Regular Separability for Well-Matched Complements of Visibly Pushdown Languages

Christof Löding

RWTH Aachen University, Germany

Based on the paper Vince Bárány, Christof Löding, and Olivier Serre. Regularity problems for visibly pushdown languages. STACS 2006.

Separability Problems (ICALP workshop), Warsaw, July 14, 2017

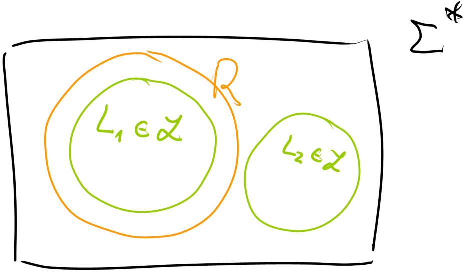
Regular Separability

Let $\mathcal{L}$ be a class of languages. The regular separability problem for $\mathcal{L}$ is the following decision problem.

Given: Two languages L_1, L_2 from $\mathcal{L}$

Question: Does there exist a regular language (the separator) with

$$L_1 \subseteq R \text{ and } L_2 \cap R = \emptyset?$$



Regular Separability

Let $\mathcal{L}$ be a class of languages. The regular separability problem for $\mathcal{L}$ is the following decision problem.

Given: Two languages L_1, L_2 from $\mathcal{L}$

Question: Does there exist a regular language (the separator) with

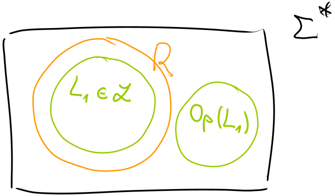
$$L_1 \subseteq R \text{ and } L_2 \cap R = \emptyset?$$

Let $Op : \mathcal{L} \to \mathcal{L}$ be a function on $\mathcal{L}$ (e.g., complement).

The specific regular separability problem for $\mathcal{L}$ and Op is

Given: A language L_1 from $\mathcal{L}$

Question: Does there exist a regular separator for L_1 and $Op(L_1)$?



Regular Separability

Let $\mathcal{L}$ be a class of languages. The regular separability problem for $\mathcal{L}$ is the following decision problem.

Given: Two languages L_1, L_2 from $\mathcal{L}$

Question: Does there exist a regular language (the separator) with

$$L_1 \subseteq R \text{ and } L_2 \cap R = \emptyset?$$

Let $Op : \mathcal{L} \to \mathcal{L}$ be a function on $\mathcal{L}$ (e.g., complement).

The specific regular separability problem for $\mathcal{L}$ and Op is

Given: A language L_1 from $\mathcal{L}$

Question: Does there exist a regular separator for L_1 and $Op(L_1)$?

In this talk: Decidability of the specific regular separability problem for visibly pushdown languages and the complement relative to well-matched words.

Outline

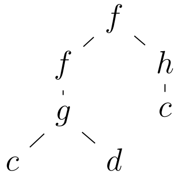
Outline

1. The Separation Problem

2. The Decidability Proof

Starting Point

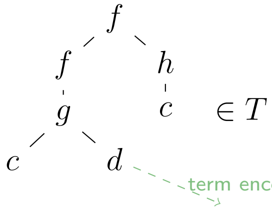
From (unranked ordered) trees to words:



$\in T$

Starting Point

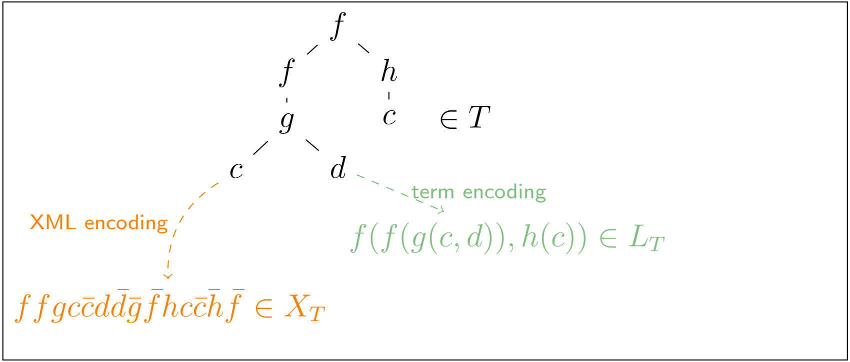
From (unranked ordered) trees to words:



