \exists i, q_i \in F\}.$$

Let $\mathsf{Inf}(r)$ be the set of states that are visited infinite many times during the run r.

Co-Büchi (or Σ_2) acceptance condition

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r = (q_i, \gamma_i)_{i \geq 1} \text{ of } \mathcal{M} \text{ on } \alpha \text{ such that } \mathsf{Inf}(r) \subseteq F\}.$$

Acceptance conditions of infinite words (continued)

$(\Sigma_1 \wedge \Pi_1)$ acceptance condition

There exist F_r, $F_s \subset Q$,

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r = (q_i, \gamma_i)_{i \geq 1} \text{ of } \mathcal{M} \text{ on } \alpha \\ \text{such that } \exists i, q_i \in F_r \wedge \forall i, q_i \in F_s\}.$$

$(\Sigma_1 \vee \Pi_1)$ acceptance condition

There exist F_r, $F_s \subset Q$,

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r = (q_i, \gamma_i)_{i \geq 1} \text{ of } \mathcal{M} \text{ on } \alpha \\ \text{such that } \exists i, q_i \in F_r \vee \forall i, q_i \in F_s\}.$$

Acceptance conditions of infinite words (continued)

Δ_2 acceptance condition

There exist F_b, $F_c \subset Q$,

$$L(\mathcal{M}) = \{\alpha \in \Sigma^\omega | \text{there is a run } r \text{ of } \mathcal{M} \text{ on } \alpha \text{ such that } \mathsf{Inf}(r) \cap F_b \neq \emptyset\}$$

$$= \{\alpha \in \Sigma^\omega | \text{there is a run } r \text{ of } \mathcal{M} \text{ on } \alpha \text{ such that } \mathsf{Inf}(r) \subset F_c\}.$$

Various acceptance conditions of ω-2DVPA

- We denote the ω-languages accepted by ω-2DVPA with different acceptance conditions as follows.
- ω-languages accepted by deterministic Turing machines with safety (resp., reachabiliy, co-Büchi, Büchi) condition is the collection of all arithmetical Π^0_1-sets (respectively, Σ^0_1-sets, Σ^0_2-sets, Π^0_2-sets).

Acceptance conditions	Subclass of $\mathbf{2DVPL}_\omega$	
Reachability	$\mathbf{2DVPL}_\omega(\Sigma_1)$	$\subseteq \Sigma^0_1$
Safety	$\mathbf{2DVPL}_\omega(\Pi_1)$	$\subseteq \Pi^0_1$
Co-Büchi	$\mathbf{2DVPL}_\omega(\Sigma_2)$	$\subseteq \Sigma^0_2$
Büchi	$\mathbf{2DVPL}_\omega(\Pi_2)$	$\subseteq \Pi^0_2$

Similarly, by $\mathbf{2DVPL}_\omega(\mathcal{C})$ we denote the ω-languages accepted by deterministic 2-stack visibly pushdown automata with an acceptance condition $\mathcal{C}$.

Theorem

There exists an infinite game in $2\mathbf{DVPL}_{\omega}(\Sigma_1 \wedge \Pi_1)$ *with only* Σ^0_1*-hard winning strategies.*

Proof.

- Let $\mathcal{R}$ be a universal 2-counter automaton.
- We construct a game $\mathbb{G}_{\mathcal{R}}$ such that the halting problem of $\mathcal{R}$ is computable in any winning strategies of player II, while player I has no winning strategy, and moreover the winning set for player II is accepted by a deterministic 2-stack visibly pushdown automaton with a $\Sigma_1 \vee \Pi_1$ acceptance condition.

