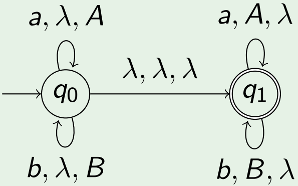
Recent advances on valence automata as a generalization of automata with storage

Phoebe Buckheister Georg Zetzsche

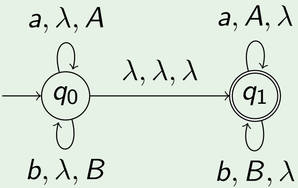
Technische Universität Kaiserslautern

Theorietag 2013

Example (Pushdown automaton)

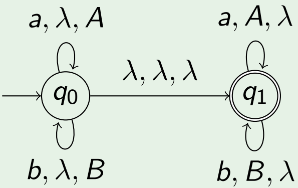


Example (Pushdown automaton)



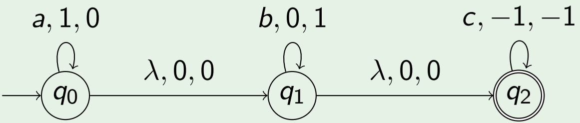
$$L = \{ww^{\text{rev}} \mid w \in \{a, b\}^*\}$$

Example (Pushdown automaton)

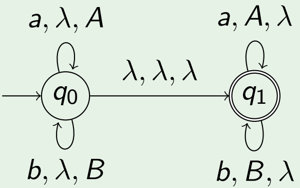


$$L = \{ww^{\text{rev}} \mid w \in \{a, b\}^*\}$$

Example (Blind counter automaton)

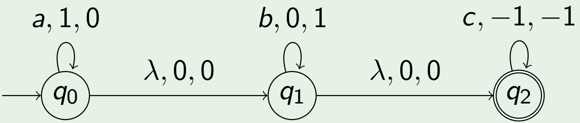


Example (Pushdown automaton)



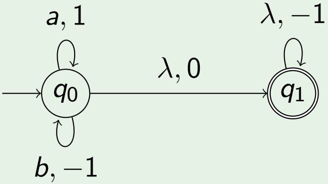
$$L = \{ww^{\text{rev}} \mid w \in \{a, b\}^*\}$$

Example (Blind counter automaton)

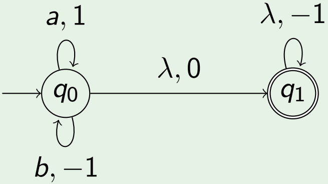


$$L = \{a^n b^n c^n \mid n \geqslant 0\}$$

Example (Partially blind counter automaton)



Example (Partially blind counter automaton)



$$L = \{w \in \{a,b\}^* \quad | \quad |p|_a \geqslant |p|_b \text{ for any prefix } p \text{ of } w\}$$

Automata models that extend finite automata by some storage mechanism:

- Pushdown automata
- Blind counter automata
- Partially blind counter automata
- Turing machines

Automata models that extend finite automata by some storage mechanism:

- Pushdown automata
- Blind counter automata
- Partially blind counter automata
- Turing machines

Each storage mechanism consists of:

- States: set S of states
- Operations: partial maps $\alpha_1, \ldots, \alpha_n : S \to S$

Model	States	Operations
Pushdown automata	$S = \Gamma^*$	$\mathrm{push}_a : w \mapsto wa,\ a \in \Gamma$ $\mathrm{pop}_a : wa \mapsto w,\ a \in \Gamma$

Model	States	Operations
Pushdown automata	$S = \Gamma^*$	$\mathrm{push}_a : w \mapsto wa,\ a \in \Gamma$ $\mathrm{pop}_a : wa \mapsto w,\ a \in \Gamma$
Blind counter automata	$S = \mathbb{Z}^n$	$\mathrm{inc}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i + 1, \ldots, x_n)$ $\mathrm{dec}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i - 1, \ldots, x_n)$

Model	States	Operations
Pushdown automata	$S = \Gamma^*$	$\mathrm{push}_a : w \mapsto wa,\ a \in \Gamma$ $\mathrm{pop}_a : wa \mapsto w,\ a \in \Gamma$
Blind counter automata	$S = \mathbb{Z}^n$	$\mathrm{inc}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i + 1, \ldots, x_n)$ $\mathrm{dec}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i - 1, \ldots, x_n)$
Partially blind counter automata	$S = \mathbb{N}^n$	$\mathrm{inc}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i + 1, \ldots, x_n)$ $\mathrm{dec}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i - 1, \ldots, x_n)$

