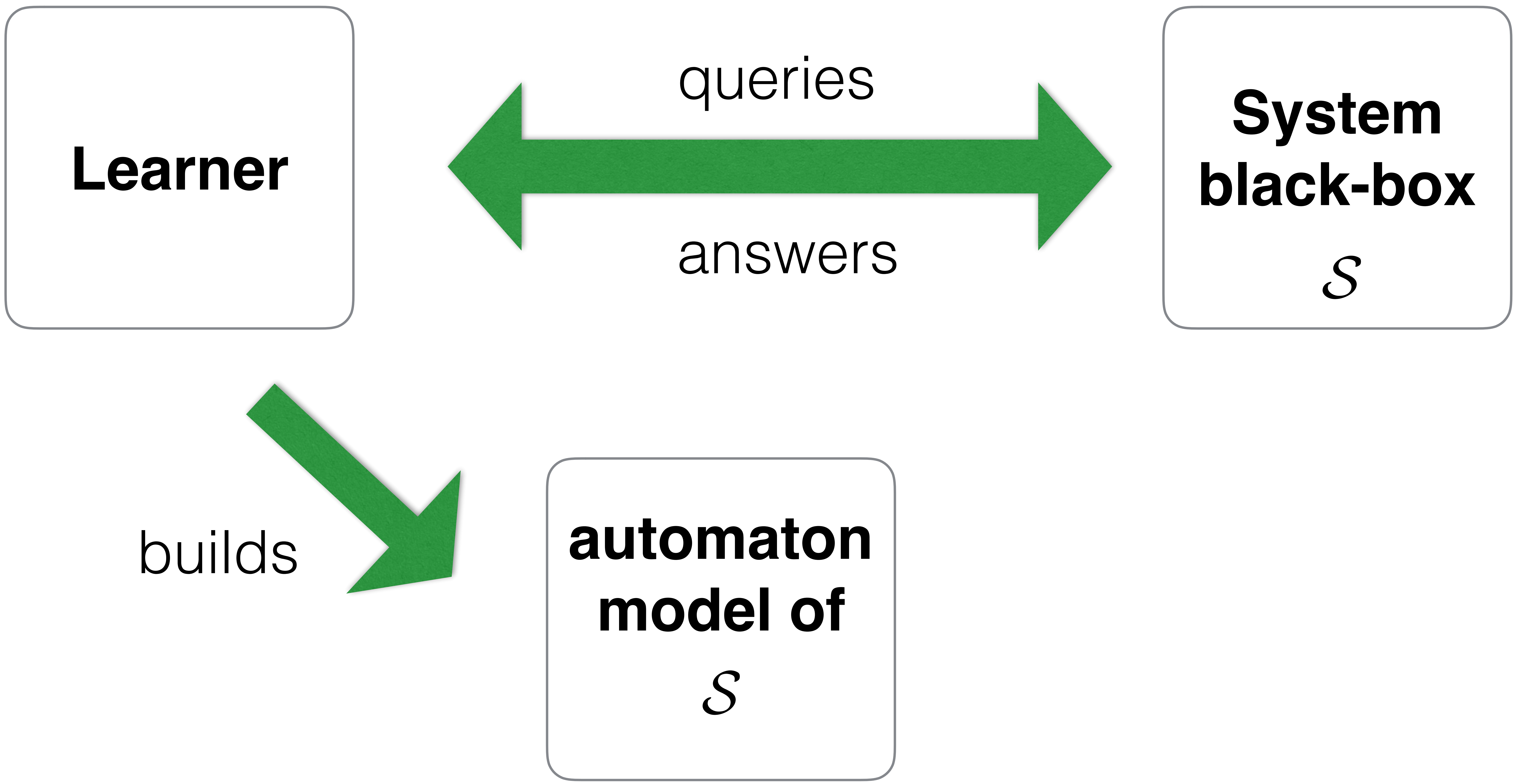
Automata learning

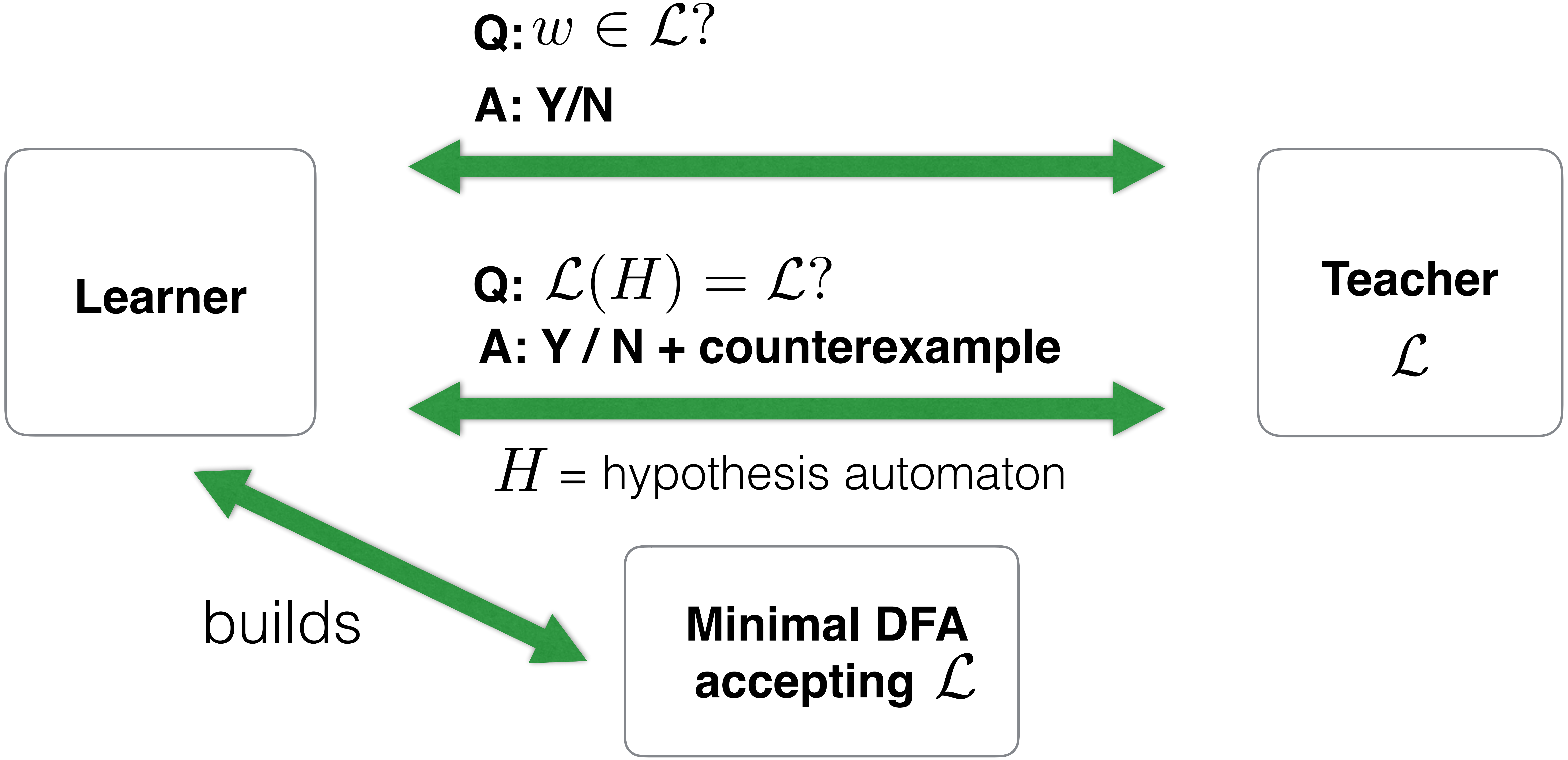


No formal specification available? **Learn it!**

L* algorithm (D.Angluin '87)

Finite alphabet of system's actions A

set of system behaviors is a **regular language** $\mathcal{L} \subseteq A^\star$



A zoo of automata

Weighted

Probabilistic

Alternating

Universal

Non-deterministic

Register

Category theory comes to the rescue!

Algorithms

Correctness proofs

involved and hard to check

Category Theory

Conceptual tools

Correctness proof(s)

Guidelines new algorithms

Unveil connections

No free lunch!

