Emmanuel Filiot

Nicolas Mazzocchi

Jean-François Raskin

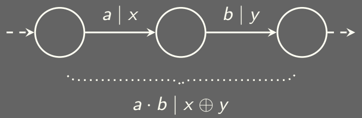
A Pattern Logic for Automata with Outputs

Université libre de Bruxelles

DLT 2018 - Tokyo

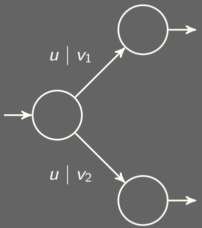
Automata with outputs in $(D, \oplus, \mathbb{0})$

Transition sequence



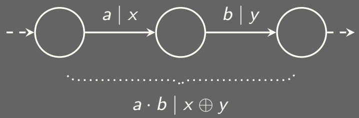
$$[\![A]\!] \subseteq \Sigma^* \times D$$

Non-determinism

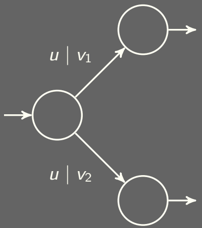


Automata with outputs in $(D, \oplus, \mathbb{0})$

Transition sequence



Non-determinism



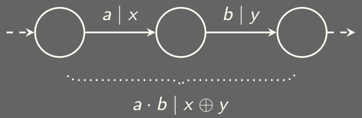
$$[\![A]\!] \subseteq \Sigma^* \times D$$

Example

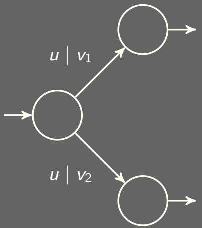
- Sum-automata over $(\mathbb{Z}, +, 0)$
- Transducers over $(\Gamma^*, \cdot, \varepsilon)$

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

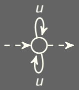
1. (challenging) structural characterisation of the subclass
2. (ad-hoc) decision procedure for the subclass (Model-Checking)

Examples

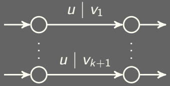
- **Sequentiality**, input determinism
- **Ambiguity**, bound on the number of accepting runs for any input
- **Valuedness**, bound on the number of output values for any input

Structural properties in literature

Exp.-ambiguity

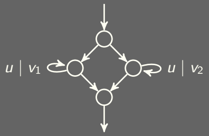


Non k-valuedness



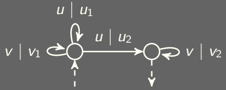
$\forall i \neq j, v_i \neq v_j$

Co-terminal circuits



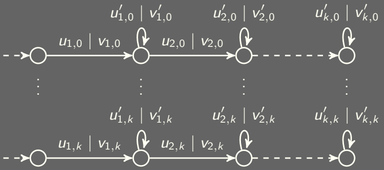
$|v_1| \neq |v_2|$

Fork property



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

Branching Twinning property of order k



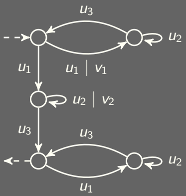
$$\bigwedge_{j\neq j'} \bigvee_{i=1}^{k} \bigwedge_{i'=1}^{i} \bigwedge \begin{cases} u_{i',j} = u_{i',j'} \\ u'_{i',j} = u'_{i',j'} \\ \delta(v_{1,j}\dots v_{i,j},\, v_{1,j}\dots v_{i,j}v'_{i,j}) \\ \neq \\ \delta(v_{1,j'}\dots v_{i,j'},\, v_{1,j'}\dots v_{i,j'}v'_{i,j'}) \end{cases}$$

Non-Finite ambiguity

Dumbbell computation

$v_1 v \neq v v_2$

W computation



$|v_1| \neq |v_2|$

Contributions

❶ Parametric Logic

Sufficient conditions for decidability of the MC problem

❷ Some instantiations

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

Pattern Logic

Definition of pattern logic: $\mathtt{PL}$

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
- $t_i \in \mathit{Terms}(\{v_1, \ldots, v_n\}, \oplus, \mathbb{0})$

Definition of pattern logic: PL

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

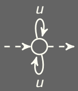
$$\varphi ::= (\exists \pi_1 = p_1 \xrightarrow{u_1 | v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n | v_n} q_n), \mathcal{C}$$
$$\mathcal{C} ::= \neg \mathcal{C} \mid \mathcal{C} \vee \mathcal{C} \mid \mathcal{C} \wedge \mathcal{C} \mid P$$

