

Petri automata for Kleene Allegories

LICS 2015

Paul Brunet & Damien Pous

Plume team – LIP, CNRS, ENS de Lyon, Inria, UCBL, Université de Lyon, UMR 5668

July 5, 2015

Motivation : Relation Algebra

Let O be any set, and R, S and T any binary relations over O .

$$\text{Rel} \models 1 \cup R^* \cdot S \subseteq (R \cup S)^*$$

$$\text{Rel} \models (R \cap S) \cdot T \subseteq (R \cdot T) \cap (S \cdot T)$$

$$\text{Rel} \models (R \cdot S) \cap T \subseteq R \cdot (S \cap R^\sim \cdot T)$$

$$\text{Rel} \models (1 \cap (R \cdot S))^* \subseteq R \cdot S \cdot R$$

Relation operators

identity relation	:	1
empty relation	:	0
composition	:	$R \cdot S$
union	:	$R \cup S$
intersection	:	$R \cap S$
converse	:	R^\sim
reflexive transitive closure	:	R^*

Motivation : Relation Algebra

Let O be any set, and R, S and T any binary relations over O .

$$\text{Rel} \models 1 \cup R^* \cdot S \subseteq (R \cup S)^*$$

$$\text{Rel} \models (R \cap S) \cdot T \subseteq (R \cdot T) \cap (S \cdot T)$$

$$\text{Rel} \models (R \cdot S) \cap T \subseteq R \cdot (S \cap R^\sim \cdot T)$$

$$\text{Rel} \not\models (1 \cap (R \cdot S))^* \subseteq R \cdot S \cdot R$$

Relation operators

identity relation	:	1
empty relation	:	0
composition	:	$R \cdot S$
union	:	$R \cup S$
intersection	:	$R \cap S$
converse	:	R^\sim
reflexive transitive closure	:	R^*

Motivation : Relation Algebra

Let O be any set, and R, S and T any binary relations over O .

$$\text{Rel} \models 1 \cup R^* \cdot S \subseteq (R \cup S)^*$$

$$\text{Rel} \models (R \cap S) \cdot T \subseteq (R \cdot T) \cap (S \cdot T)$$

$$\text{Rel} \models (R \cdot S) \cap T \subseteq R \cdot (S \cap R^\sim \cdot T)$$

$$\text{Rel} \not\models (1 \cap (R \cdot S))^* \subseteq R \cdot S \cdot R$$

Relation operators

identity relation	:	1
empty relation	:	0
composition	:	$R \cdot S$
union	:	$R \cup S$
intersection	:	$R \cap S$
converse	:	R^\sim
reflexive transitive closure	:	R^*

Simple and boring : could it be done automatically ?

Outline

- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions

Regular expressions

$$e, f \in \text{Reg}_X ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cup f \mid e^*$$

Regular expressions

$$e, f \in \text{Reg}_X ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cup f \mid e^*$$

Theorem

$$\text{Rel} \models e = f \Leftrightarrow L(e) = L(f)$$

where $L(e) \subseteq X^*$ is the *rational language* represented by e .

Regular expressions

$$e, f \in \text{Reg}_X ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cup f \mid e^*$$

Theorem

$$\text{Rel} \models e = f \Leftrightarrow L(e) = L(f)$$

where $L(e) \subseteq X^*$ is the *rational language* represented by e .

Corollary

Relational equivalence is decidable for regular expressions.

Kleene Allegories

$$e, f \in \text{Reg}_X^{\sim \cap} ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cap f \mid e \cup f \mid e^* \mid e^{\sim}$$

Kleene Allegories

$$e, f \in \text{Reg}_X^{\sim \cap} ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cap f \mid e \cup f \mid e^* \mid e^{\sim}$$

$$\text{Rel} \models e = f \Leftrightarrow L(e) = L(f)$$

Kleene Allegories

$$e, f \in \text{Reg}_X^{\sim} ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cap f \mid e \cup f \mid e^* \mid e^{\sim}$$

$$\text{Rel} \models e = f \iff L(e) = L(f)$$

Counterexample

$$L(a \cap b) = \emptyset = L(0) \quad \Bigg| \quad L(a) = \{a\} = L(a^{\sim})$$

$$\text{Rel} \not\models a \cap b = 0$$

$$\text{Rel} \not\models a = a^{\sim}$$

A different approach is needed.

Outline

- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions

Outline

- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions

Graphs/Ground terms

$$u, v \in W_X ::= 0 \mid 1 \mid x \in X \mid u \cdot v \mid u \cap v \mid u \cup v \mid u^* \mid u^\sim$$

Graphs/Ground terms

$$u, v \in W_X ::= 0 \mid 1 \mid x \in X \mid u \cdot v \mid u \cap v \mid u \cup v \mid u^* \mid u^\sim$$

$$\mathcal{G}(1) := \rightarrow \circ \rightarrow$$

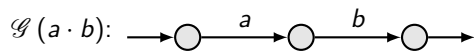
$$\mathcal{G}(x) := \rightarrow \circ \xrightarrow{x} \circ \rightarrow$$

$$\mathcal{G}(u^\sim) := \leftarrow \circ \xrightarrow{G(u)} \circ \leftarrow$$

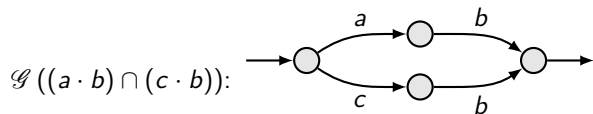
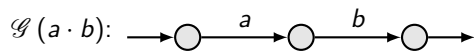
$$\mathcal{G}(u \cdot v) := \rightarrow \circ \xrightarrow{G(u)} \circ \xrightarrow{G(v)} \circ \rightarrow$$

$$\mathcal{G}(u \cap v) := \rightarrow \circ \begin{array}{l} \xrightarrow{G(u)} \\ \xrightarrow{G(v)} \end{array} \circ \rightarrow$$

