Logical aspects of the lexicographic order on 1-counter languages

Dietrich Kuske
Ilmenau, Germany

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable

2. Ésik et al.: Caucal's result does not extend to context-free languages

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable

2. Ésik et al.: Caucal's result does not extend to context-free languages

new results

Caucal's result does not extend to 1-counter languages

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable: $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages

new results

Caucal's result does not extend to 1-counter languages

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \mathrm{MSO}, (L(M), \leqslant_{\mathrm{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages:
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\mathrm{lex}}) \models \varphi\}$ undecidable (Ésik '11)

new results

Caucal's result does not extend to 1-counter languages

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $$\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\} \text{ decidable}$$
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable: $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)
   2.2 there is a PDA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Carayol & Ésik '12)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)
   2.2 there is a PDA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Carayol & Ésik '12)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable
2. $\exists$ a 1CA M s.t. $\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undec.

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)
   2.2 there is a PDA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Carayol & Ésik '12)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable
2. $\exists$ a 1CA M s.t. $\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undec.
   but $\{\varphi \in \Sigma_2 \mid \mathcal{L} \models \varphi\}$ decidable f.a. linear orders $\mathcal{L}$

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)
   2.2 there is a PDA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Carayol & Ésik '12)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable
2. $\exists$ a 1CA M s.t. $\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undec.
   but $\{\varphi \in \Sigma_2 \mid \mathcal{L} \models \varphi\}$ decidable f.a. linear orders $\mathcal{L}$
3. there is a 1CA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ not arithmetical

1. Caucal '02: validity of MSO-sentences on lexicographic order of deterministic context-free languages is uniformly decidable:
   $\{(M, \varphi) \mid M \text{ DPDA}, \varphi \in \text{MSO}, (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ decidable
2. Ésik et al.: Caucal's result does not extend to context-free languages
   2.1 there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ PDA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Ésik '11)
   2.2 there is a PDA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable (Carayol & Ésik '12)

new results

Caucal's result does not extend to 1-counter languages

1. there is $\varphi \in \Sigma_2$ s.t. $\{M \text{ 1CA} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undecidable
2. $\exists$ a 1CA M s.t. $\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ undec.
   but $\{\varphi \in \Sigma_2 \mid \mathcal{L} \models \varphi\}$ decidable f.a. linear orders $\mathcal{L}$
3. there is a 1CA M s.t. $\{\varphi \in \text{FO} \mid (L(M), \leqslant_{\text{lex}}) \models \varphi\}$ not arithmetical

Your vote, please

Your vote, please

What proof whould you like to see?

Your vote, please

What proof whould you like to see?

Option 1 There exists a 1CA M such that
$\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\mathrm{lex}}) \models \varphi\}$ is undecidable.

Your vote, please

What proof whould you like to see?

Option 1 There exists a 1CA M such that
$\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\mathrm{lex}}) \models \varphi\}$ is undecidable.

Option 2 $\{\varphi \in \Sigma_2 \mid \mathcal{L} \models \varphi\}$ is decidable for all linear orders $\mathcal{L}$.

Your vote, please

What proof whould you like to see?

Option 1 There exists a 1CA M such that $\{\varphi \in \Sigma_3 \mid (L(M), \leqslant_{\mathrm{lex}}) \models \varphi\}$ is undecidable.

▸ Proof

Option 2 $\{\varphi \in \Sigma_2 \mid \mathcal{L} \models \varphi\}$ is decidable for all linear orders $\mathcal{L}$.

▸ Proof

A simple Ehrenfeucht-Fraïssé-game for Σ_2

board: two linear orders $\mathcal{L}_0$ and $\mathcal{L}_1$

A simple Ehrenfeucht-Fraïssé-game for Σ_2

board: two linear orders $\mathcal{L}_0$ and $\mathcal{L}_1$
1st round:

- spoiler chooses $i \in \{0,1\}$ and arbitrary number m of elements $a_1 < a_2 < \cdots < a_m$ in $\mathcal{L}_i$
- duplicator answers with same number m of elements $b_1 < b_2 < \cdots < b_m$ in $\mathcal{L}_{1-i}$