Automata

$$X \to 2 \times X^A$$

DFA

$$X \to \mathbb{R} \times (\mathbb{R}^X)^A$$

WFA

$$X \to FTX$$

Transition structure

Algebraic properties

$$X \to FTX$$

$$X \to 2 \times X^A$$

DFA

$$X \to \mathbb{R} \times (\mathbb{R}^X)^A$$

WFA

$$2^{A^*}$$

acceptance

$$\mathbb{R}^{A^*}$$

Vector space

Language equivalence

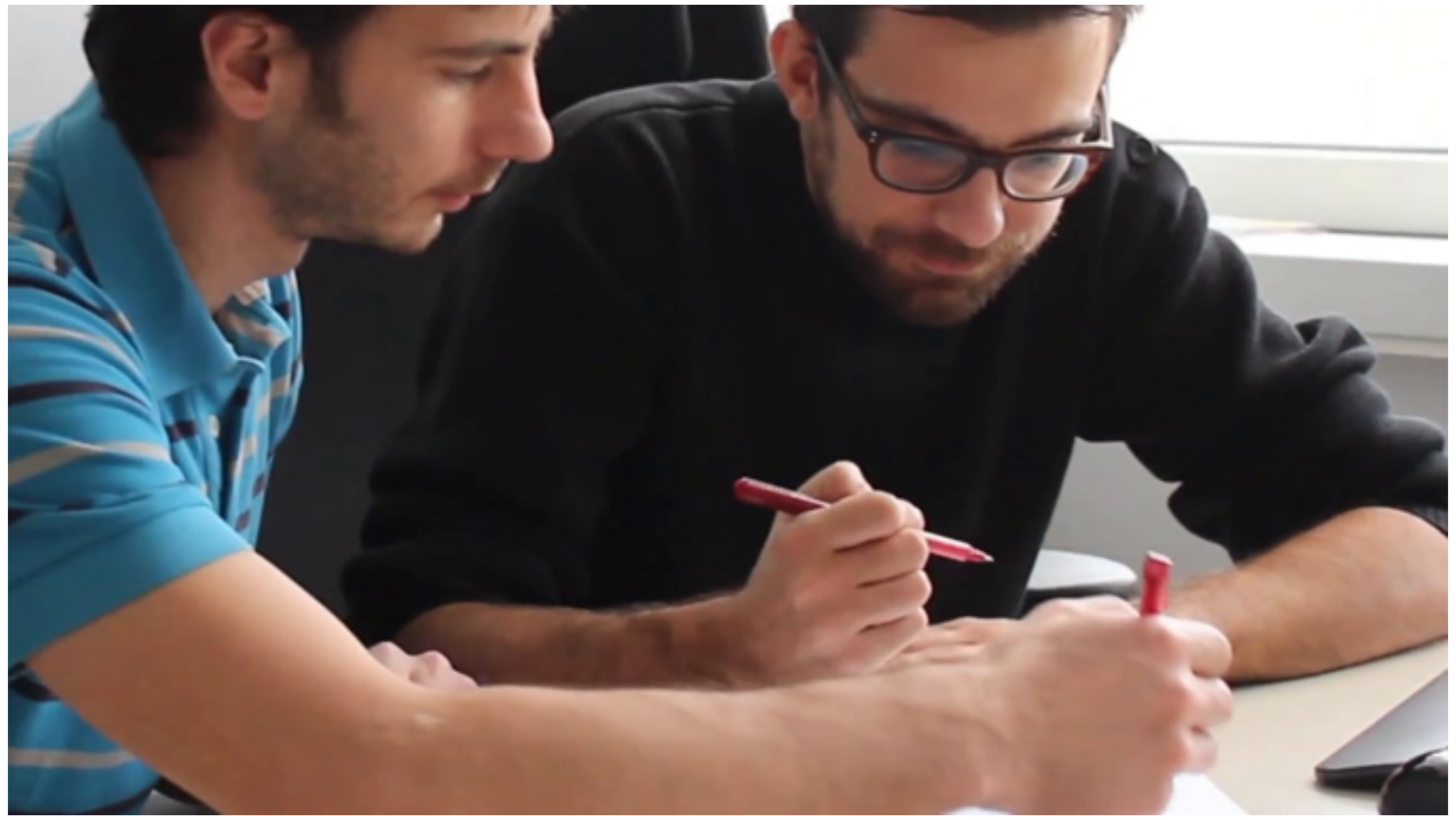
equivalence