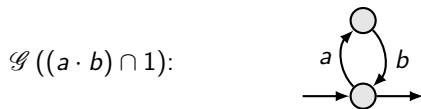
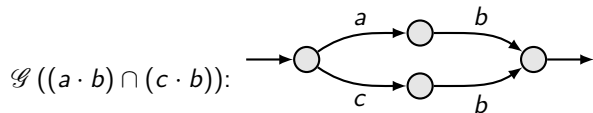
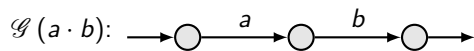
Examples



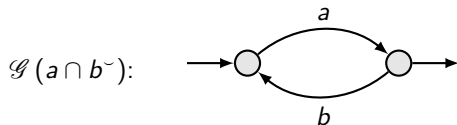
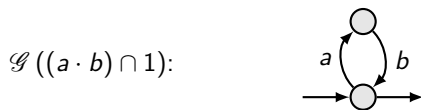
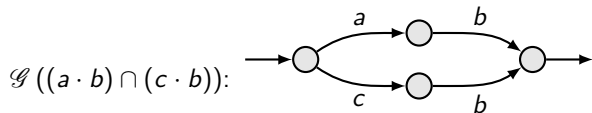
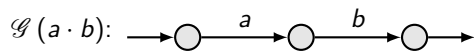
Examples



Examples



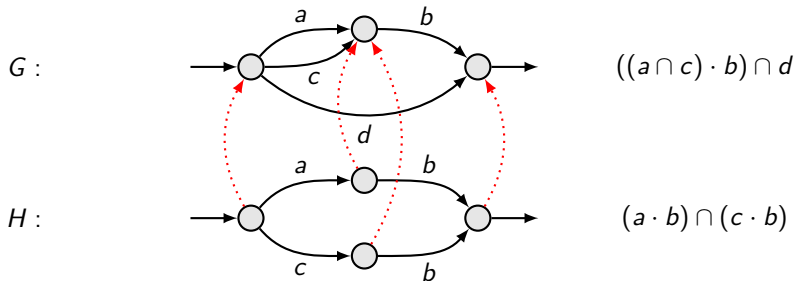
Examples



Preorder

Preorder on graphs

$G \triangleleft H$ if there exists a graph morphism from H to G .



Characterization theorem

$$u, v \in W_X ::= 1 \mid x \in X \mid u \cdot v \mid u \cap v \mid u^\sim$$

Theorem

$$\text{Rel} \models u \subseteq v \Leftrightarrow \mathcal{G}(u) \blacktriangleleft \mathcal{G}(v)$$

- P. J. Freyd and A. Scedrov. *Categories, Allegories*. NH, 1990
- H. Andréka and D. Bredikhin.

The equational theory of union-free algebras of relations.
Alg. Univ., 33(4):516–532, 1995

Graphs/Ground terms languages

$$e, f \in \text{Reg}_X^{\sim} ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cap f \mid e \cup f \mid e^* \mid e^{\sim}$$

$$\mathcal{G}(1) := \{ \rightarrow \circ \rightarrow \} \quad \mathcal{G}(x) := \left\{ \rightarrow \circ \xrightarrow{x} \circ \rightarrow \right\}$$

$$\mathcal{G}(e^{\sim}) := \{ G^{-1} \mid G \in \mathcal{G}(e) \}$$

$$\mathcal{G}(e \cdot f) := \{ G ; G' \mid G \in \mathcal{G}(e) \text{ and } G' \in \mathcal{G}(f) \}$$

$$\mathcal{G}(e \cap f) := \{ G \parallel G' \mid G \in \mathcal{G}(e) \text{ and } G' \in \mathcal{G}(f) \}$$

$$\mathcal{G}(0) := \emptyset \quad \mathcal{G}(e \cup f) := \mathcal{G}(e) \cup \mathcal{G}(f)$$

$$\mathcal{G}(e^*) := \bigcup_{n \in \mathbb{N}} \{ G_1 ; \dots ; G_n \mid \forall i, G_i \in \mathcal{G}(e) \} .$$

Characterization theorem

$\blacktriangleleft S$ is the downwards closure of S with respect to \blacktriangleleft .

Theorem

$e, f \in \text{Reg}_X^{\cup}$,

$$\text{Rel} \models e \subseteq f \Leftrightarrow \blacktriangleleft \mathcal{G}(e) \subseteq \blacktriangleleft \mathcal{G}(f)$$

Follows easily from:

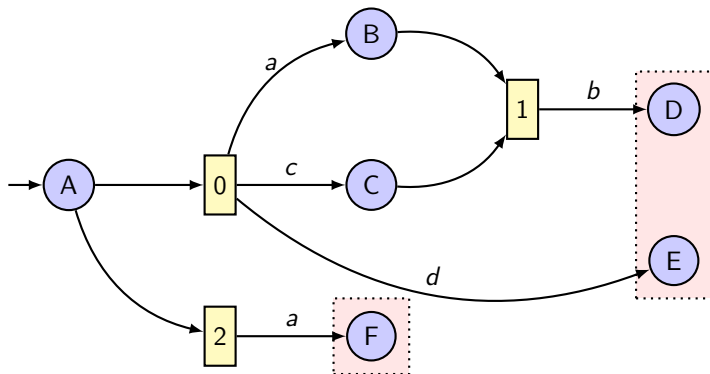
H. Andr eka, S. Mikul as, and I. N emeti. [The equational theory of Kleene lattices.](#)
TCS, 412(52):7099–7108, 2011

