Emmanuel Filiot

Nicolas Mazzocchi

Jean-François Raskin

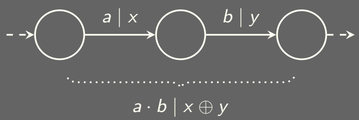
A Pattern Logic for Automata with Outputs

Université libre de Bruxelles

Highlights 2018 - Berlin

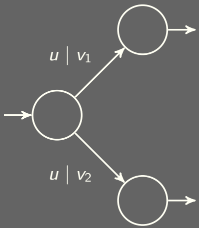
Automata with outputs in $(D, \oplus, \mathbb{0})$

Transition sequence



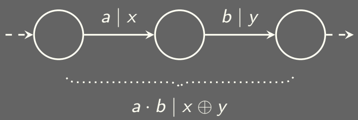
$$[\![A]\!] \subseteq \Sigma^* \times D$$

Non-determinism

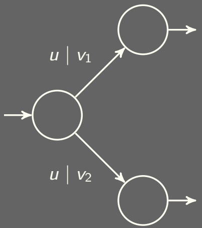


Automata with outputs in $(D, \oplus, \mathbb{0})$

Transition sequence



Non-determinism



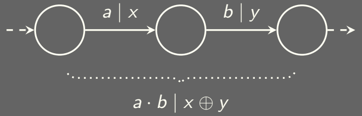
$$[\![A]\!] \subseteq \Sigma^* \times D$$

Example

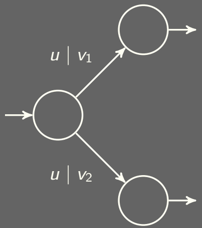
- Sum-automata over $(\mathbb{Z}, +, 0)$
- Transducers over $(\Gamma^*, \cdot, \varepsilon)$

Automata with outputs in $(D, \oplus, \mathbb{0})$

Transition sequence



Non-determinism



$$[\![A]\!] \subseteq \Sigma^* \times D$$

Classical problems

- Equivalence $[\![A]\!] = [\![B]\!]$
- Inclusion $[\![A]\!] \subseteq [\![B]\!]$

Example

- Sum-automata over $(\mathbb{Z}, +, 0)$
- Transducers over $(\Gamma^*, \cdot, \varepsilon)$

Subclasses of automata

Why?

- Recover decidability
- Improve complexity

Subclasses of automata

Why?

- Recover decidability
- Improve complexity

Class membership problem

1. (challenging) structural characterisation of the subclass
2. (ad-hoc) decision procedure for the subclass (Model-Checking)

Subclasses of automata

Why?

- Recover decidability
- Improve complexity

Class membership problem

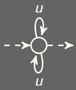
1. (challenging) structural characterisation of the subclass
2. (ad-hoc) decision procedure for the subclass (Model-Checking)

Examples

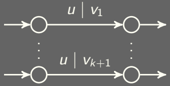
- **Sequentiality**, input determinism
- **Ambiguity**, bound on the number of accepting runs for any input
- **Valuedness**, bound on the number of output values for any input

Structural properties in literature

Exp.-ambiguity

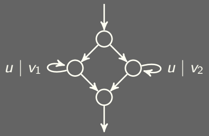


Non k-valuedness



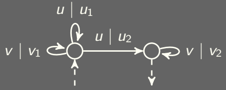
$$\forall i \neq j, v_i \neq v_j$$

Co-terminal circuits



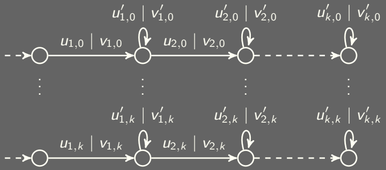
$$|v_1| \neq |v_2|$$

Fork property



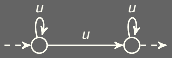
$$\delta(v_1, v_2) \neq \delta(v_1 v_1', v_2 v_2')$$

Branching Twinning property of order k

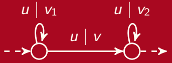


$$\bigwedge_{j\neq j'} \bigvee_{i=1}^{k} \bigwedge_{i'=1}^{i} \bigwedge \begin{cases} u_{i',j} = u_{i',j'} \\ u'_{i',j} = u'_{i',j'} \\ \delta(v_{1,j}\dots v_{i,j}, v_{1,j}\dots v_{i,j} v'_{i,j}) \\ \neq \\ \delta(v_{1,j'}\dots v_{i,j'}, v_{1,j'}\dots v_{i,j'} v'_{i,j'}) \end{cases}$$

Non-Finite ambiguity

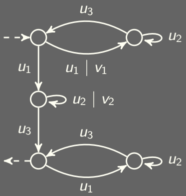


Dumbbell computation



$$v_1 v \neq v v_2$$

W computation



$$|v_1| \neq |v_2|$$

Definition of pattern logic: PL

A pattern formula over a set of output predicates $\mathcal{O}$

$$\varphi ::= (\exists \pi_1 = p_1 \xrightarrow{u_1 | v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n | v_n} q_n), \mathcal{C}$$

$$\mathcal{C} ::= \neg \mathcal{C} \mid \mathcal{C} \vee \mathcal{C} \mid \mathcal{C} \wedge \mathcal{C} \mid P$$

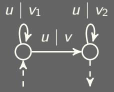
Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
Path	$\pi = \pi' \mid q = q' \mid \mathrm{init}(q) \mid \mathrm{final}(q)$
Output	$p(t_1, \ldots, t_n)$