Weighted language equivalence **or** bisimilarity

Proof methods for equivalence

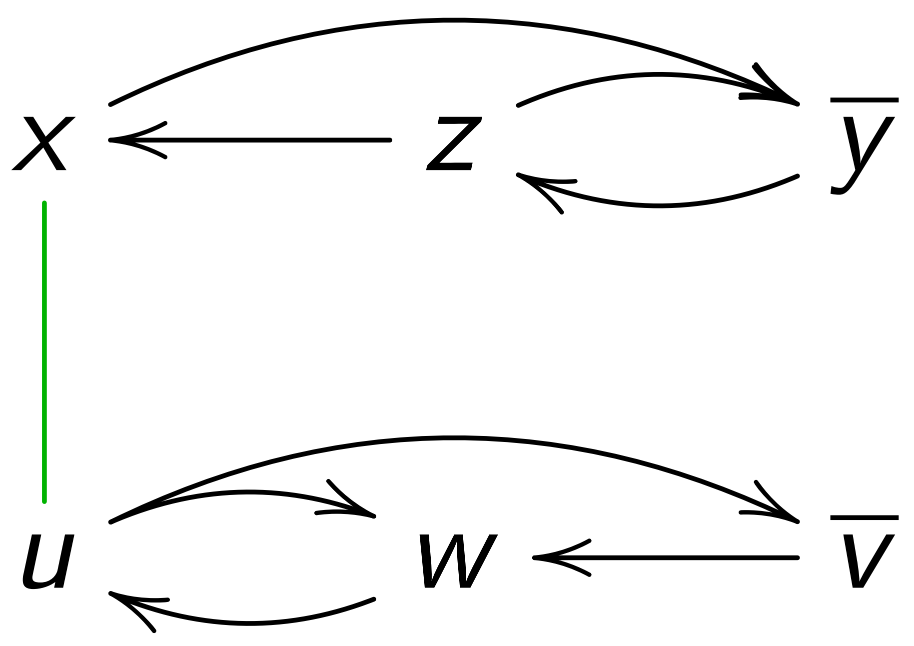
Up-to techniques

Algebraic structure → Better Proof Techniques

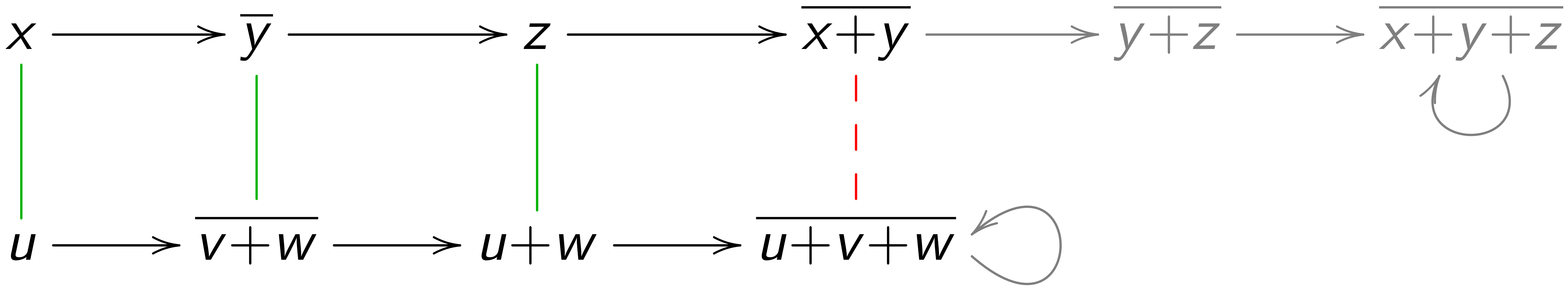


HKC algorithm - Bonchi and Pous 2014

Example

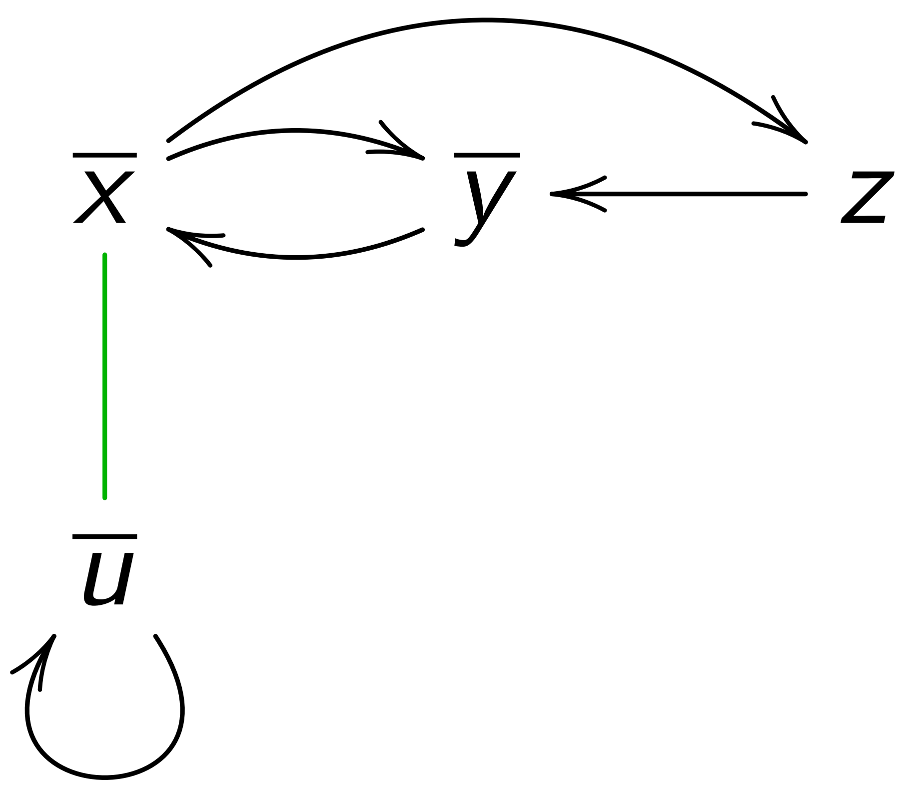


$$
\begin{array}{rl}
 & (x,\ u) \\
+ & (y,\ v+w) \\
\hline
= & (x+y,\ u+v+w)