Outline

- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions

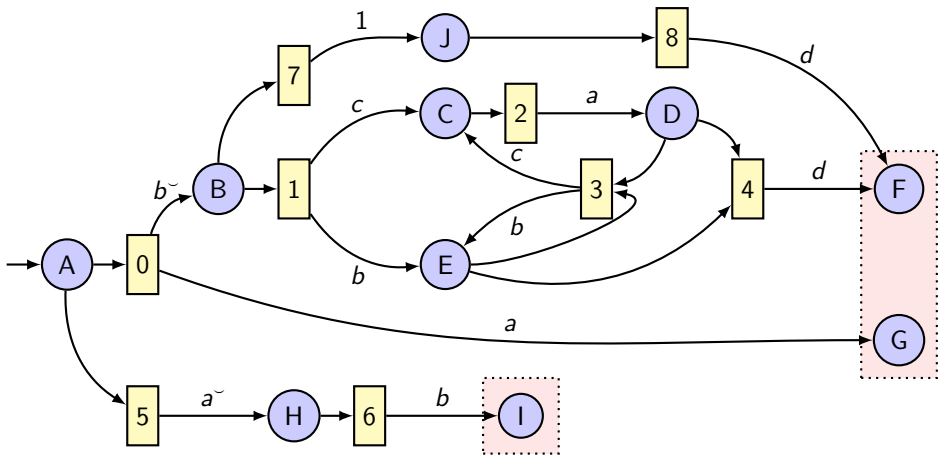
Example

$$(((a \cap c) \cdot b) \cap d) \cup a$$

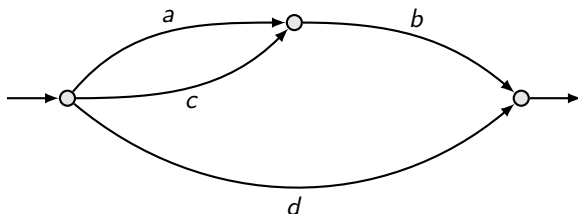
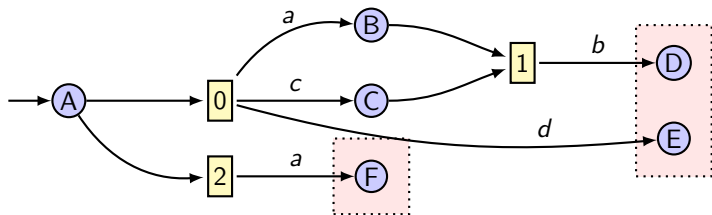


Example

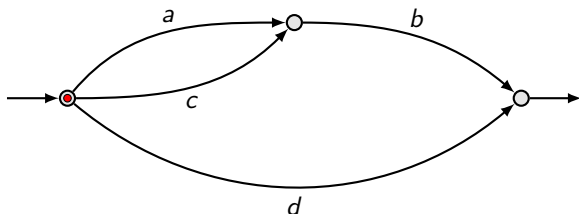
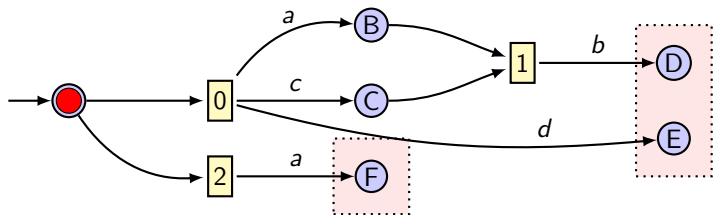
$$(b^{\sim} \cdot (a \cdot c \cap b)^{\star} \cdot d) \cap a \cup a^{\sim} \cdot b$$



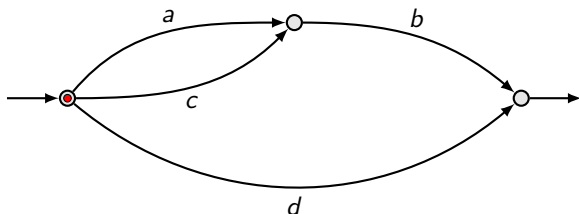
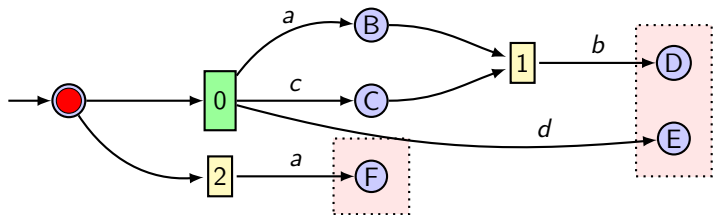
Reading a graph in an automaton



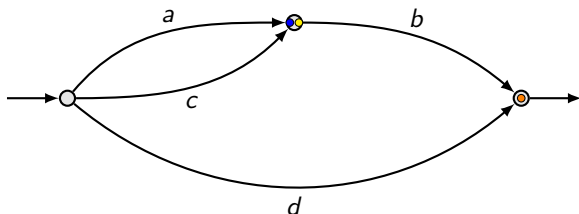
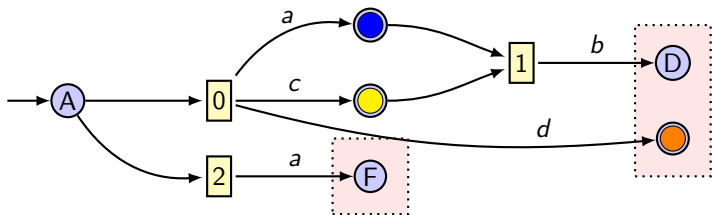
Reading a graph in an automaton