Recall 2-counter automata

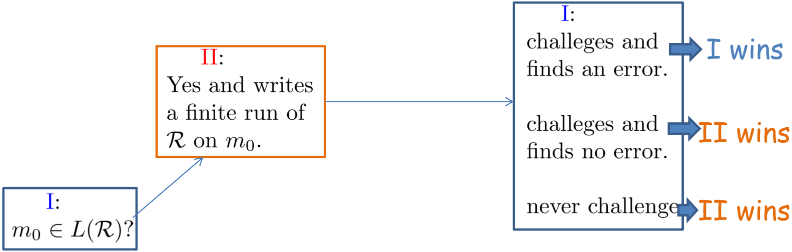
- ✓ A 2-counter automaton can be seen as a restricted 2-stack pushdown automaton with just one symbol for each stack: the number of the symbols in a stack is expresses as a nonnegative integer in a counter.
- ✓ The input is a natural number m which is initially store in one of the counter.
- ✓ By the current state and the tests results on whether each counter is zero or not, the automaton goes to next state and do operations on the two counters by increasing the counter(s) by 1, or decreasing the counter(s) by 1 if the counter is not zero.
- ✓ It is known that a (deterministic) 2-counter automaton, is equivalent to a Turing machine. Thus the halting problem for a certain (universal deterministic) 2-counter automaton is Σ_1^0-complete.

Recall 2-counter automata (continued)

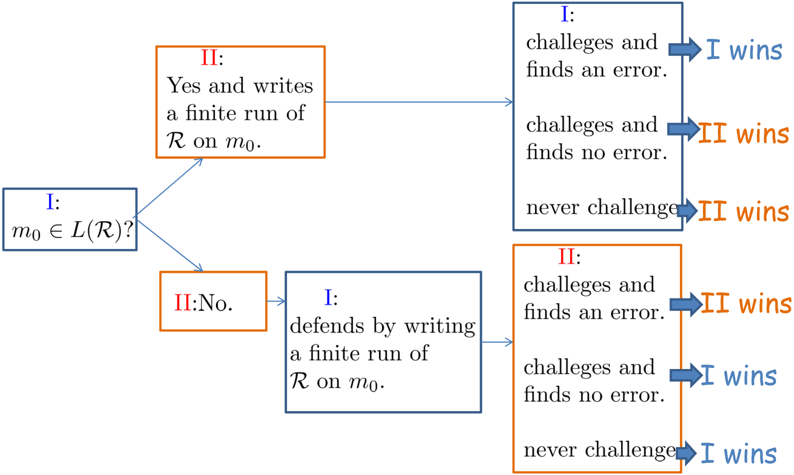
- A configuration (q, m, n) of a 2-counter automaton $\mathcal{R}$ is coded as $qa^m b^n$, where $q \in Q$, and m, n are non-negative integers in the two counters.
- A run for a natural number m on $\mathcal{R}$: $q_{\text{in}} a^{m_0} b^{n_0} \mapsto_{\mathcal{R}} q_1 a^{m_1} b^{n_1} \mapsto_{\mathcal{R}} q_2 a^{m_2} b^{n_2} \mapsto_{\mathcal{R}} \cdots$, where q_{in} is the initial state, and $m_0 = m$, $n_0 = 0$.
- A run is halting if it reaches a halting configuration.
- A natural number $m \in L(\mathcal{R})$ iff there exists a run on m such that $q_{\text{in}} a^m b^{n_0} \mapsto_{\mathcal{R}} q_1 a^{m_1} b^{n_1} \mapsto_{\mathcal{R}} \cdots \mapsto_{\mathcal{R}} q_s a^{m_s} b^{n_s}$, where $n_0 = 0$ and q_s is a halting state.

Back to the proof: construct a game $G_{\mathcal{R}}$

Let $\mathcal{R}$ be a universal 2-counter automaton.



Construct a game $G_{\mathcal{R}}$



If player II says "no", how does she challenge?

Player II wants to makes sure

(1) the sequence of configurations provided by player I is a sequence of the form $qa^m b^n$ and connected by $\rhd$,

(2) it starts with the initial configuration,

(3) any two consecutive configurations constitute a valid transition of $\mathcal{R}$, and

(4) the sequence of configurations is ended with a halting configuration.

The conditions (1), (2) and (4) are easy to check with Σ_1 conditions (i.e., player I lose with Π_1). In the following we explain how player II challenges if she thinks player I cheated by disobeying the above rule (3).

Check!

I $m_0 \in L(R)$? $q_0 a^{m_0} b^{n_0} \rhd \cdots \rhd q_i a^{m_i} b^{n_i} \rhd q_{i+1} a^{m_{i+1}} b$

II **Yes/No** **Challenge with a witness:** $c\bar{b}(\bar{a}\$)^{m_{i+1}}\bar{q} \lhd \bar{b}^{n_i}(\bar{a}\bar{\$})^{\min\{m_i, m_{i+1}\}}\bar{*}\bar{q}$

Such a play can be checked by a deterministic 2-stack visibly pushdown automaton.