A simple Ehrenfeucht-Fraïssé-game for Σ_2

board: two linear orders $\mathcal{L}_0$ and $\mathcal{L}_1$

1st round:

- spoiler chooses $i \in \{0, 1\}$ and arbitrary number m of elements $a_1 < a_2 < \cdots < a_m$ in $\mathcal{L}_i$
- duplicator answers with same number m of elements $b_1 < b_2 < \cdots < b_m$ in $\mathcal{L}_{1-i}$

2nd round:

- spoiler chooses arbitrary number n of unchosen elements $b_{m+1} < b_{m+2} < \cdots < b_{m+n}$ in $\mathcal{L}_{1-i} \setminus \{b_1, \ldots, b_n\}$
- duplicator answers with same number n of unchosen elements $a_{m+1} < a_{m+2} < \cdots < a_{m+n}$ in $\mathcal{L}_i \setminus \{a_1, \ldots, a_n\}$

A simple Ehrenfeucht-Fraïssé-game for Σ_2

board: two linear orders $\mathcal{L}_0$ and $\mathcal{L}_1$
1st round:

- spoiler chooses $i \in \{0, 1\}$ and arbitrary number m of elements $a_1 < a_2 < \cdots < a_m$ in $\mathcal{L}_i$
- duplicator answers with same number m of elements $b_1 < b_2 < \cdots < b_m$ in $\mathcal{L}_{1-i}$

2nd round:

- spoiler chooses arbitrary number n of unchosen elements $b_{m+1} < b_{m+2} < \cdots < b_{m+n}$ in $\mathcal{L}_{1-i} \setminus \{b_1, \ldots, b_n\}$
- duplicator answers with same number n of unchosen elements $a_{m+1} < a_{m+2} < \cdots < a_{m+n}$ in $\mathcal{L}_i \setminus \{a_1, \ldots, a_n\}$

duplicator wins if tuples $(a_1, \ldots, a_{m+n})$ and $(b_1, \ldots, b_{m+n})$ are ordered in the same way

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.

Duplicator chooses m consecutive elements $b_1 \lessdot b_2 \lessdot \cdots \lessdot b_m$ in $\mathcal{L}$.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.
Duplicator chooses m consecutive elements $b_1 \lessdot b_2 \lessdot \cdots \lessdot b_m$ in $\mathcal{L}$.

Spoiler chooses n elements from $\mathcal{L} \setminus \{b_1, \ldots, b_m\}$.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.
Duplicator chooses m consecutive elements $b_1 \lessdot b_2 \lessdot \cdots \lessdot b_m$ in $\mathcal{L}$.

Spoiler chooses n elements from $\mathcal{L} \setminus \{b_1, \ldots, b_m\}$. Some of these elements lie to the left, some to the right of interval $[b_1, b_m]$, but none in this interval.

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.
Duplicator chooses m <u>consecutive</u> elements $b_1 \lessdot b_2 \lessdot \cdots \lessdot b_m$ in $\mathcal{L}$.

Spoiler chooses n elements from $\mathcal{L} \setminus \{b_1, \ldots, b_m\}$. Some of these elements lie to the left, some to the right of interval $[b_1, b_m]$, but none in this interval.
Duplicator chooses corresponding elements in $\mathbb{Z} \setminus [a_1, a_m]$

Observation

Duplicator has a winning strategy on $(\mathcal{L}_0, \mathcal{L}_1)$ iff $\mathcal{L}_1 \equiv_{\Sigma_2} \mathcal{L}_2$.

Application

$\mathcal{L}$ linear order without endpoints s.t. size of finite intervals is not finitely bounded. Then $(\mathbb{Z}, \leqslant) \equiv_{\Sigma_2} \mathcal{L}$.

Proof

Spoiler chooses $a_1 < a_2 < \cdots < a_m$ in $\mathbb{Z}$.
Duplicator chooses m consecutive elements $b_1 \lessdot b_2 \lessdot \cdots \lessdot b_m$ in $\mathcal{L}$.