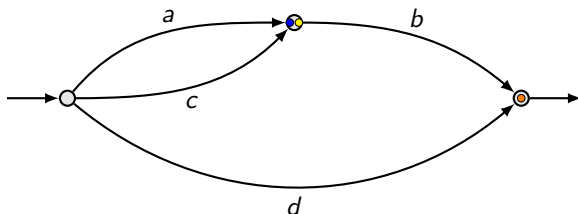
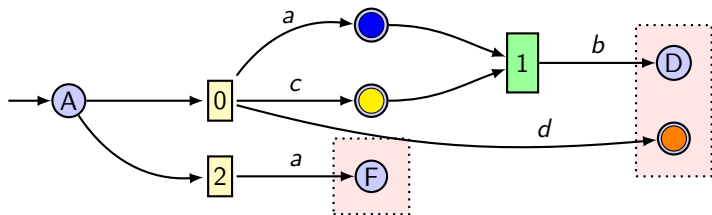
Reading a graph in an automaton



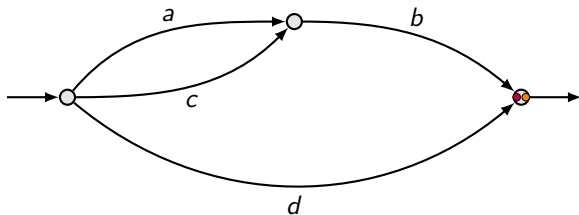
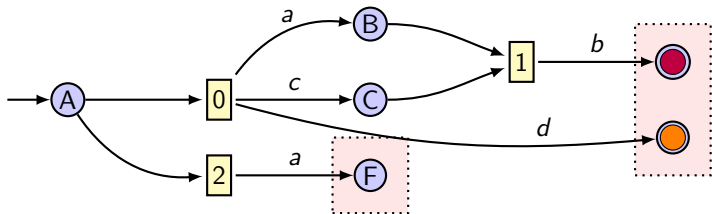
Reading a graph in an automaton



Reading a graph in an automaton

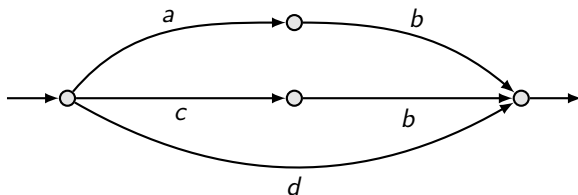
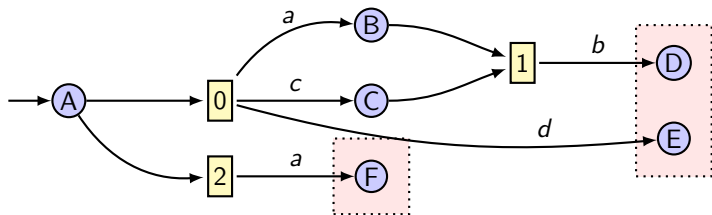


Reading a graph in an automaton

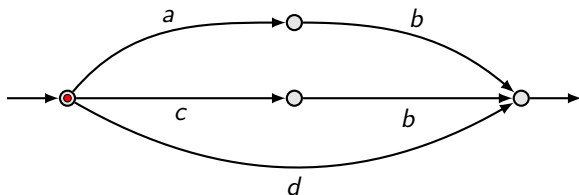
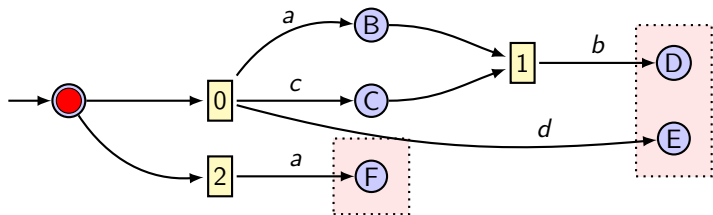


Success!

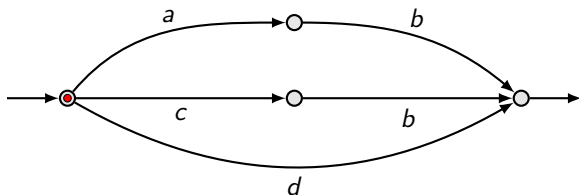
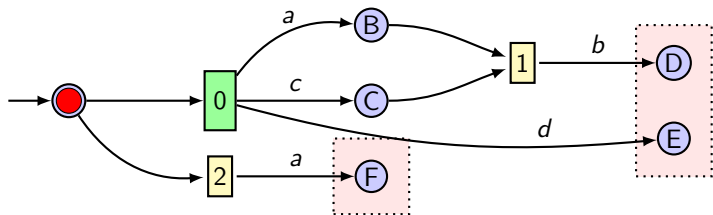
Reading a graph in an automaton



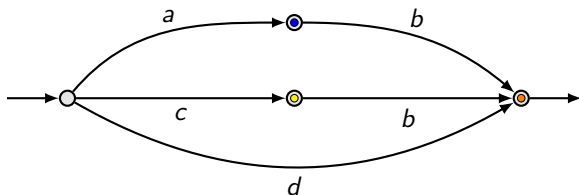
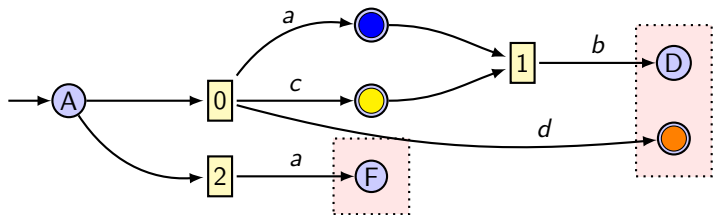
Reading a graph in an automaton