- L regular language represented as an NFA
- $t_i \in \mathit{Terms}(\{v_1, \ldots, v_n\}, \oplus, \mathbb{0})$

Example: Dumbbell computation

$$\left(\begin{array}{c} \exists \pi'_1 = q'_1 \rightarrow q_1 \quad \exists \pi \;\; = q_1 \xrightarrow{u \;\, | v} q_2 \quad \exists \pi'_2 = q_2 \rightarrow q'_2 \\ \exists \pi_1 = q_1 \xrightarrow{u_1 | v_1} q_1 \quad \exists \pi_2 = q_2 \xrightarrow{u_2 | v_2} q_2 \end{array} \right)$$

$$\mathrm{init}(q'_1) \wedge \mathrm{final}(q'_2) \wedge u_1 = u \wedge u = u_2 \wedge v_1 \oplus v \neq v \oplus v_2$$



$$v_1 v \neq v v_2$$

Definition of pattern logic: PL

A pattern formula over a set of output predicates $\mathcal{O}$

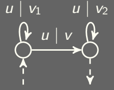
$$\varphi ::= (\exists \pi_1 = p_1 \xrightarrow{u_1 | v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n | v_n} q_n), \mathcal{C}$$
$$\mathcal{C} ::= \neg \mathcal{C} \mid \mathcal{C} \vee \mathcal{C} \mid \mathcal{C} \wedge \mathcal{C} \mid P$$

Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
Path	$\pi = \pi' \mid q = q' \mid \mathrm{init}(q) \mid \mathrm{final}(q)$
Output	$p(t_1, \ldots, t_n)$

- L regular language represented as an NFA
- $t_i \in \mathit{Terms}(\{v_1, \ldots, v_n\}, \oplus, \mathbb{0})$

Example: Dumbbell computation in $\mathrm{PL}^+[\neq]$

$$\begin{pmatrix} \exists \pi'_1 = q'_1 \to q_1 & \exists \pi \ = q_1 \xrightarrow{u \ | v} q_2 & \exists \pi'_2 = q_2 \to q'_2 \\ & \exists \pi_1 = q_1 \xrightarrow{u_1 | v_1} q_1 \quad \exists \pi_2 = q_2 \xrightarrow{u_2 | v_2} q_2 & \end{pmatrix}$$
$$\mathrm{init}(q'_1) \wedge \mathrm{final}(q'_2) \wedge u_1 = u \wedge u = u_2 \wedge v_1 \oplus v \neq v \oplus v_2$$



$v_1 v \neq v v_2$

Way to obtain decidability

$$\boxed{\mathsf{A}} \models \boxed{(\exists \pi_1 = p_1 \xrightarrow{u_1|v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n|v_n} q_n), \mathcal{C}}$$

Way to obtain decidability

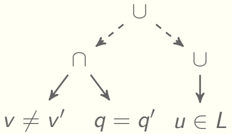
A $\models$ $(\exists \pi_1 = p_1 \xrightarrow{u_1|v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n|v_n} q_n), \mathcal{C}$

Acceptor of paths

n paths of A

$\pi_1 \otimes \ldots \otimes \pi_n$

constraint $\mathcal{C}$



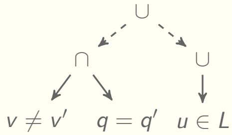
Way to obtain decidability

$$\mathsf{A} \models (\exists \pi_1 = p_1 \xrightarrow{u_1|v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n|v_n} q_n), \mathcal{C}$$

Acceptor of paths

n paths of A $\cap$ constraint $\mathcal{C}$ $\neq \varnothing$

$\pi_1 \otimes \ldots \otimes \pi_n$



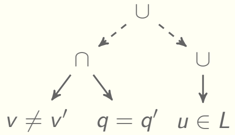
Way to obtain decidability

$$\mathsf{A} \models (\exists \pi_1 = p_1 \xrightarrow{u_1|v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n|v_n} q_n), \mathcal{C}$$

Acceptor of paths

n paths of A $\bigcap$ constraint $\mathcal{C}$ $\neq \varnothing$

$\pi_1 \otimes \ldots \otimes \pi_n$



Sufficient conditions for decidability

- generalise NFA
- decide emptiness
- recognise each predicate (and negation)
- closed under $\cap$ and $\cup$

Complexity Results

Instances

- $\mathrm{PL}_{\mathsf{nfa}}$ defined as $\mathrm{PL}[\varnothing]$ over the trivial monoid
- $\mathrm{PL}_{\mathsf{trans}}$ defined as $\mathrm{PL}^{+}[\not\sqsubseteq, <_{len}, \leq_{len}, \in N, \notin N]$ over $(\Gamma^*, \cdot, \varepsilon)$
- $\mathrm{PL}_{\mathsf{sum}}$ defined as $\mathrm{PL}[\leq, \in S]$ over $(\mathbb{Z}, +, 0)$
- $\mathrm{PL}_{\mathsf{sum}}^{\neq}$ defined as $\mathrm{PL}^{+}[\neq]$ over $(\mathbb{Z}, +, 0)$

Logic \ Setting	General	Fixed Formula
$\mathrm{PL}_{\mathsf{nfa}}$	PSPACE-C	NLOGSPACE-C
$\mathrm{PL}_{\mathsf{trans}}$		NLOGSPACE-C
$\mathrm{PL}_{\mathsf{sum}}$		NP-C binary NLOGSPACE-C unary
$\mathrm{PL}_{\mathsf{sum}}^{\neq}$		PTIME \ NLOGSPACE-H

Thanks