$f(f(g(c,d)),h(c)) \in L_T$

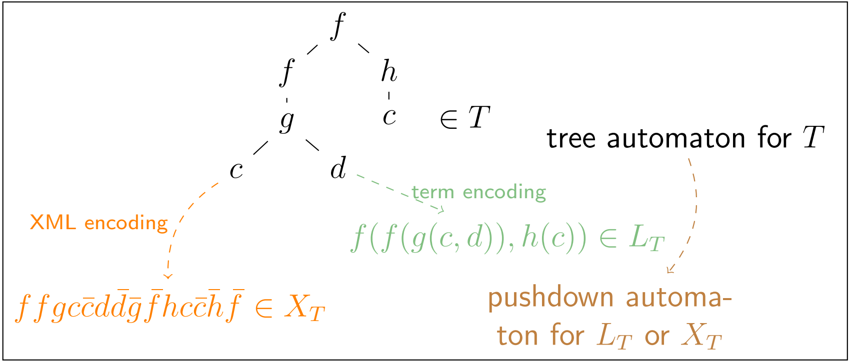
Starting Point

From (unranked ordered) trees to words:



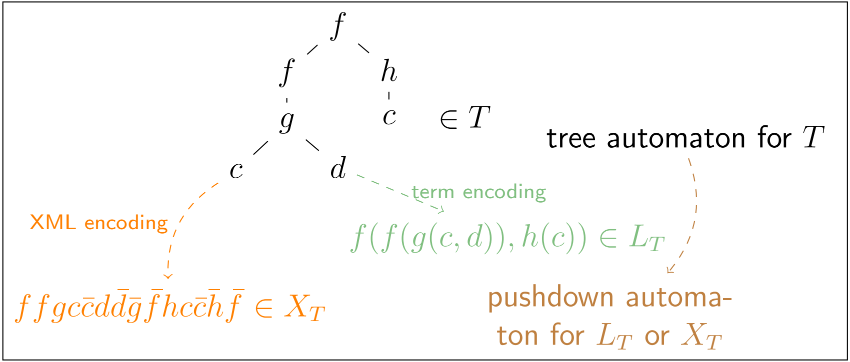
Starting Point

From (unranked ordered) trees to words:



Starting Point

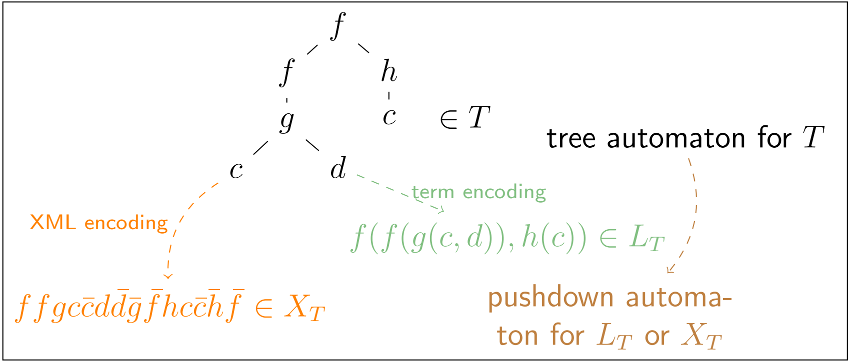
From (unranked ordered) trees to words:



Is L_T / X_T regular?