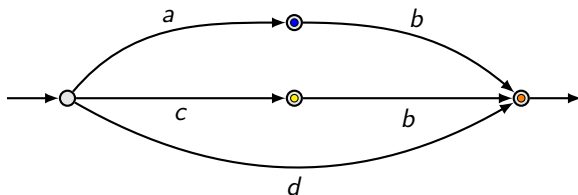
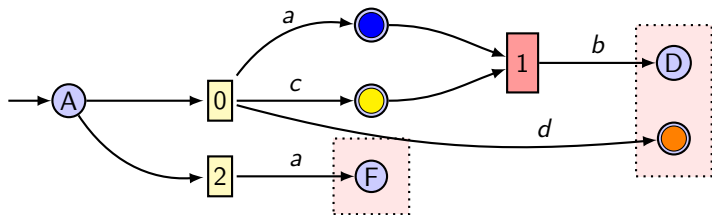
Reading a graph in an automaton



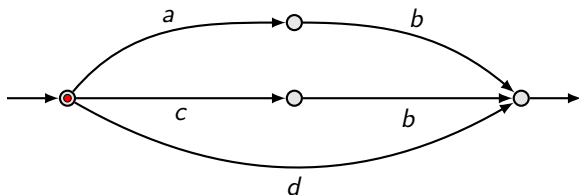
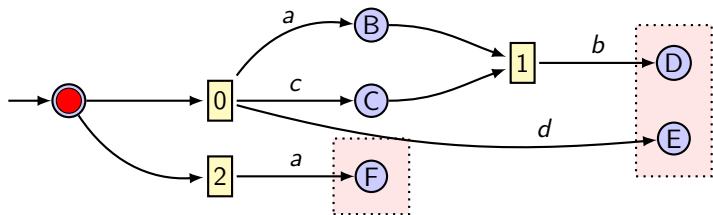
Reading a graph in an automaton



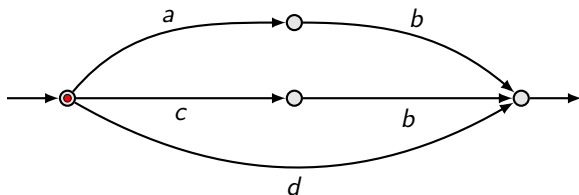
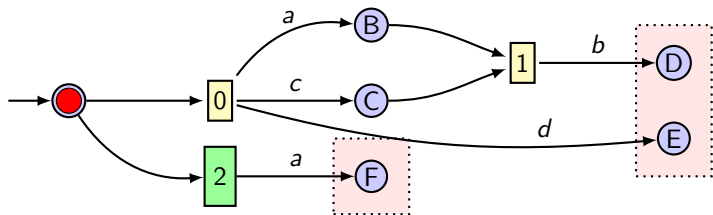
Reading a graph in an automaton

**Failure!**

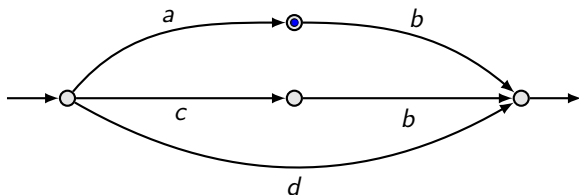
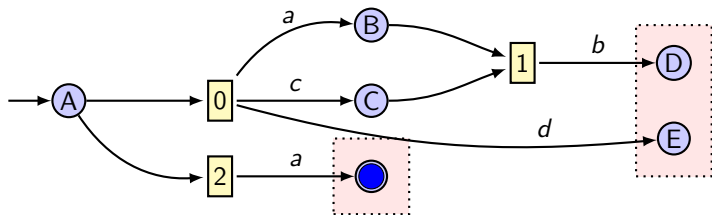
Reading a graph in an automaton



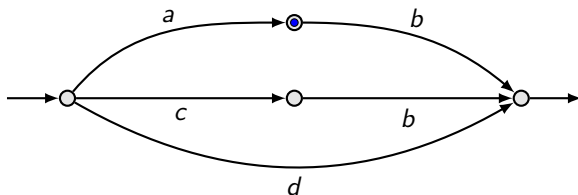
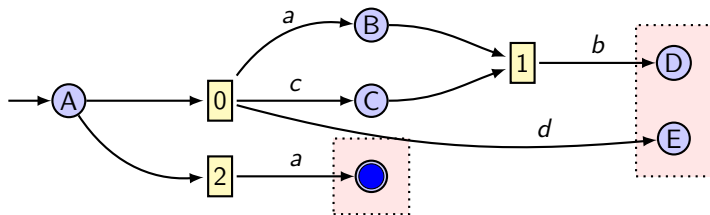
Reading a graph in an automaton



Reading a graph in an automaton



Reading a graph in an automaton

**Failure!**

Language recognised by an automaton

Correctness

For any $e \in \text{Reg}_X^{\cup \cap}$,

e

Language recognised by an automaton

Correctness

For any $e \in \text{Reg}_X^{\cup}$,

$$\mathcal{A}(e)$$

Language recognised by an automaton

Correctness

For any $e \in \text{Reg}_X^{\cup}$,

$$\mathcal{L}(\mathcal{A}(e))$$

Language recognised by an automaton

Correctness

For any $e \in \text{Reg}_X^{\cup}$,

$$\mathcal{L}(\mathcal{A}(e)) = \mathcal{G}(e).$$

Language recognised by an automaton

Correctness

For any $e \in \text{Reg}_X^{\checkmark}$,

$$\mathcal{L}(\mathcal{A}(e)) = \checkmark \mathcal{G}(e).$$

So far:

$e, f \in \text{Reg}_X^{\checkmark}$

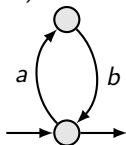
$$\text{Rel} \models e \subseteq f \Leftrightarrow \checkmark \mathcal{G}(e) \subseteq \checkmark \mathcal{G}(f) \Leftrightarrow \mathcal{L}(\mathcal{A}(e)) \subseteq \mathcal{L}(\mathcal{A}(f)).$$