Spoiler chooses n elements from $\mathcal{L} \setminus \{b_1, \ldots, b_m\}$. Some of these elements lie to the left, some to the right of interval $[b_1, b_m]$, but none in this interval.
Duplicator chooses corresponding elements in $\mathbb{Z} \setminus [a_1, a_m]$ – and wins. $\square$

Lemma

$\mathcal{L}$ linear order without endpoints.

1. If the size of finite intervals in $\mathcal{L}$ is not finitely bounded, then $\mathcal{L} \equiv_{\Sigma_2} (\mathbb{Z}, \leqslant)$.

Lemma

$\mathcal{L}$ linear order without endpoints.

1. If the size of finite intervals in $\mathcal{L}$ is not finitely bounded, then $\mathcal{L} \equiv_{\Sigma_2} (\mathbb{Z}, \leqslant)$.
2. If there are infinitely many finite intervals of size N, but none of size $N + 1$, then
$\mathcal{L} \equiv_{\Sigma_2} \mathbf{N} \cdot (\mathbb{Q}, \leqslant)$.

Lemma

$\mathcal{L}$ linear order without endpoints.

1. If the size of finite intervals in $\mathcal{L}$ is not finitely bounded, then $\mathcal{L} \equiv_{\Sigma_2} (\mathbb{Z}, \leqslant)$.
2. If there are infinitely many finite intervals of size N, but none of size $N+1$, then $\mathcal{L} \equiv_{\Sigma_2} \mathbf{N} \cdot (\mathbb{Q}, \leqslant)$.

Proof of 2nd statement as for 1st on previous slide. $\square$

Lemma and observation

$\mathcal{L}$ linear order without endpoints.

1. If the size of finite intervals in $\mathcal{L}$ is not finitely bounded, then $\mathcal{L} \equiv_{\Sigma_2} (\mathbb{Z}, \leqslant) \cong (1^*2 \cup 2^*, \leqslant_{\mathrm{lex}})$.
2. If there are infinitely many finite intervals of size N, but none of size $N+1$, then
$\mathcal{L} \equiv_{\Sigma_2} \mathbf{N} \cdot (\mathbb{Q}, \leqslant)$.

Proof of 2nd statement as for 1st on previous slide. $\square$

Lemma and observation

$\mathcal{L}$ linear order without endpoints.

1. If the size of finite intervals in $\mathcal{L}$ is not finitely bounded, then $\mathcal{L} \equiv_{\Sigma_2} (\mathbb{Z}, \leqslant) \cong (1^*2 \cup 2^*, \leqslant_{\mathrm{lex}})$.
2. If there are infinitely many finite intervals of size N, but none of size $N+1$, then
   $\mathcal{L} \equiv_{\Sigma_2} \mathbf{N} \cdot (\mathbb{Q}, \leqslant) \cong (\{0,2\}^*\{1, 11, \ldots, 1^{N-1}, 2\}), \leqslant_{\mathrm{lex}})$.

Proof of 2nd statement as for 1st on previous slide. $\square$

more generally . . .

Proposition

If $\mathcal{L}$ is a linear order, then there exists a regular language L such that $\mathcal{L} \equiv_{\Sigma_2} (L, \leqslant_{\mathrm{lex}})$.

more generally . . .

Proposition

If $\mathcal{L}$ is a linear order, then there exists a regular language L such that $\mathcal{L} \equiv_{\Sigma_2} (L, \leqslant_{\mathrm{lex}})$.

Theorem

For every linear order $\mathcal{L}$, the Σ_2-theory is decidable

more generally ...

Proposition

If $\mathcal{L}$ is a linear order, then there exists a regular language L such that $\mathcal{L} \equiv_{\Sigma_2} (L, \leqslant_{\text{lex}})$.

Theorem

For every linear order $\mathcal{L}$, the Σ_2-theory is decidable ...

... since it equals the Σ_2-theory of $(L, \leqslant_{\text{lex}})$ for some regular language L

more generally ...

Proposition

If $\mathcal{L}$ is a linear order, then there exists a regular language L such that $\mathcal{L} \equiv_{\Sigma_2} (L, \leqslant_{\mathrm{lex}})$.

Theorem

For every linear order $\mathcal{L}$, the Σ_2-theory is decidable ...