Model	States	Operations
Pushdown automata	$S = \Gamma^*$	$\text{push}_a : w \mapsto wa,\ a \in \Gamma$ $\text{pop}_a : wa \mapsto w,\ a \in \Gamma$
Blind counter automata	$S = \mathbb{Z}^n$	$\text{inc}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i + 1, \ldots, x_n)$ $\text{dec}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i - 1, \ldots, x_n)$
Partially blind counter automata	$S = \mathbb{N}^n$	$\text{inc}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i + 1, \ldots, x_n)$ $\text{dec}_i : (x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i - 1, \ldots, x_n)$

Observation

Here, a sequence $\beta_1, \ldots, \beta_k$ of operations is valid if and only if

$$\beta_1 \circ \cdots \circ \beta_k = \text{id}$$

Definition

A *monoid* is

- a set M together with
- an associative binary operation $\cdot : M \times M \to M$ and
- a neutral element $1 \in M$ ($a1 = 1a = a$ for any $a \in M$).

Definition

A *monoid* is

- a set M together with
- an associative binary operation $\cdot : M \times M \to M$ and
- a neutral element $1 \in M$ ($a1 = 1a = a$ for any $a \in M$).

Storage mechanisms as monoids

- Let S be a set of states and $\alpha_1, \ldots, \alpha_n : S \to S$ partial maps.
- The set of all compositions of $\alpha_1, \ldots, \alpha_n$ is a monoid M.
- The identity map is the neutral element of M.
- M is a decription of the storage mechanism.

Valence automata

Common generalization: Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.

Valence automata

Common generalization: Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and

Valence automata

Common generalization: Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and
  - $m_1 \cdots m_n = 1$.

Valence automata

Common generalization: Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and
  - $m_1 \cdots m_n = 1$.

Language class

$\mathrm{VA}(M)$ languages accepted by valence automata over M.

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed?
- For which are all languages context-free?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed?
- For which are all languages context-free?
- For which can we decide, for example, emptiness?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed?
- For which are all languages context-free?
- For which can we decide, for example, emptiness?
- Which language classes can be obtained this way?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed? ← tomorrow
- For which are all languages context-free?
- For which can we decide, for example, emptiness?
- Which language classes can be obtained this way?

Classical results can now be generalized:

Questions

- Which storage mechanisms increase the expressive power?
- For which can we avoid silent transitions?
- For which do we have semilinearity of all languages?
- For which is the language class Boolean closed? ← tomorrow
- For which are all languages context-free?
- For which can we decide, for example, emptiness?
- Which language classes can be obtained this way?

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.
Let $\Gamma = (V, E)$ be a graph. Let

$$X_\Gamma = \{a_v, \bar{a}_v \mid v \in V\}$$

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.
Let $\Gamma = (V, E)$ be a graph. Let

$$X_\Gamma = \{a_v, \bar{a}_v \mid v \in V\}$$
$$R_\Gamma = \{a_v \bar{a}_v = 1 \mid v \in V\}$$

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.
Let $\Gamma = (V, E)$ be a graph. Let

$$\begin{aligned} X_\Gamma &= \{a_v, \bar{a}_v \mid v \in V\} \\ R_\Gamma &= \{a_v\bar{a}_v = 1 \mid v \in V\} \\ &\quad \cup \{xy = yx \mid x \in \{a_u, \bar{a}_u\}, y \in \{a_v, \bar{a}_v\}, \{u, v\} \in E\} \end{aligned}$$

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.
Let $\Gamma = (V, E)$ be a graph. Let

$$X_\Gamma = \{a_v, \bar{a}_v \mid v \in V\}$$
$$R_\Gamma = \{a_v\bar{a}_v = 1 \mid v \in V\}$$
$$\cup \{xy = yx \mid x \in \{a_u, \bar{a}_u\}, y \in \{a_v, \bar{a}_v\}, \{u, v\} \in E\}$$

$$\mathbb{M}\Gamma = X_\Gamma^*/R_\Gamma$$

Monoids defined by graphs

By graphs, we mean undirected graphs with loops allowed.
Let $\Gamma = (V, E)$ be a graph. Let

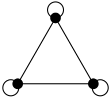
$$\begin{aligned} X_\Gamma &= \{a_v, \bar{a}_v \mid v \in V\} \\ R_\Gamma &= \{a_v \bar{a}_v = 1 \mid v \in V\} \\ &\quad \cup \{xy = yx \mid x \in \{a_u, \bar{a}_u\}, y \in \{a_v, \bar{a}_v\}, \{u, v\} \in E\} \end{aligned}$$

$$\mathbb{M}\Gamma = X_\Gamma^* / R_\Gamma$$

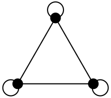
Intuition

- $\mathbb{B}$: bicyclic monoid, $\mathbb{B} = \{a, \bar{a}\}^* / \{a\bar{a} = 1\}$.
- $\mathbb{Z}$: group of integers
- For each unlooped vertex, we have a copy of $\mathbb{B}$
- For each looped vertex, we have a copy of $\mathbb{Z}$
- $\mathbb{M}\Gamma$ consists of sequences of such elements
- An edge between vertices means that elements can commute