Outline

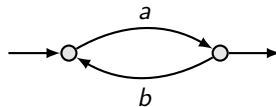
- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions

Restriction: identity-free lattice terms

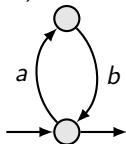
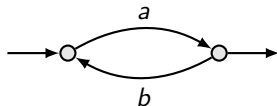
$\mathcal{G}((a \cdot b) \cap 1)$:



$\mathcal{G}(a \cap b^{\sim})$:



Restriction: identity-free lattice terms

 $\mathcal{G}((a \cdot b) \cap 1)$: $\mathcal{G}(a \cap b^-)$:

Identity-free Kleene Lattice

$$u, v \in W_X^- ::= 1 \mid x \in X \mid u \cdot v \mid u \cap v \mid u^-$$

$$e, f \in \text{Reg}_X^{\cap -} ::= 0 \mid 1 \mid x \in X \mid e \cdot f \mid e \cap f \mid e \cup f \mid e^+ \mid e^-$$

Decision procedure

$$e, f \in \text{Reg}_X^{\cap -}$$

$$\text{Rel} \models e \subseteq f \Leftrightarrow \mathcal{G}(e) \subseteq \mathcal{G}(f) \Leftrightarrow \mathcal{L}(\mathcal{A}(e)) \subseteq \mathcal{L}(\mathcal{A}(f)).$$

Problem:

How to compare two Petri automata?

Decision procedure

$$e, f \in \text{Reg}_X^{\cap -}$$

$$\text{Rel} \models e \subseteq f \Leftrightarrow \mathcal{G}(e) \subseteq \mathcal{G}(f) \Leftrightarrow \mathcal{L}(\mathcal{A}(e)) \subseteq \mathcal{L}(\mathcal{A}(f)).$$

Problem:

How to compare two Petri automata?

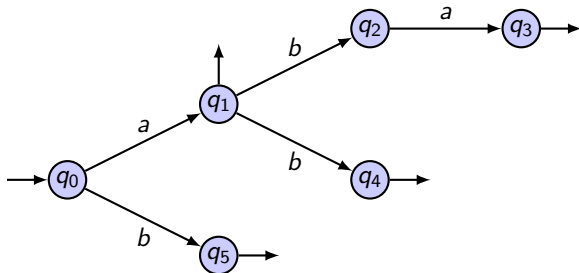
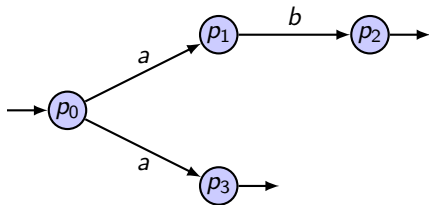
$\mathcal{L}(\mathcal{A}_1) \subseteq \mathcal{L}(\mathcal{A}_2)$ if and only if there is a **simulation** relation

$$\leq \subseteq \mathcal{P}(P_1) \times \mathcal{P}(P_2 \twoheadrightarrow P_1)$$

between the configurations of \mathcal{A}_1 and the partial maps from the places of \mathcal{A}_2 to the places of \mathcal{A}_1 .

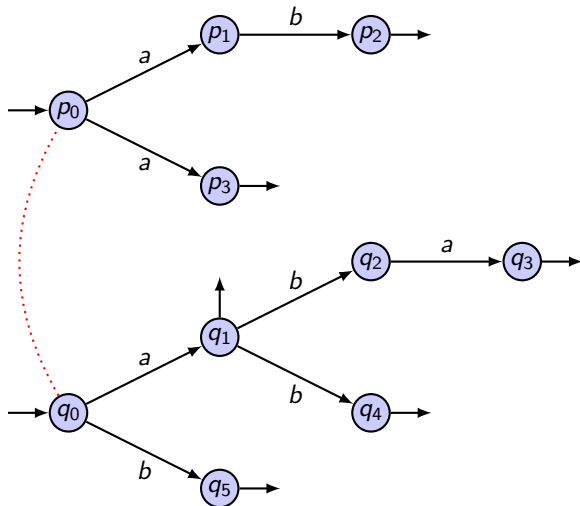
Simulations - Non-deterministic Finite Automata

$$\leq \subseteq Q_1 \times \mathcal{P}(Q_2)$$



Simulations - Non-deterministic Finite Automata

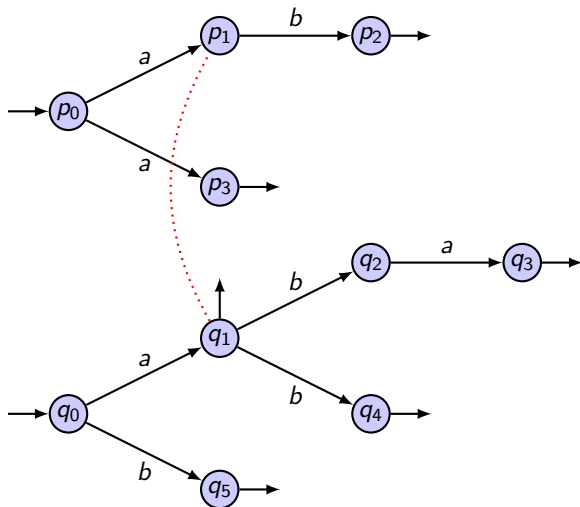
$$\leq \subseteq Q_1 \times \mathcal{P}(Q_2)$$