... since it equals the Σ_2-theory of $(L, \leqslant_{\mathrm{lex}})$ for some regular language L which is decidable by Caucal's result.

more generally ...

Proposition

If $\mathcal{L}$ is a linear order, then there exists a regular language L such that $\mathcal{L} \equiv_{\Sigma_2} (L, \leqslant_{\mathrm{lex}})$.

Theorem

For every linear order $\mathcal{L}$, the Σ_2-theory is decidable ...

... since it equals the Σ_2-theory of $(L, \leqslant_{\mathrm{lex}})$ for some regular language L which is decidable by Caucal's result.

goto summary

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof strategy

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof strategy

Let $A \subseteq \mathbb{N}$ be computably enumerable, but not computable.

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof strategy

Let $A \subseteq \mathbb{N}$ be computably enumerable, but not computable.

There exists a two-counter machine M that halts from $(n, 0)$ iff $n \in A$.

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

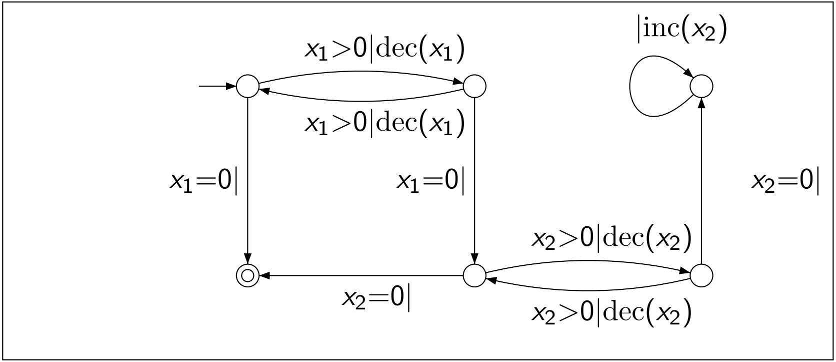
Proof strategy

Let $A \subseteq \mathbb{N}$ be computably enumerable, but not computable.

There exists a two-counter machine M that halts from $(n, 0)$ iff $n \in A$.

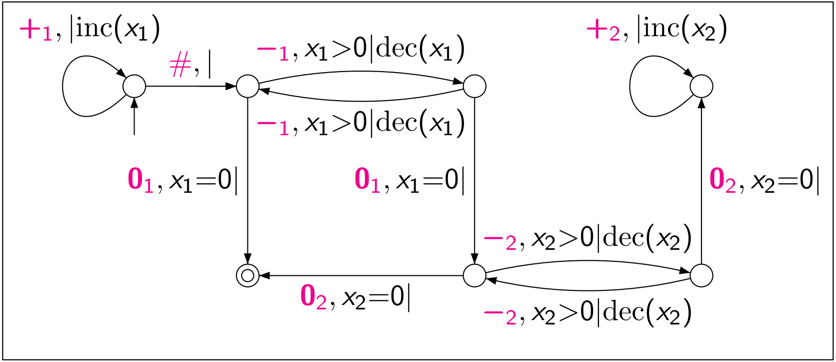
Build, from M, a one-counter language L such that "M halts from $(n, 0)$" is a Σ_3-statement on $(L, \leqslant_{\mathrm{lex}})$.