Examples

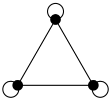


Examples



$\mathbb{Z}^3$

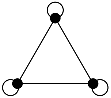
Examples



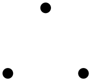
$\mathbb{Z}^3$

Blind multicounter

Examples

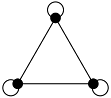


$\mathbb{Z}^3$



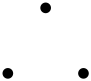
Blind multicounter

Examples



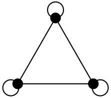
$\mathbb{Z}^3$

Blind multicounter



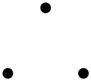
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Examples



$\mathbb{Z}^3$

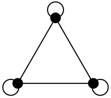
Blind multicounter



$\mathbb{B} * \mathbb{B} * \mathbb{B}$

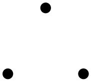
Pushdown

Examples



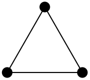
$\mathbb{Z}^3$

Blind multicounter

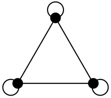


$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown

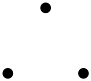


Examples



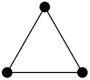
$\mathbb{Z}^3$

Blind multicounter



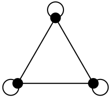
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



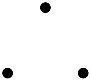
$\mathbb{B}^3$

Examples



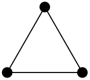
$\mathbb{Z}^3$

Blind multicounter



$\mathbb{B} * \mathbb{B} * \mathbb{B}$

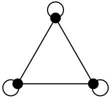
Pushdown



$\mathbb{B}^3$

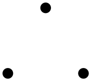
Partially blind multicounter

Examples



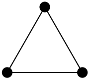
$\mathbb{Z}^3$

Blind multicounter



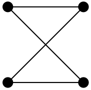
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown

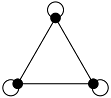


$\mathbb{B}^3$

Partially blind multicounter

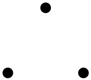


Examples



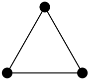
$\mathbb{Z}^3$

Blind multicounter



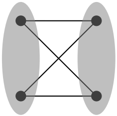
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown

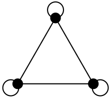


$\mathbb{B}^3$

Partially blind multicounter

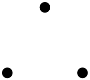


Examples



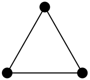
$\mathbb{Z}^3$

Blind multicounter



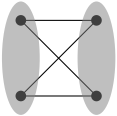
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



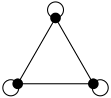
$\mathbb{B}^3$

Partially blind multicounter



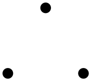
$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

Examples



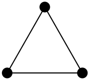
$\mathbb{Z}^3$

Blind multicounter



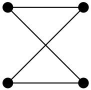
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



$\mathbb{B}^3$

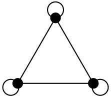
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

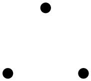
Infinite tape (TM)

Examples



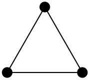
$\mathbb{Z}^3$

Blind multicounter



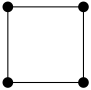
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



$\mathbb{B}^3$

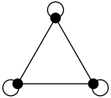
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

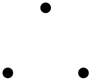
Infinite tape (TM)

Examples



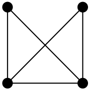
$\mathbb{Z}^3$

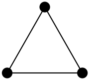
Blind multicounter



$\mathbb{B} * \mathbb{B} * \mathbb{B}$

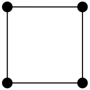
Pushdown





$\mathbb{B}^3$

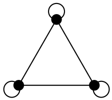
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

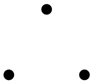
Infinite tape (TM)

Examples



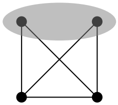
$\mathbb{Z}^3$

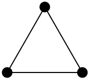
Blind multicounter



$\mathbb{B} * \mathbb{B} * \mathbb{B}$

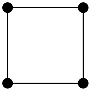
Pushdown





$\mathbb{B}^3$

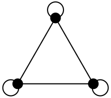
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

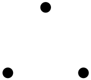
Infinite tape (TM)