Starting Point

From (unranked ordered) trees to words:



Is L_T / X_T regular within the set of correct encodings L_{terms} / L_{XML}?

$$\exists R \text{ regular: } L_T = R \cap L_{\text{terms}}?$$
$$\exists R \text{ regular: } X_T = R \cap L_{\text{XML}}?$$

Examples

Trees over $\{f, g, c, d\}$

- $T =$ trees containing exactly one c

  Regular within both encodings

Examples

Trees over $\{f, g, c, d\}$

- $T =$ trees containing exactly one c

  Regular within both encodings

- $T =$ trees containing exactly one f and a c in the subtree below that f

  X_T regular within L_{XML}: there is c between unique f and $\bar{f}$

Examples

Trees over $\{f, g, c, d\}$

- $T =$ trees containing exactly one c

  Regular within both encodings

- $T =$ trees containing exactly one f and a c in the subtree below that f

  X_T regular within L_{XML}: there is c between unique f and $\bar{f}$

  L_T not regular within L_{terms}:
  $g(g(g(\cdots g(f(g(\cdots)))\cdots)\cdots c\cdots))))\cdots)$

Examples

Trees over $\{f, g, c, d\}$

- $T =$ trees containing exactly one c

  Regular within both encodings

- $T =$ trees containing exactly one f and a c in the subtree below that f

  X_T regular within L_{XML}: there is c between unique f and $\bar{f}$

  L_T not regular within L_{terms}:
  $g(g(g(\cdots g(f(g(\cdots)))\cdots)\cdots c\cdots))))\cdots)$

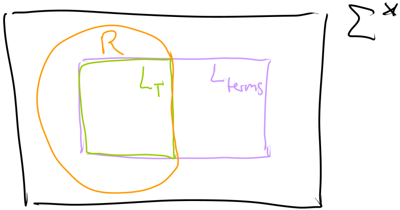
- $T =$ trees containing at least one f with a c the subtree below

  Not regular within both encodings

As a Separation Problem

Regularity of L_T within L_{terms} is the specific regular separability problem for

- the class $\mathcal{L} = \{L_T \mid T \text{ regular set of trees}\}$ and
- the operation $Op : L_T \mapsto L_{\text{terms}} \setminus L_T$

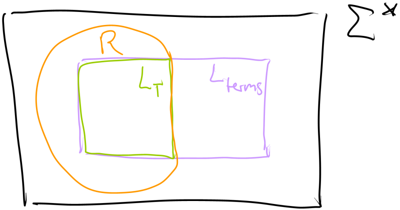


$$L_T = R \cap L_{\text{terms}}$$
$$\Leftrightarrow$$
$$L_T \subseteq R \text{ and } R \cap (L_{\text{terms}} \setminus L_T).$$

As a Separation Problem

Regularity of L_T within L_{terms} is the specific regular separability problem for

- the class $\mathcal{L} = \{L_T \mid T \text{ regular set of trees}\}$ and
- the operation $Op : L_T \mapsto L_{\text{terms}} \setminus L_T$



$$L_T = R \cap L_{\text{terms}}$$
$$\Leftrightarrow$$
$$L_T \subseteq R \text{ and } R \cap (L_{\text{terms}} \setminus L_T).$$

We solve a similar problem for the more general class of visibly pushdown languages.

Visibly Pushdown Automata (VPA)

Visibly pushdown alphabet $\Sigma = \langle \Sigma_{\mathrm{c}}, \Sigma_{\mathrm{r}}, \Sigma_{\mathrm{int}} \rangle$ with

- Σ_{c} = calls: push one letter onto the stack
- Σ_{r} = returns: pop one letter from the stack
- Σ_{int} = internal actions: stack remains unchanged

Example alphabets: trees with label alphabet Λ

- term encoding: $\langle \{\,(\,\}, \{\,)\,\}, \Lambda \rangle$
- XML encoding: $\langle \Lambda, \{\bar{a} \mid a \in \Lambda\}, \emptyset \rangle$

Visibly Pushdown Automata (VPA)

Visibly pushdown alphabet $\Sigma = \langle \Sigma_c, \Sigma_r, \Sigma_{int} \rangle$ with