$$p_0 \leq \{q_0\}$$

Simulations - Non-deterministic Finite Automata

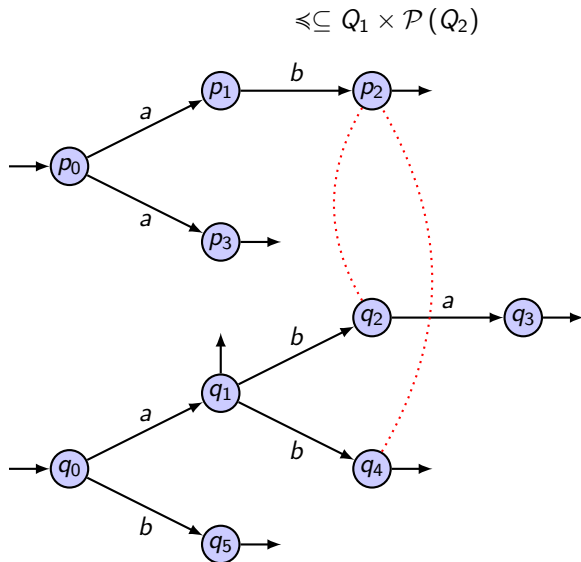
$$\leq \subseteq Q_1 \times \mathcal{P}(Q_2)$$



$$p_0 \leq \{ q_0 \}$$

$$p_1 \leq \{ q_1 \}$$

Simulations - Non-deterministic Finite Automata



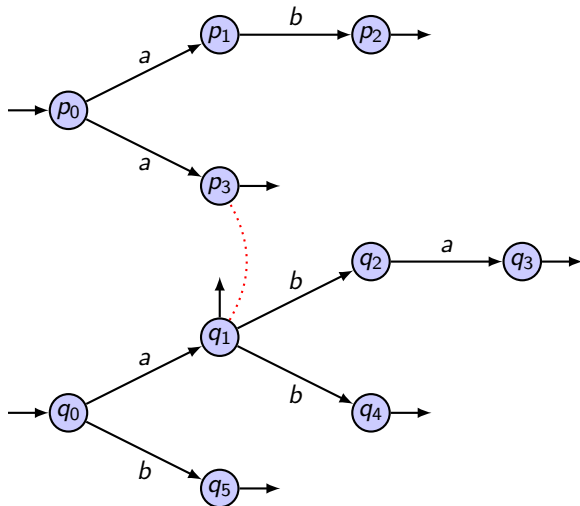
$$p_0 \leq \{ q_0 \}$$

$$p_1 \leq \{ q_1 \}$$

$$p_2 \leq \{ q_2, q_4 \}$$

Simulations - Non-deterministic Finite Automata

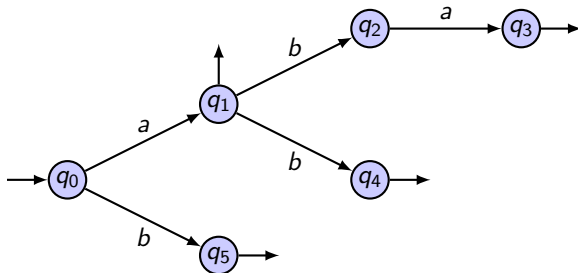
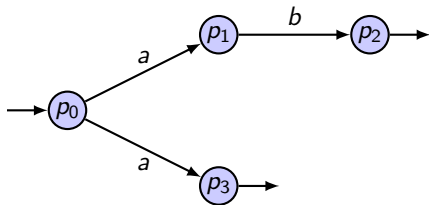
$$\leq \subseteq Q_1 \times \mathcal{P}(Q_2)$$



$$\begin{aligned} p_0 &\leq \{ q_0 \} \\ p_1 &\leq \{ q_1 \} \\ p_2 &\leq \{ q_2, q_4 \} \\ p_3 &\leq \{ q_1 \} \end{aligned}$$

Simulations - Non-deterministic Finite Automata

$$\leq \subseteq Q_1 \times \mathcal{P}(Q_2)$$



$$p_0 \leq \{ q_0 \}$$

$$p_1 \leq \{ q_1 \}$$

$$p_2 \leq \{ q_2, q_4 \}$$

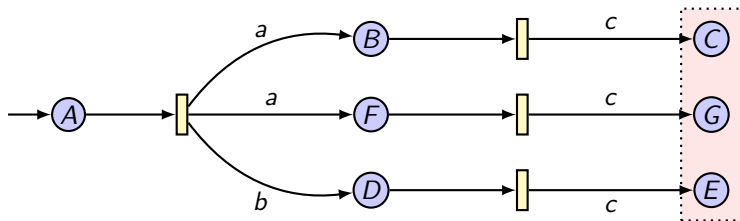
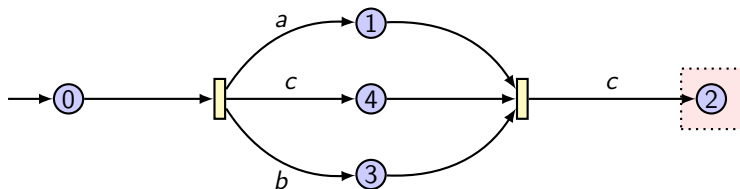
$$p_3 \leq \{ q_1 \}$$