Examples



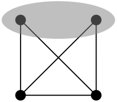
$\mathbb{Z}^3$

Blind multicounter

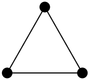


$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown

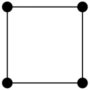


$(\mathbb{B} * \mathbb{B}) \times \mathbb{B} \times \mathbb{B}$



$\mathbb{B}^3$

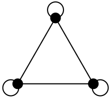
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

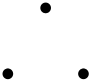
Infinite tape (TM)

Examples



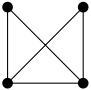
$\mathbb{Z}^3$

Blind multicounter



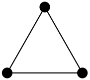
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



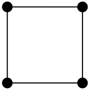
$(\mathbb{B} * \mathbb{B}) \times \mathbb{B} \times \mathbb{B}$

Pushdown + partially blind counters



$\mathbb{B}^3$

Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

Infinite tape (TM)

Silent Transitions

A transition that reads no input is called *silent transition* or λ-*transition*.

Silent Transitions

A transition that reads no input is called *silent transition* or λ-*transition*.

$\mathsf{VA}^+(M)$ languages accepted by VA over M without silent transitions

Silent Transitions

A transition that reads no input is called *silent transition* or λ-*transition*.

$\mathsf{VA}^+(M)$ languages accepted by VA over M without silent transitions

Important problem

- When can silent transitions be eliminated?
- Without silent transitions, decide membership using exponential number of storage computations.
- Elimination can be regarded as a precomputation.

Silent Transitions

A transition that reads no input is called *silent transition* or λ-*transition*.

$\mathsf{VA}^+(M)$ languages accepted by VA over M without silent transitions

Important problem

- When can silent transitions be eliminated?
- Without silent transitions, decide membership using exponential number of storage computations.
- Elimination can be regarded as a precomputation.

Question

For which storage mechanisms can we avoid silent transitions?

Silent Transitions

A transition that reads no input is called *silent transition* or λ-*transition*.

$\mathsf{VA}^+(M)$ languages accepted by VA over M without silent transitions

Important problem

- When can silent transitions be eliminated?
- Without silent transitions, decide membership using exponential number of storage computations.
- Elimination can be regarded as a precomputation.

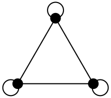
Question

For which storage mechanisms can we avoid silent transitions?

Known so far

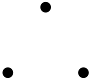
- Pushdown automata (Greibach 1965)
- Blind counter automata (Greibach 1978)
- Partially blind counter automata (Greibach 1978 / Jantzen 1979)

Examples, again



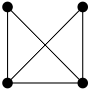
$\mathbb{Z}^3$

Blind multicounter



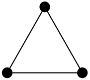
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



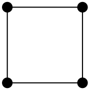
$(\mathbb{B} * \mathbb{B}) \times \mathbb{B} \times \mathbb{B}$

Pushdown + partially blind counters



$\mathbb{B}^3$

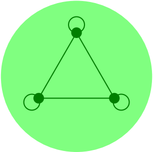
Partially blind multicounter



$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

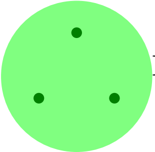
Infinite tape (TM)

Examples, again



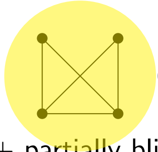
$\mathbb{Z}^3$

Blind multicounter



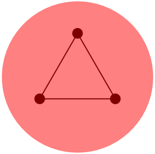
$\mathbb{B} * \mathbb{B} * \mathbb{B}$

Pushdown



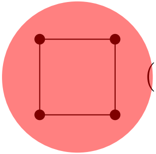
$(\mathbb{B} * \mathbb{B}) \times \mathbb{B} \times \mathbb{B}$