A two-counter machine M



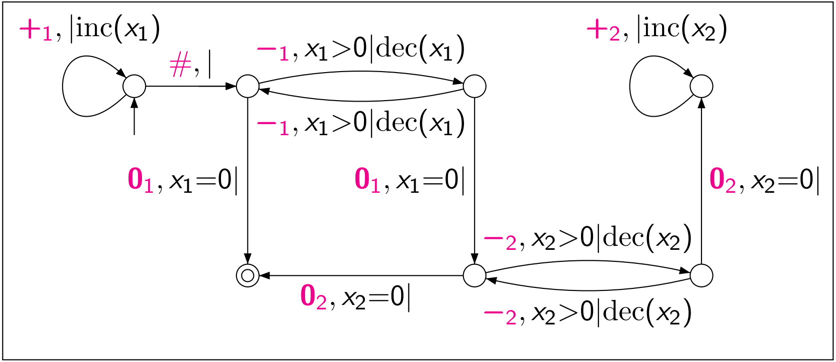
$n \in A$
iff M halts from $(n, 0)$

The two-counter automaton $2CA$



$n \in A$
iff M halts from $(n, 0)$

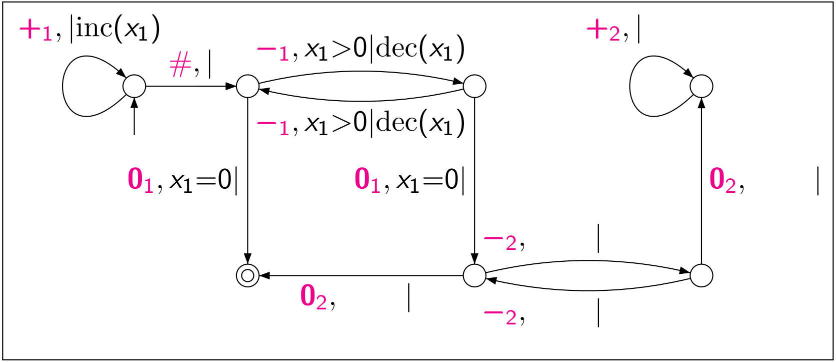
The two-counter automaton $2CA$



$n \in A$
iff M halts from $(n, 0)$
iff some word from $L(2CA)$ starts with $+_1^n\#$

The one-counter automaton $1CA_1$

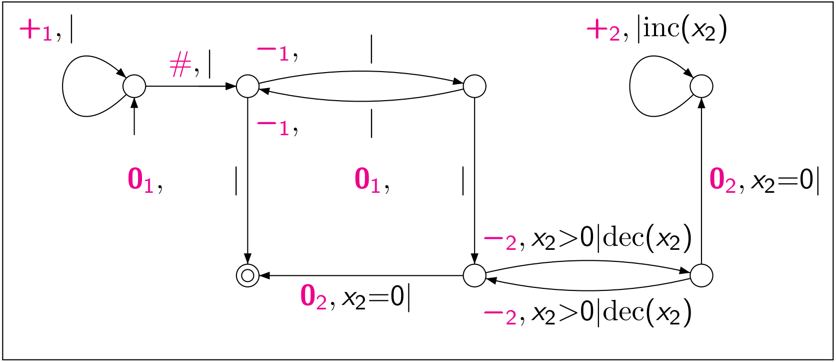
tests and manipulates only first counter



$n \in A$
iff M halts from $(n, 0)$
iff some word from $L(2CA)$ starts with $+_1^n\#$

The one-counter automaton $1CA_2$

tests and manipulates only second counter



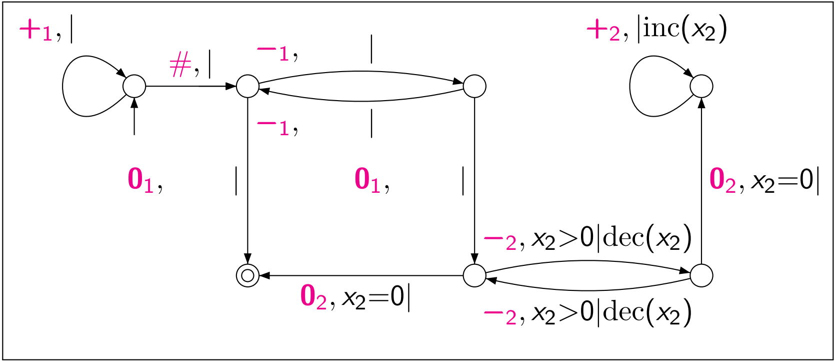
$n \in A$

iff M halts from $(n, 0)$

iff some word from $L(2CA)$ starts with $+_1^n\#$

The one-counter automaton $1CA_2$

tests and manipulates only second counter



$n \in A$
iff M halts from $(n, 0)$
iff some word from $L(2CA)$ starts with $+_1^n\#$
iff some word from $L(1CA_1) \cap L(1CA_2)$ starts with $+_1^n\#$.