- Σ_c = calls: push one letter onto the stack
- Σ_r = returns: pop one letter from the stack
- Σ_{int} = internal actions: stack remains unchanged

Example alphabets: trees with label alphabet Λ

- term encoding: $\langle \{\, (\,\}, \{\,) \,\}, \Lambda \rangle$
- XML encoding: $\langle \Lambda, \{\bar{a} \mid a \in \Lambda\}, \emptyset \rangle$

Visibly pushdown automaton $\mathcal{A} = (Q, \Sigma, \Gamma, q_0, \delta, F)$

- Q finite set of states, q_0 initial state
- Γ stack alphabet
- deterministic transitions δ of the form
$$\begin{array}{ll} q \xrightarrow{a} q'X & a \in \Sigma_c \\ qX \xrightarrow{a} q' & a \in \Sigma_r \\ q \xrightarrow{a} q' & a \in \Sigma_{int} \end{array}$$
- Acceptance: final states F + empty stack

Regular Separation for VPLs

Observation:

- For a regular tree language T, the languages L_T of term encodings and X_T of XML encodings are visibly pushdown languages (VPLs).
- Solving the regular separation problem for VPLs would solve the question of regularity within correct term or XML encodings.

Regular Separation for VPLs

Observation:

- For a regular tree language T, the languages L_T of term encodings and X_T of XML encodings are visibly pushdown languages (VPLs).
- Solving the regular separation problem for VPLs would solve the question of regularity within correct term or XML encodings.

Theorem (Kopczynski'16). The regular separability problem for VPLs is undecidable.

Well-Matched Words

The set L_{wm} of well-matched words over $\Sigma = \langle \Sigma_c, \Sigma_r, \Sigma_{int} \rangle$:

- empty word and each $a \in \Sigma_{int}$ is well matched
- awb is well matched for $a \in \Sigma_c$, $b \in \Sigma_r$, and w well matched
- $w_1 w_2$ is well matched for w_1 and w_2 well matched

Examples:

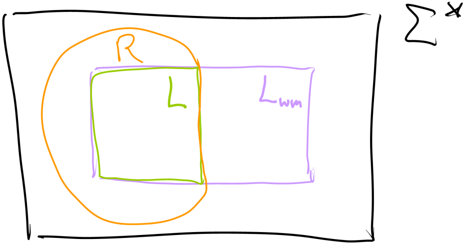
- term encoding: $\langle \{ \, (\, \}, \{ \,) \, \}, \Lambda \rangle$
- XML encoding: $\langle \Lambda, \{\bar{a} \mid a \in \Lambda\}, \emptyset \rangle$

In both cases, encodings of trees are well matched. But also other words are well matched, for example:

- $f((gfg(a))b(a))ggg()$
- $fff\bar{g}\bar{f}\bar{g}$

Regular Separation for Well-Matched Complements

Theorem (Barany,L.,Serre 2006). The specific regular separability problem for the class of VPLs and the relative complement operation on well-matched words is decidable: Given a VPL L, it is decidable whether there is a regular language R with $L = R \cap L_{wm}$.



Regular Separation for Well-Matched Complements

Theorem (Barany,L.,Serre 2006). The specific regular separability problem for the class of VPLs and the relative complement operation on well-matched words is decidable: Given a VPL L, it is decidable whether there is a regular language R with $L = R \cap L_{wm}$.

For term encodings, the difference between L_{wm} and L_{terms} is "small enough" to obtain decidability of the initial problem:

Corollary. For a regular tree language T, it is decidable whether the language L_T of term encodings for T is regular within the set L_{term} of all term encodings.

Regular Separation for Well-Matched Complements

Theorem (Barany,L.,Serre 2006). The specific regular separability problem for the class of VPLs and the relative complement operation on well-matched words is decidable: Given a VPL L, it is decidable whether there is a regular language R with $L = R \cap L_{wm}$.