Assume player II has a winning strategy σ, then

$$L(\mathcal{R}) = \{m : \text{player II follows her winning strategy } \sigma \text{ and answers "yes" to } m \text{ in the game } \mathbb{G}_{\mathcal{R}} \text{ in } 2\mathbf{DVPL}_\omega(\Sigma_1 \wedge \Pi_1)\}.$$

Since the halting problem of $\mathcal{R}$ is Σ^0_1-complete, any winning strategy for player II is Σ^0_1-hard.

$\square$

Corollary

The determinacy of games in $2\mathbf{DVPL}_\omega(\Sigma_1 \wedge \Pi_1)$ with an oracle implies ACA_0. In fact, they are equivalent to each other over RCA_0.

Sketch.

- We use a 2-counter automaton $\mathcal{R}$ with an oracle function $f : \mathbb{N} \to \mathbb{N}$, denoted as $\mathcal{R}^f$. $m_0 \in L(\mathcal{R}^f)$ iff there exists a run on m s.t.

$$q_0 a^{m_0} b^{n_0} \rhd q_1 a^{m_1} b^{n_1} \rhd \cdots \rhd q_s a^{m_s} b^{n_s},$$

  where $n_0 = f(m_0)$ and q_s is halting.

- The game $\mathbb{G}_{\mathcal{R}^f}$

I $m_0 \in L(R)?$ $\quad \boldsymbol{q_0 a^{m_0} b^{f(m_0)}} \rhd \cdots \rhd \boldsymbol{q_i a^{m_i} b^{n_i}} \rhd \boldsymbol{q_{i+1} a^{m_{i+1}} b}$ **Check!**

II **Yes/No** **Challenge with a witness:** $c\bar{b}(\bar{a}\$)^{m_{i+1}}\bar{q} \lhd \bar{b}^{n_i}(\bar{a}\bar{\$})^{\min\{m_i, m_{i+1}\}}\bar{*}\bar{q}$

- Such a play can be checked by a deterministic 2-stack visibly pushdown automaton with an oracle tape, in which the oracle tape is read-only, non-real-time and in the form $1^{f(0)}01^{f(1)}01^{f(2)}\cdots$.

Corollary

For any n, there exists an infinite game in $2\mathbf{DVPL}_\omega(\mathcal{B}(\Sigma_1))$ with only Σ^0_n-hard winning strategies.

- The brief idea is as follows. Take the case $n = 3$ as an example. Let A be any Σ^0_3 set. Then there is a 2-counter automaton $\mathcal{R}$ such that $m_0 \in A$ if and only if

$$\exists m_1 \forall m_2 (\mathcal{R} \text{ halts on } m = 2^{m_0} 3^{m_1} 5^{m_2}).$$

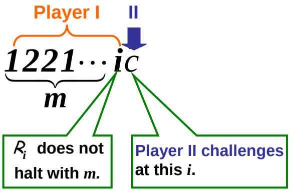
We can construct game starts with player I by asking m_0 in A or not.

Theorem

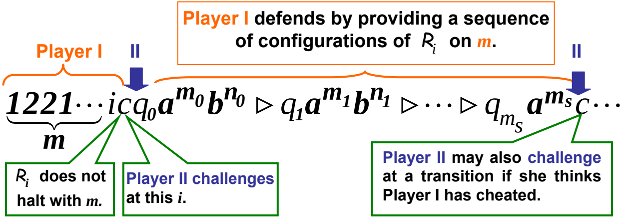
The determinacy of games in $2\mathbf{DVPL}_\omega(\Pi_1)$ implies Σ^0_1-SP.

Proof.

- Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be two 2-counter automata such that $L(\mathcal{R}_1) \bigcap L(\mathcal{R}_2) = \emptyset$.
- Player I has a sequence of m's, and for each m, he chooses i such that $\mathcal{R}_i$ does not halt with m.
- Player II may challenge player I's choice i at any m.