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: Exponential Ambiguity

$$\begin{pmatrix} \exists \pi_0 = q_0 \longrightarrow q & \\ \exists \pi_1 = q \longrightarrow q_1 & \\ \exists \pi = q \xrightarrow{u_1} q & \exists \pi' = q \xrightarrow{u_2} q \end{pmatrix} \bigwedge \begin{cases} \mathrm{init}(q_0) \\ \mathrm{final}(q_1) \\ \pi \neq \pi' \wedge u_1 = u_2 \end{cases}$$



Definition of pattern logic: PL

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

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

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

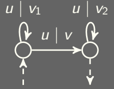
Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
Path	$\pi = \pi' \mid q = q' \mid \mathrm{init}(q) \mid \mathrm{final}(q)$
Output	$p(t_1, \ldots, t_n)$

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

Example: Dumbbell computation

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

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



$v_1 v \neq v v_2$

Definition of pattern logic: PL

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

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

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

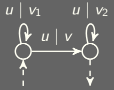
Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
Path	$\pi = \pi' \mid q = q' \mid \mathrm{init}(q) \mid \mathrm{final}(q)$
Output	$p(t_1, \ldots, t_n)$

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

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

$$\begin{pmatrix} \exists \pi_1' = q_1' \rightarrow q_1 & \exists \pi = q_1 \xrightarrow{u \,|\, v} q_2 & \exists \pi_2' = q_2 \rightarrow q_2' \\ \exists \pi_1 = q_1 \xrightarrow{u_1 | v_1} q_1 & & \exists \pi_2 = q_2 \xrightarrow{u_2 | v_2} q_2 \end{pmatrix}$$

$$\mathrm{init}(q_1') \wedge \mathrm{final}(q_2') \wedge u_1 = u \wedge u = u_2 \wedge v_1 \oplus v \neq v \oplus v_2$$



$v_1 v \neq v v_2$

Way to obtain decidability

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

Way to obtain decidability

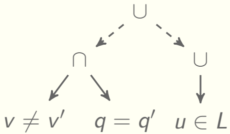
A $\models$ $(\exists \pi_1 = p_1 \xrightarrow{u_1|v_1} q_1), \ldots, (\exists \pi_n = p_n \xrightarrow{u_n|v_n} q_n), \mathcal{C}$

Acceptor of paths

n paths of A

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

constraint $\mathcal{C}$

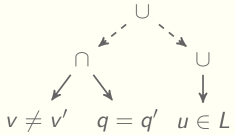


Way to obtain decidability

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 $\mathcal{C}$ $\neq \varnothing$

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

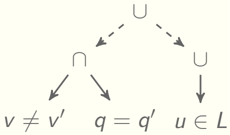


Way to obtain decidability

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

Acceptor of paths $\boxed{n \text{ paths of A} \quad \cap \quad \text{constraint } \mathcal{C}} \neq \varnothing$

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



Sufficient conditions for decidability

- generalise NFA
- decide emptiness
- recognise each predicate (and negation)
- closed under ∩ and ∪

Instances

Logic for NFA: $\mathrm{PL}_{\textsf{nfa}}$

$\mathrm{PL}_{\textsf{nfa}}$ defined as $\mathrm{PL}[\varnothing]$ over the trivial monoid

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

Input $u \sqsubseteq u' \mid u \in L \mid |u| \leq |u'|$

Path $\pi = \pi' \mid q = q' \mid \mathsf{init}(q) \mid \mathsf{final}(q)$

- L regular language represented as an NFA

Complexity

- general: PSpace-C
- fixed formula: NLogSpace-C

Logics for Sum-Automata: $\mathrm{PL}_{\mathsf{sum}}$, $\mathrm{PL}^{\neq}_{\mathsf{sum}}$

$\mathrm{PL}_{\mathsf{sum}}$ defined as $\mathrm{PL}[\leq, \in S]$ over $(\mathbb{Z}, +, 0)$

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

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

Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
State	$\pi = \pi' \mid q = q' \mid \mathsf{init}(q) \mid \mathsf{final}(q)$
Output	$t \leq t' \mid t \in S$