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \ \ldots\ < \quad +_1^n\# \ < \ \cdots$$

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \quad \ldots \quad < \quad +_1^n\# \quad < \quad \cdots$$

$$+_1^n\# s_1 < +_1^n\# s_2 < +_1^n\# s_3$$

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \quad \dots \quad < \quad +_1^n\# \quad < \quad \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 \quad < \quad \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$\# < +_1\# < +_1^2\# < \ldots < +_1^n\# < \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 < \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 < +_1^n\#v\#1\{0,1\}^*1$$

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \quad \dots \quad < \quad +_1^n\# \quad < \quad \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 \quad < \quad \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$\begin{array}{ccc} +_1^n\#v\#0\{0,1\}^*1 & < & +_1^n\#v\#1\{0,1\}^*1 \\ \cong (\mathbb{Q}, \leqslant) = \eta & & \cong (\mathbb{Q}, \leqslant) \end{array}$$

$$\# < +_1\# < +_1^2\# < \dots < +_1^n\# < \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 < \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 < +_1^n\#v\#m_1 < \qquad +_1^n\#v\#1\{0,1\}^*1$$

$$\cong (\mathbb{Q},\leqslant) = \eta \qquad \cong (\mathbb{Q},\leqslant)$$

if $+_1^n\#v \in L(1CA_1)$

$$\# < +_1\# < +_1^2\# < \dots < +_1^n\# < \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 < \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 < +_1^n\#v\#m_1 < +_1^n\#v\#m_2 < +_1^n\#v\#1\{0,1\}^*1$$

$\cong (\mathbb{Q}, \leqslant) = \eta$ $\qquad\qquad$ $\cong (\mathbb{Q}, \leqslant)$

if $+_1^n\#v \in L(1CA_1)$ $\qquad\qquad$ if $+_1^n\#v \in L(1CA_2)$

$$\# < +_1\# < +_1^2\# < \dots < +_1^n\# < \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 < \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 < +_1^n\#v\#m_1 < +_1^n\#v\#m_2 < +_1^n\#v\#1\{0,1\}^*1$$

$$\cong (\mathbb{Q},\leqslant) = \eta \qquad \cong (\mathbb{Q},\leqslant)$$

if $+_1^n\#v \in L(1CA_1)$ if $+_1^n\#v \in L(1CA_2)$

consequences

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \quad \dots \quad < \quad +_1^n\# \quad < \quad \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 \quad < \quad \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 \quad < +_1^n\#v\#m_1 < +_1^n\#v\#m_2 < \quad +_1^n\#v\#1\{0,1\}^*1$$

$$\cong (\mathbb{Q}, \leqslant) = \eta \qquad\qquad\qquad\qquad \cong (\mathbb{Q}, \leqslant)$$

if $+_1^n\#v \in L(1CA_1)$ if $+_1^n\#v \in L(1CA_2)$

consequences

- $+_1^n\#v\#$ is replaced by $\eta + \mathbf{2} + \eta$ if v encodes halting computation from $(n, 0)$, and by $\eta + \eta = \eta = \eta + \mathbf{1} + \eta$ otherwise

$$\# \quad < \quad +_1\# \quad < \quad +_1^2\# \quad < \quad \dots \quad < \quad +_1^n\# \quad < \quad \cdots$$

$$+_1^n\#s_1 < +_1^n\#s_2 < +_1^n\#s_3 \quad < \quad \{+_1^n\#v\# \mid v \in \Delta^*\}$$

$$+_1^n\#v\#0\{0,1\}^*1 \quad < +_1^n\#v\#m_1 < +_1^n\#v\#m_2 < \quad +_1^n\#v\#1\{0,1\}^*1$$

$$\cong (\mathbb{Q}, \leqslant) = \eta \qquad\qquad \cong (\mathbb{Q}, \leqslant)$$

if $+_1^n\#v \in L(1CA_1)$ if $+_1^n\#v \in L(1CA_2)$

consequences

- $+_1^n\#v\#$ is replaced by $\eta + \mathbf{2} + \eta$ if v encodes halting computation from $(n, 0)$, and by $\eta + \eta = \eta = \eta + \mathbf{1} + \eta$ otherwise
- $+_1^n\#$ is replaced by $\mathbf{3} + \eta + \mathbf{2} + \eta$ if there is halting computation from $(n, 0)$, and by $\mathbf{3} + \eta$ otherwise