- Then player I defends by producing an infinite sequence of configurations of $\mathcal{R}_i$ on m, $q_0 a^{m_0} b^{n_0} \triangleright q_1 a^{m_1} b^{n_1} \cdots$, where $m_0 = m$ and $n_0 = 0$.



- While player I is producing a sequence of configurations, player II may challenge at any point she thinks player cheated.
- Player I's winning set is accepted by a 2DVPA(Π_1). Assume that player I has a winning strategy σ, then the desired separating set is $X = \{m : \text{player I follows } \sigma \text{ and picks } \mathcal{R}_2 \text{ for } m\}$. □

Corollary

The determinacy of the games in 2$\mathbf{DVPL}_\omega(\Pi_1)$ *with an oracle is equivalent to WKL*$_0$ *over RCA*$_0$*.*

Theorem

The determinacy of $2\mathbf{DVPL}_\omega(\Delta_2)$ *implies the determinacy of* Δ^0_1 *games in* ω^ω*, which is equivalent to* ATR_0*.*

- We mimic the proof of Δ^0_2-Det(in 2^ω) $\to$ Δ^0_1-Det(in ω^ω) in [NMT07][1].
- By using their coding technique, we write $\widetilde{\alpha} \in \omega^\omega$ for the unique sequence coded by $\alpha \in 2^\omega$. Note that not all sequences in 2^ω code a sequence in ω^ω.

[1] T. Nemoto , M.Y. Ould MedSalem, K. Tanaka. Infinite games in the Cantor space and subsystems of second order arithmetic. MLQ, 2007, 53(3): 226-236.

- Then, a play $\widetilde{\alpha}$ in Δ^0_1 game in ω^ω can be translated into a play α in 2^ω and α is winning for player 0 (resp. player 1) iff

  (a) $\widetilde{\alpha}$ is a winning play (resp. $\widetilde{\alpha}$ is not a winning play) in the Δ^0_1 game in ω^ω while both players obey the rules to produce a play α, or

  (b) while they are producing α, player 1 (resp. player 0) breaks the rules. [a Σ^0_2 statement]

  which constitutes a Σ^0_2 winning set for player 0 (resp. player 1). Thus the game is Δ^0_2 in 2^ω.

  Note that the increase in complexity of winning condition is mainly due to the complexity of the coding rules that we follow.

▶ Now we convert this Δ^0_1 game in ω^ω to a $2\mathbf{DVPL}_\omega(\Delta_2)$ using the coding rules given by the above Δ^0_2 game in 2^ω.

- The coding rule does not need any modification for $2\mathbf{DVPL}_\omega(\Delta_2)$. So, for simplicity, the players are assumed to obey this rule and we just treat the above case (a).

- Given a Δ^0_1 game, there exist two 2-counter automata $\mathcal{R}_0$ and $\mathcal{R}_1$ such that

$$s \text{ is a winning play } \leftrightarrow \exists n\ \sharp s[n] \in L(\mathcal{R}_0) \leftrightarrow \neg\exists n\ \sharp s[n] \in L(\mathcal{R}_1),$$

  where $\sharp s[n]$ denotes a code of the initial n-segment of s.

We construct a game $\mathbb{G}_{\mathcal{R}_1,\mathcal{R}_2}$ as follows.

- When a player produces a finite sequence $\alpha[n]$ in the Δ^0_2 game in 2^ω such that $\sharp\widetilde{\alpha}[n] \in L(\mathcal{R}_i)(i \in \{0, 1\})$, player i in the game $\mathbb{G}_{\mathcal{R}_0,\mathcal{R}_1}$ starts providing a sequence of configurations of $\mathcal{R}_i$ on $\sharp\widetilde{\alpha}[n]$, which player i claims to halt in finite steps.

- While player i is making such a sequence of configuration of $\mathcal{R}_i$ on $\sharp\widetilde{\alpha}[n]$, the player $1 - i$ may challenge at any point.

- We can see that the winning set for player 0 in the constructed game $\mathbb{G}_{\mathcal{R}_1,\mathcal{R}_2}$ is in $2\mathbf{DVPL}_\omega(\Delta_2)$. Moreover, if player i has a winning strategy in $2\mathbf{DVPL}_\omega(\Delta_2)$, then player i also has a winning strategy in the original Δ^0_1 game in ω^ω.