- L regular language represented as an NFA
- S semi-linear set represented as an $\exists\mathsf{FO}[\leq, +, 0, 1]$ formula
- $t, t' \in \mathit{Terms}(\{v_1, \ldots, v_n\}, +, 0)$

We also consider $\mathrm{PL}^{\neq}_{\mathsf{sum}}$ define as $\mathrm{PL}^{+}[\neq]$ over $(\mathbb{Z}, +, 0)$

Complexity

- general: PSPACE-C
- fixed formula: NP-C \ PTIME

Logic for Transducers : $\mathrm{PL}_{\textbf{trans}}$

$\mathrm{PL}_{\textbf{trans}}$ defined as $\mathrm{PL}^{+}[\not\sqsubseteq, <_{len}, \leq_{len}, \in N, \notin N]$ over $(\Gamma^*, \cdot, \varepsilon)$

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

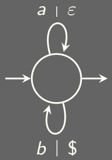
Input	$u \sqsubseteq u' \mid u \in L \mid \lvert u \rvert \leq \lvert u' \rvert$
Path	$\pi = \pi' \mid q = q' \mid \mathrm{init}(q) \mid \mathrm{final}(q)$
Output	$t \not\sqsubseteq t' \mid \lvert t \rvert <_{len} \lvert t' \rvert \mid \lvert t \leq_{len} \lvert t' \rvert \mid t \in N \mid t \notin N$

- L, N regular language represented as an NFA
- $t, t' \in \mathit{Terms}(\{v_1, \ldots, v_n\}, \cdot, \varepsilon)$

Complexity

- general: PSPACE-C
- fixed formula: NLOGSPACE-C

Acceptor of paths



Non-regular language

Parikh Automata

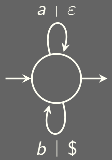
Syntax. P is a tuple (A, S) where

- A automaton with outputs in $(\mathbb{N}^d, +_d, 0_d)$
- $S \subseteq \mathbb{N}^d$ semilinear set

Alternative Semantics.

$$[\![P]\!] = \{(u, v) \mid (u, v) \in [\![A]\!] \wedge v \in S\}$$

Acceptor of paths



Non-regular language

Parikh Automata

Syntax. P is a tuple (A, S) where

- A automaton with outputs in $(\mathbb{N}^d, +_d, 0_d)$
- $S \subseteq \mathbb{N}^d$ semilinear set

Alternative Semantics.

$$[\![P]\!] = \{(u, v) \mid (u, v) \in [\![A]\!] \wedge v \in S\}$$

Theorem $[\![P]\!] = \varnothing$ [Figueira & Libkin 2015]

The emptiness problem of a Parikh Automata is NP-C *and* NLogSpace-C *if the dimension is fixed and weights are in* $\{0, 1\}$*.*

Conclusion

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

Future works

- Universal quantifications
- Better algorithmic complexities
- Extensions (trees, infinite words)

Thanks

Conclusion

Logic \Setting	General	Fixed Formula
PL_{nfa}	PSpace-C	NLogSpace-C
PL_{trans}	PSpace-C	NLogSpace-C
PL_{sum}	PSpace-C	NP-C binary NLogSpace-C unary
$PL^{\neq}_{sum}$	PSpace-C	PTime \ NLogSpace-H

C. Choffrut.
Une caracterisation des fonctions sequentielles et des fonctions sous-sequentielles en tant que relations rationnelles.
Theor. Comput. Sci., 5(3), 1977.

L. Daviaud, I. Jecker, P.-A. Reynier, and D. Villevalois.
Degree of sequentiality of weighted automata.
In *FOSSACS*, 2017.

D. Figueira and L. Libkin.
Path logics for querying graphs: Combining expressiveness and efficiency.
In *LICS*, pages 329–340, 2015.

E. Filiot, R. Gentilini, and J.-F. Raskin.
Finite-valued weighted automata.
In *FSTTCS*, pages 133–145, 2014.

I. Jecker and E. Filiot.
Multi-sequential word relations.
IJFCS, 29(2), 2018.

J. Sakarovitch and R. de Souza.
On the decidability of bounded valuedness for transducers.
In *MFCS*, pages 588–600, 2008.

A. Weber and H. Seidl.
On the degree of ambiguity of finite automata.
Theor. Comput. Sci., 325–349, 1991.