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof.

with $(L, \leqslant_{\mathrm{lex}}) \cong \sum_{n\in\mathbb{N}} \left(\mathbf{3} + \begin{cases} \eta + \mathbf{2} + \eta & (n \in A) \\ \eta & (n \notin A) \end{cases} \right)$

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof.

with $(L, \leqslant_{\mathrm{lex}}) \cong \sum_{n \in \mathbb{N}} \left(\mathbf{3} + \begin{cases} \eta + \mathbf{2} + \eta & (n \in A) \\ \eta & (n \notin A) \end{cases} \right)$ we get

$n \in A \iff (L, \leqslant_{\mathrm{lex}}) \models$ between intervals of size 3 no. n and $n+1$, there is some finite interval of size 2

Theorem

There exists a one-counter language L s.t. the Σ_3-theory of $(L, \leqslant_{\mathrm{lex}})$ is undecidable.

Proof.

with $(L, \leqslant_{\mathrm{lex}}) \cong \sum_{n\in\mathbb{N}} \left(\mathbf{3} + \begin{cases} \eta + \mathbf{2} + \eta & (n \in A) \\ \eta & (n \notin A) \end{cases}\right)$ we get

$$n \in A \iff (L, \leqslant_{\mathrm{lex}}) \models \begin{array}{l} \text{between intervals of size 3 no. } n \text{ and } n+1, \\ \text{there is some finite interval of size 2} \end{array}$$

which is a Σ_3-statement. $\square$

Theorem

1. There exists
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\mathrm{lex}}) \models \varphi$ is undecidable.

Theorem

1. There exist
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\mathrm{lex}}) \models \varphi$ is undecidable,
   1.2 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1 \cup L_2, \leqslant_{\mathrm{lex}})$ is undecidable.

Theorem

1. There exist
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\text{lex}}) \models \varphi$ is undecidable,
   1.2 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1 \cup L_2, \leqslant_{\text{lex}})$ is undecidable.

2. There is no linear order $\mathcal{L}$ such that the Σ_2-theory of $\mathcal{L}$ is undecidable.

Theorem

1. There exist
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\mathrm{lex}}) \models \varphi$ is undecidable,
   1.2 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1 \cup L_2, \leqslant_{\mathrm{lex}})$ is undecidable, and
   1.3 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the FO-theory of $(L_1 \cup L_2, \leqslant_{\mathrm{lex}})$ is not arithmetical.
2. There is no linear order $\mathcal{L}$ such that the Σ_2-theory of $\mathcal{L}$ is undecidable.

Theorem

1. There exist
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\mathrm{lex}}) \models \varphi$ is undecidable,
   1.2 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1 \cup L_2, \leqslant_{\mathrm{lex}})$ is undecidable, and
   1.3 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the FO-theory of $(L_1 \cup L_2, \leqslant_{\mathrm{lex}})$ is not arithmetical.
2. There is no linear order $\mathcal{L}$ such that the Σ_2-theory of $\mathcal{L}$ is undecidable.

A non-modest summary

These results cannot be improved any further.

Theorem

1. There exist
   1.1 $\varphi \in \Sigma_2$ such that the set of pairs of disjoint deterministic real-time 1-counter languages L_1 and L_2 with $(L_1 \cup L_2, \leqslant_{\text{lex}}) \models \varphi$ is undecidable,
   1.2 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1 \cup L_2, \leqslant_{\text{lex}})$ is undecidable, and
   1.3 a pair of disjoint deterministic real-time 1-counter languages L_1 and L_2 such that the FO-theory of $(L_1 \cup L_2, \leqslant_{\text{lex}})$ is not arithmetical.
2. There is no linear order $\mathcal{L}$ such that the Σ_2-theory of $\mathcal{L}$ is undecidable.

A non-modest summary

These results cannot be improved any further (if you hesitate to consider blind one-counter automata ...).