Pushdown + partially blind counters



$\mathbb{B}^3$

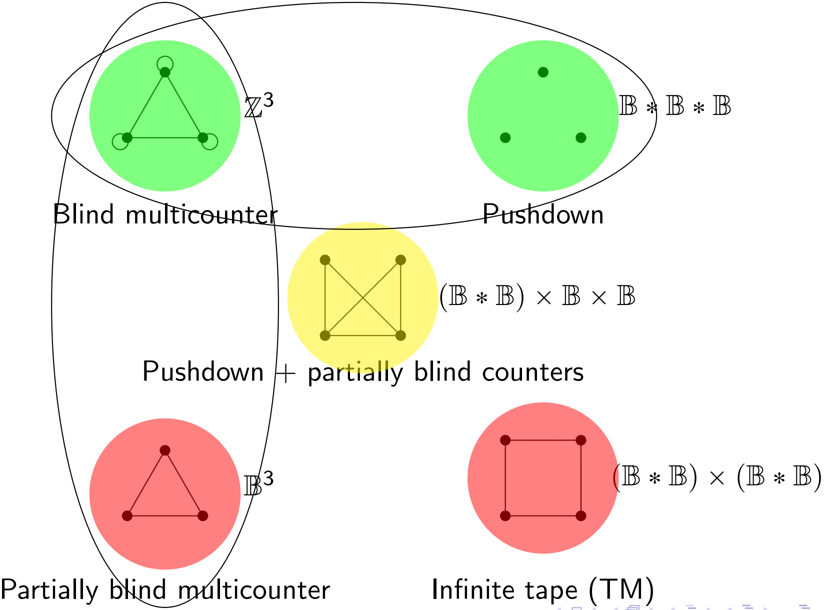
Partially blind multicounter

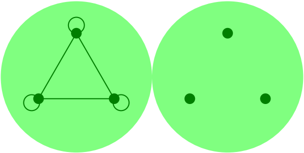


$(\mathbb{B} * \mathbb{B}) \times (\mathbb{B} * \mathbb{B})$

Infinite tape (TM)

Examples, again

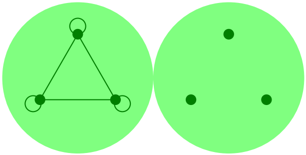


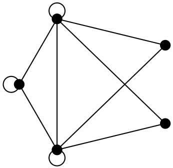


Theorem (Z., ICALP 2013)

Let Γ be a graph such that

- *any two looped vertices are adjacent,*
- *no two unlooped vertices are adjacent.*

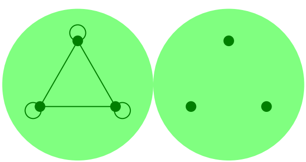


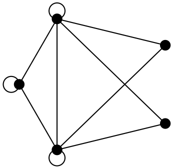


Theorem (Z., ICALP 2013)

Let Γ be a graph such that

- *any two looped vertices are adjacent,*
- *no two unlooped vertices are adjacent.*





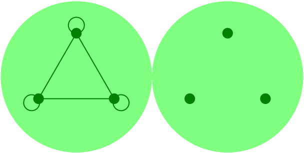
Theorem (Z., ICALP 2013)

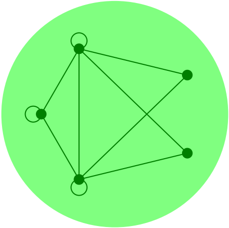
Let Γ *be a graph such that*

- *any two looped vertices are adjacent,*
- *no two unlooped vertices are adjacent.*

Then $\mathsf{VA}(\mathbb{M}\Gamma) = \mathsf{VA}^{+}(\mathbb{M}\Gamma)$ *if and only if* Γ *does not contain*

as an induced subgraph.





Theorem (Z., ICALP 2013)

Let Γ *be a graph such that*

- *any two looped vertices are adjacent,*
- *no two unlooped vertices are adjacent.*

Then $\mathsf{VA}(\mathbb{M}\Gamma) = \mathsf{VA}^{+}(\mathbb{M}\Gamma)$ *if and only if* Γ *does not contain*

as an induced subgraph.

Positive case

Definition

Let $\mathcal{C}$ be the smallest class of monoids such that

- $1 \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M \times \mathbb{Z} \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M * \mathbb{B} \in \mathcal{C}$

Positive case

Definition

Let $\mathcal{C}$ be the smallest class of monoids such that

- $1 \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M \times \mathbb{Z} \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M * \mathbb{B} \in \mathcal{C}$

Lemma

Let Γ be a graph such that

- *any two looped vertices are adjacent,*
- *no two unlooped vertices are adjacent,*
-  *does not appear as an induced subgraph.*

Then, $\mathbb{M}\Gamma \in \mathcal{C}$.

Positive case

Definition

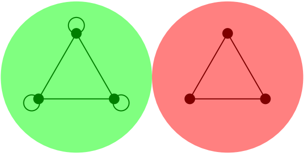
Let $\mathcal{C}$ be the smallest class of monoids such that

- $1 \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M \times \mathbb{Z} \in \mathcal{C}$
- if $M \in \mathcal{C}$, then $M * \mathbb{B} \in \mathcal{C}$

Interpretation of $\mathcal{C}$

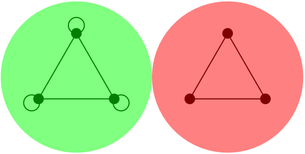
$\mathcal{C}$ corresponds to the class of storage mechanisms obtained by

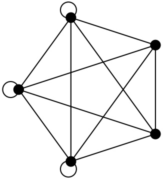
- adding a blind counter ($M \times \mathbb{Z}$) and
- building stacks ($M * \mathbb{B}$).