For term encodings, the difference between L_{wm} and L_{terms} is "small enough" to obtain decidability of the initial problem:

Corollary. For a regular tree language T, it is decidable whether the language L_T of term encodings for T is regular within the set L_{term} of all term encodings.

Remark: For the XML encoding the difference between L_{wm} and L_{XML} is "too large". The decidability of regularity within L_{XML} is an open problem (originally asked by Segoufin/Vianu'02).

Outline

1 The Separation Problem

2 The Decidability Proof

Visibly 1-Counter Automata

A deterministic **visibly 1-counter automaton (V1CA)** $\mathcal{C}$ is a finite automaton with a (non-negative) counter that is

- incremented for call symbols,
- decremented for return symbols (blocks if return on value 0),
- left unchanged for internal symbols.

An m-V1CA can distinguish the counter values $0, 1, \ldots, m-1, \geq m$ by transition functions $\delta_0, \ldots, \delta_m$.

Acceptance: final state and counter 0

A 0-V1CA is also called **visibly 1-counter net (V1CN)**.

Visibly 1-Counter Automata

A deterministic **visibly 1-counter automaton (V1CA)** $\mathcal{C}$ is a finite automaton with a (non-negative) counter that is

- incremented for call symbols,
- decremented for return symbols (blocks if return on value 0),
- left unchanged for internal symbols.

An m-V1CA can distinguish the counter values $0, 1, \ldots, m-1, \geq m$ by transition functions $\delta_0, \ldots, \delta_m$.

Acceptance: final state and counter 0

A 0-V1CA is also called **visibly 1-counter net (V1CN)**.

Observation: The VPLs of the form $L = R \cap L_{wm}$ for regular R, are precisely those definable by V1CNs.

Deciding Definability by V1CN

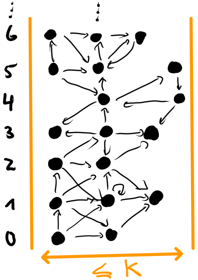
Lemma. A VPL L is regular within the set L_{wm} of well-matched words if, and only if, L is definable by a visibly one-counter net.

We show decidability of this problem:

Given a VPA $\mathcal{A}$, is it equivalent to a V1CN?

Slender Configuration Graphs

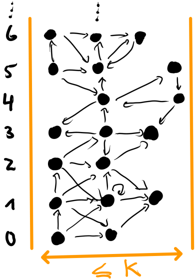
A configuration of a V1CN is (q, n) (state and counter value).
The configuration graph $G_{\mathcal{C}}$ of a V1CN $\mathcal{C}$ is "slender": for each counter value at most K configurations for K = number of states of the V1CN.



Slender Configuration Graphs

A configuration of a V1CN is (q, n) (state and counter value).

The configuration graph $G_{\mathcal{C}}$ of a V1CN $\mathcal{C}$ is "slender": for each counter value at most K configurations for K = number of states of the V1CN.



- A configuration of a VPA is a word $q\sigma$ (state + stack)
- Define $q\sigma \sim q'\sigma'$ if $|\sigma| = |\sigma'|$ and the same words are accepted from the two configurations.

Necessary condition: If the VPA $\mathcal{A}$ is equivalent to a V1CN, then merging equivalent configurations yields a slender graph $G_{\mathcal{A}}/_{\sim}$.

Example

Alphabet: $\Sigma_{\mathrm{c}} = \{a, b\}$, $\Sigma_{\mathrm{r}} = \{a', b'\}$, $\Sigma_{\mathrm{int}} = \emptyset$.