$\square$

Corollary

The determinacy of games $2\mathbf{DVPL}_\omega(\Sigma_2)$ is equivalent to ATR_0 over RCA_0.

Proof.

By the above theorem,

$$2\mathbf{DVPL}_\omega(\Sigma_2)\text{-Det} \to 2\mathbf{DVPL}_\omega(\Delta_2)\text{-Det} \to \mathrm{ATR}_0.$$

By [NMT07],

$$\mathrm{ATR}_0 \to \Sigma^0_2\text{-Det in } 2^\omega \to 2\mathbf{DVPL}_\omega(\Sigma_2)\text{-Det}.$$

$\square$

Determinacy strength of infinite games in pushdown ω-languages

- We treat ω-languages accepted by nondeterministic ω-PDA with different acceptance conditions.

Accepting conditions	Subclass of $(r\text{-})\mathbf{PDL}_\omega$[2]
Reachability	$(r\text{-})\mathbf{PDL}_\omega(\Sigma_1)$
Safety	$(r\text{-})\mathbf{PDL}_\omega(\Pi_1)$
Co-Büchi	$(r\text{-})\mathbf{PDL}_\omega(\Sigma_2)$
Büchi	$(r\text{-})\mathbf{PDL}_\omega(\Pi_2)$

Recall:

- $\mathbf{PDL}_\omega(\Pi_2) = \mathbf{CFL}_\omega$.
- $\mathbf{DPDL}_\omega(\Pi_2) \subsetneq \mathbf{DPDL}_\omega(\mathcal{B}(\Sigma_2)) = \mathbf{DCFL}_\omega$.
- $\mathbf{PDL}_\omega(\Pi_1)$ (respectively, $\mathbf{PDL}_\omega(\Sigma_1)$, $\mathbf{PDL}_\omega(\Sigma_2)$, $\mathbf{PDL}_\omega(\Pi_2)$) is a subclass of arithmetical Π^0_1 (respectively, Σ^0_1, Σ^0_2, Σ^1_1) class.

[2] The symbol r denotes the real-time case.

Intuitively...

- A play in an infinite game in 2**DVPL**$_\omega$:

Reading head

$$q_0 a^{m_0} b^{n_0} \triangleright \cdots \triangleright q_i a^{m_i} b^{n_i} \triangleright q_{i+1} a^{m_{i+1}} b c \boxed{\bar{b}(\bar{a}\$)^{m_{i+1}} \bar{q} \triangleleft \bar{b}^{n_i} (\bar{a}\$)^{\min\{m_i, m_{i+1}\}} \bar{*} \bar{q}} \cdots$$

To be compared by two stack, which is provoked by a **challenge** of the other player in the game.

- A play in an infinite game in **PDL**$_\omega$:

Reading head

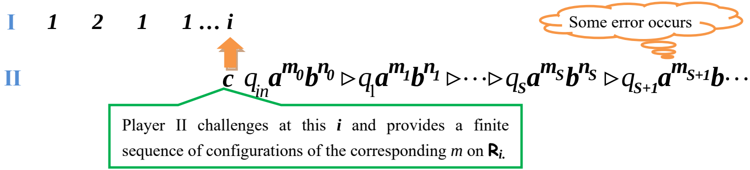
$$q_0 a^{m_0} b^{n_0} \triangleright \cdots \triangleright q_i a^{m_i} b^{n_i} \triangleright q_{i+1} a^{m_{i+1}} b \cdots$$

Has been compared due to the **nondeterminism** of the pushdown automata.

Theorem

The determinacy of games in r-$\mathbf{PDL}_\omega(\Sigma_1)$ implies Σ^0_1-SP.

Sketch.



Instead of checking by a challenge of player II, a pushdown automaton itself can nondeterministically check whether player I makes a mistake or not.

Remark

Safety condition is not the complement of reachability condition for nondeterministic pushdown languages.

Corollary

The determinacy of the games in r-$\mathbf{PDL}_\omega(\Sigma_1)$ with an oracle is equivalent to WKL_0 over RCA_0.