Theorem (Z., ICALP 2013)

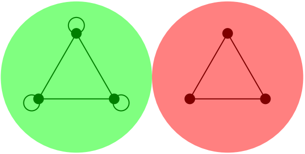
Let Γ be a graph such that between any two distinct vertices, there is an edge.

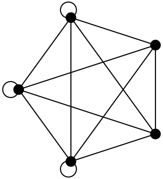




Theorem (Z., ICALP 2013)

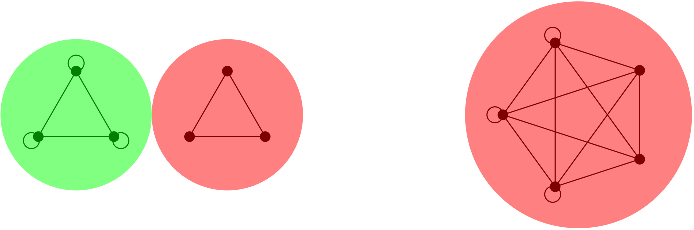
Let Γ be a graph such that between any two distinct vertices, there is an edge.





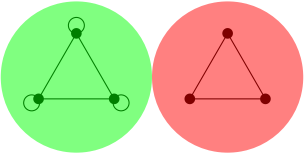
Theorem (Z., ICALP 2013)

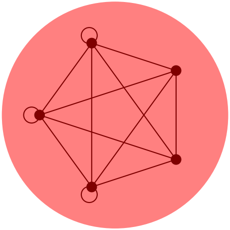
Let Γ be a graph such that between any two distinct vertices, there is an edge. Then $\mathsf{VA}(\mathbb{M}\Gamma) = \mathsf{VA}^+(\mathbb{M}\Gamma)$ *if and only if the number of unlooped nodes is* $\leqslant 1$.



Theorem (Z., ICALP 2013)

Let Γ *be a graph such that between any two distinct vertices, there is an edge. Then* $\mathsf{VA}(\mathbb{M}\Gamma) = \mathsf{VA}^{+}(\mathbb{M}\Gamma)$ *if and only if the number of unlooped nodes is* $\leqslant 1$.





Theorem (Z., ICALP 2013)

Let Γ *be a graph such that between any two distinct vertices, there is an edge. Then* $\mathsf{VA}(\mathbb{M}\Gamma) = \mathsf{VA}^+(\mathbb{M}\Gamma)$ *if and only if the number of unlooped nodes is* $\leqslant 1$. *In other words:*

$$\mathsf{VA}(\mathbb{B}^r \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s) \quad \textit{iff} \quad r \leqslant 1.$$

Observation

$\mathsf{VA}(\mathbb{B} \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B} \times \mathbb{Z}^s)$ already follows from the first theorem.

Observation

$\mathsf{VA}(\mathbb{B} \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B} \times \mathbb{Z}^s)$ already follows from the first theorem.

Proving $\mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s) \subsetneq \mathsf{VA}(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r \geqslant 2$

- Use Greibach's and Jantzen's language

$$L_1 = \{wc^n \mid w \in \{0,1\}^*, n \leqslant \mathrm{bin}(w)\},$$

$$\mathrm{bin}(v0) = 2 \cdot \mathrm{bin}(v), \quad \mathrm{bin}(v1) = 2 \cdot \mathrm{bin}(v) + 1, \quad \mathrm{bin}(\lambda) = 0.$$

Observation

$\mathsf{VA}(\mathbb{B} \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B} \times \mathbb{Z}^s)$ already follows from the first theorem.

Proving $\mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s) \subsetneq \mathsf{VA}(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r \geqslant 2$

- Use Greibach's and Jantzen's language

$$L_1 = \{wc^n \mid w \in \{0,1\}^*, n \leqslant \mathrm{bin}(w)\},$$

$$\mathrm{bin}(v0) = 2 \cdot \mathrm{bin}(v), \quad \mathrm{bin}(v1) = 2 \cdot \mathrm{bin}(v) + 1, \quad \mathrm{bin}(\lambda) = 0.$$

- Known that $L_1 \in \mathsf{VA}(\mathbb{B}^2)$.
- Show $L_1 \notin \mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r, s \in \mathbb{N}$.

Observation

$\mathsf{VA}(\mathbb{B} \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B} \times \mathbb{Z}^s)$ already follows from the first theorem.