States: q_0, q_1 with final state $F = \{q_1\}$

Transitions:

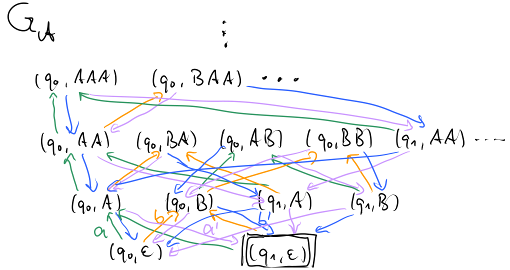
$$\begin{array}{ll} q_0/q_1 \xrightarrow{a} q_0 A & q_0/q_1 \xrightarrow{b} q_0 B \\ q_0/q_1 A \xrightarrow{a'} q_1 & q_0/q_1 B \xrightarrow{a'} q_0 \\ q_0/q_1 A \xrightarrow{b'} q_0 & q_0/q_1 B \xrightarrow{b'} q_1 \end{array}$$

Example

Alphabet: $\Sigma_c = \{a, b\}$, $\Sigma_r = \{a', b'\}$, $\Sigma_{int} = \emptyset$.
States: q_0, q_1 with final state $F = \{q_1\}$
Transitions: $q_0/q_1 \xrightarrow{a} q_0 A \quad q_0/q_1 \xrightarrow{b} q_0 B$

$$q_0/q_1 A \xrightarrow{a'} q_1 \quad q_0/q_1 B \xrightarrow{a'} q_0$$

$$q_0/q_1 A \xrightarrow{b'} q_0 \quad q_0/q_1 B \xrightarrow{b'} q_1$$



Example

Alphabet: $\Sigma_c = \{a, b\}$, $\Sigma_r = \{a', b'\}$, $\Sigma_{int} = \emptyset$.

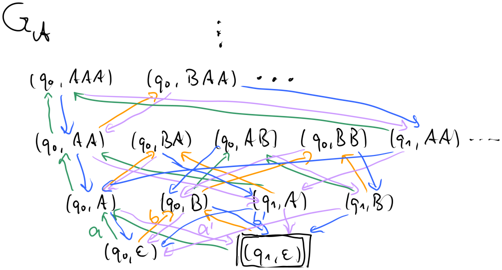
States: q_0, q_1 with final state $F = \{q_1\}$

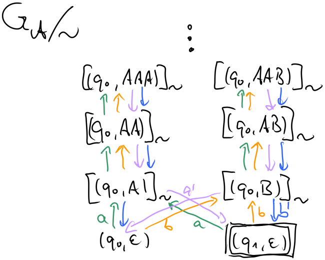
Transitions:

$$q_0/q_1 \xrightarrow{a} q_0 A \quad q_0/q_1 \xrightarrow{b} q_0 B$$

$$q_0/q_1 A \xrightarrow{a'} q_1 \quad q_0/q_1 B \xrightarrow{a'} q_0$$

$$q_0/q_1 A \xrightarrow{b'} q_0 \quad q_0/q_1 B \xrightarrow{b'} q_1$$





Deciding Slenderness

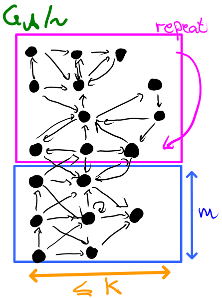
VPA $\mathcal{A}$

(1) $G_{\mathcal{A}}/_{\sim}$ slender? – decidable

(2) $G_{\mathcal{A}}/_{\sim}$ slender

$\Downarrow$

$G_{\mathcal{A}}/_{\sim}$ ultimately periodic

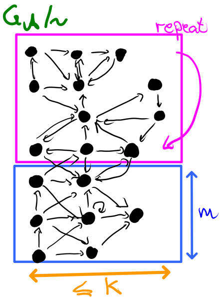


Deciding Slenderness

VPA $\mathcal{A}$ $\longrightarrow$

(1) $G_{\mathcal{A}}/_{\sim}$ slender? – decidable

(2) $G_{\mathcal{A}}/_{\sim}$ slender $\Downarrow$ $G_{\mathcal{A}}/_{\sim}$ ultimately periodic

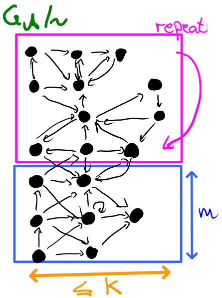


- One can construct a regular set Rep of representatives of the equivalence classes of $\sim$.