+ condition
for accepting
states

Simulation - Petri Automata

Simulation relation

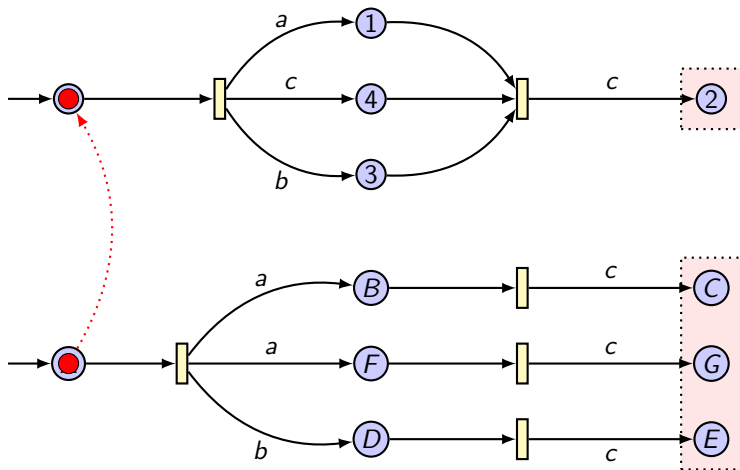
$$\leq \subseteq \mathcal{P}(P_1) \times \mathcal{P}(P_2 \rightarrow P_1)$$



Simulation - Petri Automata

Simulation relation

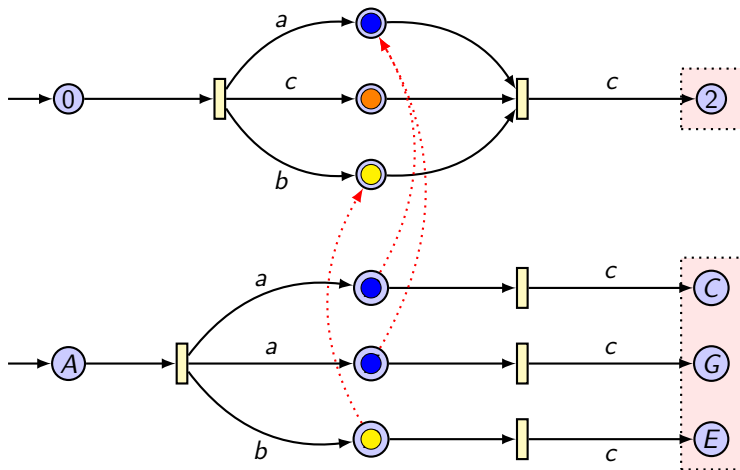
$$\leq \subseteq \mathcal{P}(P_1) \times \mathcal{P}(P_2 \rightarrow P_1)$$



Simulation - Petri Automata

Simulation relation

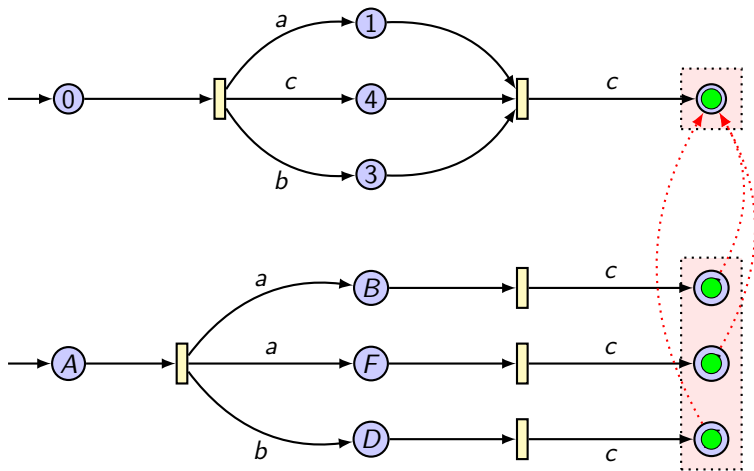
$$\leq \subseteq \mathcal{P}(P_1) \times \mathcal{P}(P_2 \twoheadrightarrow P_1)$$



Simulation - Petri Automata

Simulation relation

$$\leq \subseteq \mathcal{P}(P_1) \times \mathcal{P}(P_2 \rightarrow P_1)$$



Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.

Ongoing/Future work

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.

Ongoing/Future work

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.
- **Decidability** of simple automata equivalence, thus of relational equivalence for Identity-free Kleene Lattices.

Ongoing/Future work

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.
- **Decidability** of simple automata equivalence, thus of relational equivalence for Identity-free Kleene Lattices.
- Simple Petri automata equivalence is **EXPSPACE-complete**.

This decision procedure was implemented in OCAML, and is available as an online application.

Ongoing/Future work

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.
- **Decidability** of simple automata equivalence, thus of relational equivalence for Identity-free Kleene Lattices.
- Simple Petri automata equivalence is **EXPSPACE-complete**.

This decision procedure was implemented in OCAML, and is available as an online application.

Ongoing/Future work

- Converting back Petri automata into expressions.

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.
- **Decidability** of simple automata equivalence, thus of relational equivalence for Identity-free Kleene Lattices.
- Simple Petri automata equivalence is **EXPSPACE-complete**.

This decision procedure was implemented in OCAML, and is available as an online application.

Ongoing/Future work

- Converting back Petri automata into expressions.
- (Un)decidability with 1 and/or $_ \sim$.

Conclusion and future work

Results

- **Reduction** of relational equivalence to equality of closed graph languages.
- Representation of closed graph languages through **Petri automata**.
- **Decidability** of simple automata equivalence, thus of relational equivalence for Identity-free Kleene Lattices.
- Simple Petri automata equivalence is **EXPSPACE-complete**.

This decision procedure was implemented in OCAML, and is available as an online application.

Ongoing/Future work

- Converting back Petri automata into expressions.
- (Un)decidability with 1 and/or $_ \sim$.
- Complete axiomatization.

That's it!

Thank you !

[http://perso.ens-lyon.fr/paul.brunet/rklm.](http://perso.ens-lyon.fr/paul.brunet/rklm)

Plan

- 1 Expressions
 - Kleene Algebra
 - Kleene Allegories
- 2 Graph languages
 - Ground terms
 - Kleene Allegories expressions
- 3 Petri Automata
 - Examples
 - Recognition by Petri automata
- 4 Comparing automata
- 5 Conclusions