Proving $\mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s) \subsetneq \mathsf{VA}(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r \geqslant 2$

- Use Greibach's and Jantzen's language

$$L_1 = \{wc^n \mid w \in \{0,1\}^*, n \leqslant \mathrm{bin}(w)\},$$

$$\mathrm{bin}(v0) = 2 \cdot \mathrm{bin}(v), \quad \mathrm{bin}(v1) = 2 \cdot \mathrm{bin}(v) + 1, \quad \mathrm{bin}(\lambda) = 0.$$

- Known that $L_1 \in \mathsf{VA}(\mathbb{B}^2)$.
- Show $L_1 \notin \mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r, s \in \mathbb{N}$.
- Count *fooling sets*, concept from state complexity.

Observation

$\mathsf{VA}(\mathbb{B} \times \mathbb{Z}^s) = \mathsf{VA}^+(\mathbb{B} \times \mathbb{Z}^s)$ already follows from the first theorem.

Proving $\mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s) \subsetneq \mathsf{VA}(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r \geqslant 2$

- Use Greibach's and Jantzen's language

$$L_1 = \{wc^n \mid w \in \{0,1\}^*, n \leqslant \mathrm{bin}(w)\},$$

$$\mathrm{bin}(v0) = 2 \cdot \mathrm{bin}(v), \quad \mathrm{bin}(v1) = 2 \cdot \mathrm{bin}(v) + 1, \quad \mathrm{bin}(\lambda) = 0.$$

- Known that $L_1 \in \mathsf{VA}(\mathbb{B}^2)$.
- Show $L_1 \notin \mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s)$ for $r, s \in \mathbb{N}$.
- Count *fooling sets*, concept from state complexity.
- Languages in $\mathsf{VA}^+(\mathbb{B}^r \times \mathbb{Z}^s)$ have polynomially many fooling sets
- L_1 has exponential number of fooling sets

Semilinearity I

For which monoids M are all languages in $\mathsf{VA}(M)$ semilinear?

- Parikh's Theorem: Pushdown automata
- Ibarra + Greibach: Blind counter automata

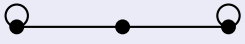
Semilinearity I

For which monoids M are all languages in $\mathsf{VA}(M)$ semilinear?

- Parikh's Theorem: Pushdown automata
- Ibarra + Greibach: Blind counter automata

Theorem (Buckheister, Z., MFCS 2013)

All languages in $\mathsf{VA}(\mathbb{M}\Gamma)$ *are semilinear if and only if*

1. Γ *contains neither*  *nor*  *as an induced subgraph and*
2. Γ*, minus loops, is a transitive forest.*

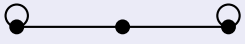
Semilinearity I

For which monoids M are all languages in $\mathsf{VA}(M)$ semilinear?

- Parikh's Theorem: Pushdown automata
- Ibarra + Greibach: Blind counter automata

Theorem (Buckheister, Z., MFCS 2013)

All languages in $\mathsf{VA}(\mathbb{M}\Gamma)$ *are semilinear if and only if*

1. Γ *contains neither*  *nor*  *as an induced subgraph and*
2. Γ, *minus loops, is a transitive forest.*

- For 4 forbidden induced subgraphs, non-semilinear languages from Petri net and trace theory

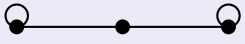
Semilinearity I

For which monoids M are all languages in $\mathsf{VA}(M)$ semilinear?

- Parikh's Theorem: Pushdown automata
- Ibarra + Greibach: Blind counter automata

Theorem (Buckheister, Z., MFCS 2013)

All languages in $\mathsf{VA}(\mathbb{M}\Gamma)$ *are semilinear if and only if*

1. Γ *contains neither*  *nor*  *as an induced subgraph and*
2. Γ, *minus loops, is a transitive forest.*

- For 4 forbidden induced subgraphs, non-semilinear languages from Petri net and trace theory
- $\mathsf{VA}(\mathbb{B}) \subseteq \mathsf{CF}$
- $M \mapsto M \times \mathbb{Z}$, $(M, M') \mapsto M * M'$ preserve semilinearity

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathsf{VA}(M) = \mathsf{REG}$
- $\mathsf{VA}(M) = \mathsf{VA}(G)$ *for a torsion group G*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages*

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathsf{VA}(M) = \mathsf{REG}$
- $\mathsf{VA}(M) = \mathsf{VA}(G)$ *for a torsion group G*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages*

Theorem (Buckheister, Z., MFCS 2013)

For every torsion group G, all languages in $\mathsf{VA}(G)$ are semilinear.