Theorem

$RCA_0 \vdash \mathbf{PDL}_\omega(\Pi_1)$-*Det.*

Proof idea

- Assume a pushdown automaton $\mathcal{M}$ with a Π_1 acceptance condition. We can construct a pushdown game $\mathcal{G}_\mathcal{P}$ such that
  - if there exists a computable winning strategy σ for player i in $\mathbb{G}(L(\mathcal{M}))$, then there exists a winning strategy σ' for player i in $\mathcal{G}_\mathcal{P}$ which is computable from σ, and vice versa.
- From Walukiewicz (1996, 2001), we can show that there is a winning strategy in $\mathcal{G}_\mathcal{P}$ and it is computable.

Note that a pushdown game is played on an infinite graph, which is generated by a pushdown process.

Theorem

$$\begin{array}{ccc} r\text{-}\mathbf{PDL}_\omega(\Sigma_2)\text{-}Det & \leftarrow \ ATR_0 \rightarrow & 2\mathbf{DVPL}_\omega(\Sigma_2)\text{-}Det \leftrightarrow 2\mathbf{DVPL}_\omega(\Pi_2)\text{-}Det \\ \downarrow & \uparrow & \downarrow \\ r\text{-}\mathbf{PDL}_\omega(\Delta_2)\text{-}Det & \rightarrow \Delta^0_1\text{-}Det \leftarrow & 2\mathbf{DVPL}_\omega(\Delta_2)\text{-}Det \\ & & \\ r\text{-}\mathbf{PDL}_\omega(\Sigma_1 \wedge \Pi_1)\text{-}Det & \leftrightarrow \ ACA_0 \leftrightarrow & 2\mathbf{DVPL}_\omega(\Sigma_1 \wedge \Pi_1)\text{-}Det \\ & & \\ r\text{-}\mathbf{PDL}_\omega(\Sigma_1)\text{-}Det & \leftrightarrow \ WKL_0 \leftrightarrow & 2\mathbf{DVPL}_\omega(\Sigma_1)\text{-}Det \leftrightarrow 2\mathbf{DVPL}_\omega(\Pi_1)\text{-}Det \end{array}$$

Corollary

For an acceptance condition $\mathcal{C} \in \{\Sigma_1, \Sigma_1 \wedge \Pi_1, \Delta_2, \Sigma_2\}$,

$$r\text{-}\mathbf{PDL}_\omega(\mathcal{C})\text{-}Det \leftrightarrow \mathbf{PDL}_\omega(\mathcal{C})\text{-}Det.$$

Determinacy strength of infinite games in 1-counter ω-languages

We can easily observe that all the arguments about pushdown automata, in fact, replaced by (nondeterministic) 1-counter automata, namely pushdown automata that can check whether the counter is zero or not with only one stack symbol.

Theorem

For an acceptance condition $\mathcal{C} \in \{\Sigma_1, \Sigma_1 \wedge \Pi_1, \Delta_2, \Sigma_2\}$,

$$r\text{-}\mathbf{CL}_\omega(\mathcal{C})\text{-}Det \leftrightarrow \mathbf{CL}_\omega(\mathcal{C})\text{-}Det \leftrightarrow \mathbf{PDL}_\omega(\mathcal{C})\text{-}Det.$$

Ongoing and future works

- Study the quantitative analysis of concurrent games in pushdown ω-languages.

- Investigate the determinacy strength of the Blackwell-type games in ω-languages, and its relation to probabilistic automata and other stochastic systems.

Reference

- R. Alur, P. Madhusudan. Visibly pushdown languages. In Proceedings of the 36th Annual ACM Symposium on Theory of Computing, STOC'04, 2004.
- O. Finkel. The determinacy of context-free games. Journal of Symbolic Logic, 2013, 78(4): 1115-1134.
- C. Löding, P. Madhusudan, O. Serre. Visibly pushdown games. Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2004: Springer Berlin Heidelberg, 2005: 408-420.
- T. Nemoto , M.Y. Ould MedSalem, K. Tanaka. Infinite games in the Cantor space and subsystems of second order arithmetic. Mathematical Logic Quarterly, 2007, 53(3): 226-236.
- Walukiewicz I. Pushdown processes: Games and model checking. Computer Aided Verification. Springer Berlin Heidelberg, 1996: 62-74. Journal version appeared in Information and Computation, 2001, 164(2): 234-263.

Thank you!