Deciding Slenderness

VPA $\mathcal{A}$ —— (1) $G_{\mathcal{A}}/_{\sim}$ slender? – decidable ——→

(2) $G_{\mathcal{A}}/_{\sim}$ slender $\Downarrow$ $G_{\mathcal{A}}/_{\sim}$ ultimately periodic

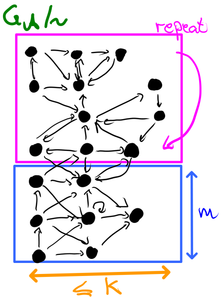


- One can construct a regular set Rep of representatives of the equivalence classes of $\sim$.
- Then $G_{\mathcal{A}}/_{\sim}$ slender if there is a bound K such that Rep contains at most K words of each length. This is decidable for regular languages (Păun,Salomaa 1995).

Deciding Slenderness

VPA $\mathcal{A}$ —— (1) $G_\mathcal{A}/_\sim$ slender? – decidable ——→

(2) $G_\mathcal{A}/_\sim$ slender
$\Downarrow$
$G_\mathcal{A}/_\sim$ ultimately periodic



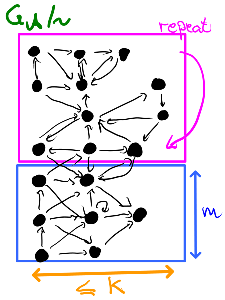
- One can construct a regular set Rep of representatives of the equivalence classes of $\sim$.
- Then $G_\mathcal{A}/_\sim$ slender if there is a bound K such that Rep contains at most K words of each length. This is decidable for regular languages (Păun,Salomaa 1995).
- Ultimate periodicity of $G_\mathcal{A}/_\sim$ is obtained from the structure of Rep.

Completing the Proof

VPA $\mathcal{A}$

(1) $G_{\mathcal{A}}/\sim$ slender? – decidable

(2) $G_{\mathcal{A}}/\sim$ slender $\Downarrow$ $G_{\mathcal{A}}/\sim$ ultimately periodic



yields

m-V1CA for $L(\mathcal{A})$

decidable

V1CN for $L(\mathcal{A})$

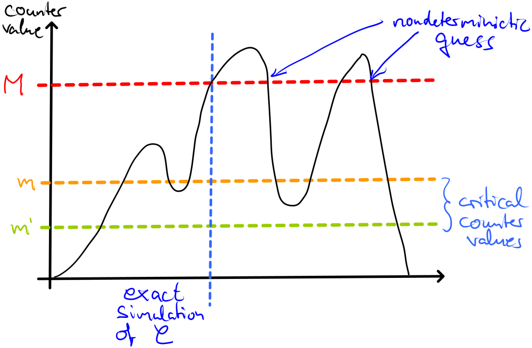
Deciding V1CN Definability

Theorem (Barany,L.,Serre 2006). For a given m-V1CA $\mathcal{C}$ and $m' < m$ it is decidable whether $\mathcal{C}$ is equivalent to an m'-V1CA.

Deciding V1CN Definability

Theorem (Barany,L.,Serre 2006). For a given m-V1CA $\mathcal{C}$ and $m' < m$ it is decidable whether $\mathcal{C}$ is equivalent to an m'-V1CA.

Idea: Construct candidate m'-V1CA $\mathcal{C}'$ that counts up to some large M (depending on $\mathcal{C}$) in its state space and guesses when it falls below M again.



Conclusion and Outlook

We have shown:

- It is decidable whether a given VPA is equivalent to a V1CN.
- This solves the specific regular separability problem for VPLs and their well-matched complements.
- It also implies that it is decidable for a regular tree language whether its set of term encodings is regular within the set of all term encodings.

Open questions:

- Decidability of other (specific) regular separability problems for (sub-classes) of visibly pushdown languages?
- In particular: Is it decidable for a regular tree language whether its set of XML encodings is regular within the set of all XML encodings?