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathrm{VA}(M) = \mathrm{REG}$
- $\mathrm{VA}(M) = \mathrm{VA}(G)$ *for a torsion group G*
- $\mathrm{VA}(M)$ *contains the blind one-counter languages*
- $\mathrm{VA}(M)$ *contains the partially blind one-counter languages*

Theorem (Buckheister, Z., MFCS 2013)

For every torsion group G, all languages in $\mathrm{VA}(G)$ are semilinear.

- Non-effective construction!

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathsf{VA}(M) = \mathsf{REG}$
- $\mathsf{VA}(M) = \mathsf{VA}(G)$ *for a torsion group G*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages*

Theorem (Buckheister, Z., MFCS 2013)

For every torsion group G, all languages in $\mathsf{VA}(G)$ are semilinear.

- Non-effective construction! (Cannot be made effective.)

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathrm{VA}(M) = \mathrm{REG}$
- $\mathrm{VA}(M) = \mathrm{VA}(G)$ *for a torsion group G*
- $\mathrm{VA}(M)$ *contains the blind one-counter languages*
- $\mathrm{VA}(M)$ *contains the partially blind one-counter languages*

Theorem (Buckheister, Z., MFCS 2013)

For every torsion group G, all languages in $\mathrm{VA}(G)$ are semilinear.

- Non-effective construction! (Cannot be made effective.)
- Decompose computations into loops (and rest of bounded length).

Semilinearity II

A group G is called a *torsion group* if for every $g \in G$, there is a $k \in \mathbb{N} \setminus \{0\}$ with $g^k = 1$.

Theorem (Render 2010)

For every monoid M, at least one of the following holds:

- $\mathsf{VA}(M) = \mathsf{REG}$
- $\mathsf{VA}(M) = \mathsf{VA}(G)$ *for a torsion group G*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages*

Theorem (Buckheister, Z., MFCS 2013)

For every torsion group G, all languages in $\mathsf{VA}(G)$ are semilinear.

- Non-effective construction! (Cannot be made effective.)
- Decompose computations into loops (and rest of bounded length).
- Set of vectors counting loops is upward-closed w.r.t. some WQO.

Semilinearity II

Theorem

For every monoid M, at least one of the following holds:

- *All languages in* $\mathsf{VA}(M)$ *are semilinear.*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages.*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages.*

Semilinearity II

Theorem

For every monoid M, at least one of the following holds:

- *All languages in* $\mathsf{VA}(M)$ *are semilinear.*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages.*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages.*

Observation

$\mathsf{VA}(M)$ always full semi-AFL (closed under rational transductions, union).

Semilinearity II

Theorem

For every monoid M, at least one of the following holds:

- *All languages in* $\mathsf{VA}(M)$ *are semilinear.*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages.*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages.*

Observation

$\mathsf{VA}(M)$ always full semi-AFL (closed under rational transductions, union).

Consequence

Let $S = \{a^n \mid n \text{ is a square}\}$. Let $\mathcal{T}$ be the smallest full semi-AFL containing S. Then $\mathcal{T}$ does not arise as $\mathsf{VA}(M)$ from a monoid M.

Semilinearity II

Theorem

For every monoid M, at least one of the following holds:

- *All languages in* $\mathsf{VA}(M)$ *are semilinear.*
- $\mathsf{VA}(M)$ *contains the blind one-counter languages.*
- $\mathsf{VA}(M)$ *contains the partially blind one-counter languages.*

Observation

$\mathsf{VA}(M)$ always full semi-AFL (closed under rational transductions, union).

Consequence

Let $S = \{a^n \mid n \text{ is a square}\}$. Let $\mathcal{T}$ be the smallest full semi-AFL containing S. Then $\mathcal{T}$ does not arise as $\mathsf{VA}(M)$ from a monoid M.

- S is not semilinear
- $\mathcal{T}$ does not contain $\{a^n b^n \mid n \geqslant 0\}$.

Conclusion

- Avoidability of silent transitions

Conclusion

- Avoidability of silent transitions
- Generalizations of Parikh's Theorem

Conclusion

- Avoidability of silent transitions
- Generalizations of Parikh's Theorem
- Language classes arising as $\mathsf{VA}(M)$

Conclusion

- Avoidability of silent transitions
- Generalizations of Parikh's Theorem
- Language classes arising as $\mathsf{VA}(M)$

More classical results can be generalized:

Ongoing work

- Computability of the downward closure (scattered subwords)?

Conclusion

- Avoidability of silent transitions
- Generalizations of Parikh's Theorem
- Language classes arising as $\mathsf{VA}(M)$

More classical results can be generalized:

Ongoing work

- Computability of the downward closure (scattered subwords)?
- Decidability of questions for Büchi variants.