\end{array}
$$

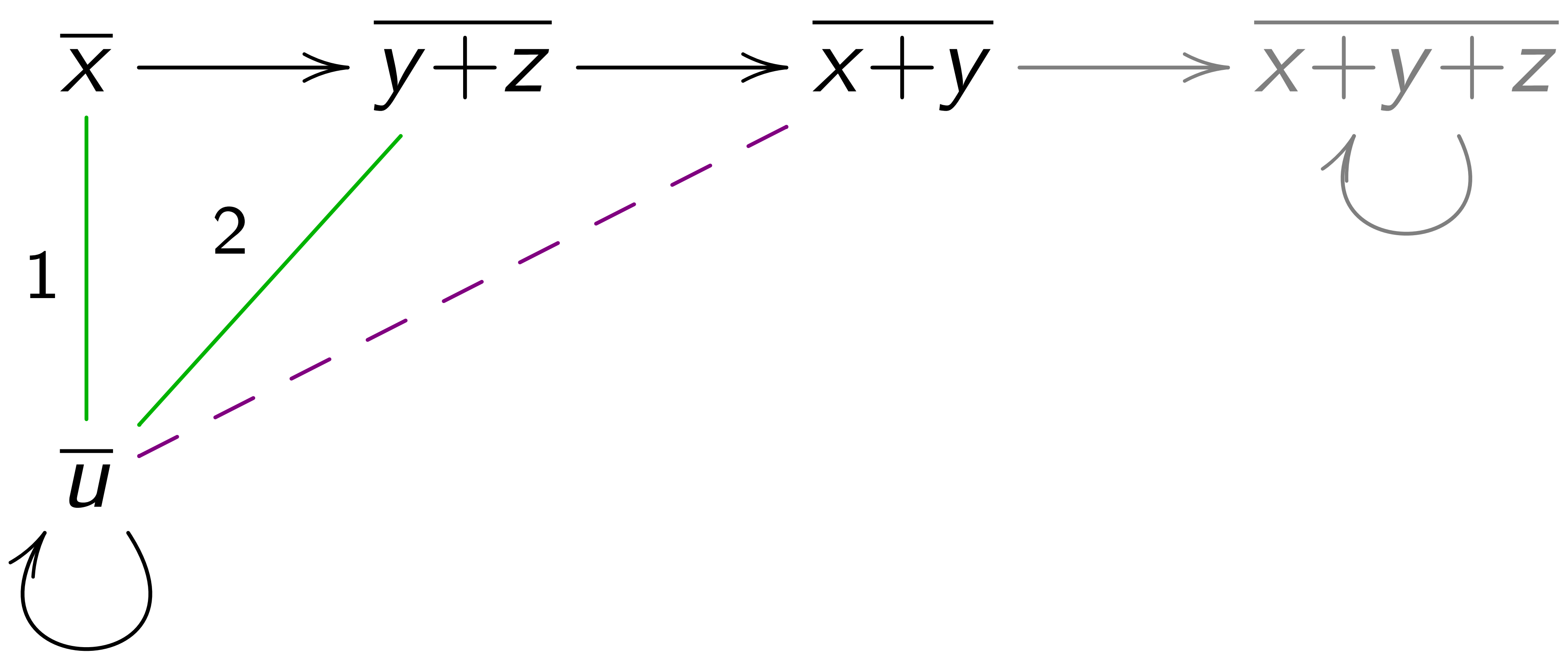


using bisimulations up to union

Another example



$$
\begin{aligned}
x+y &= u+y && (1)\\
&= y+z+y && (2)\\
&= y+z &&\\
&= u && (2)
\end{aligned}
$$



Bisimulations up-to **congruence**
HKC algorithm of Bonchi&Pous

More examples

Up-To Techniques for Weighted Systems. (TACAS ’17)

Filippo Bonchi, Barbara König, Sebastian Küpper

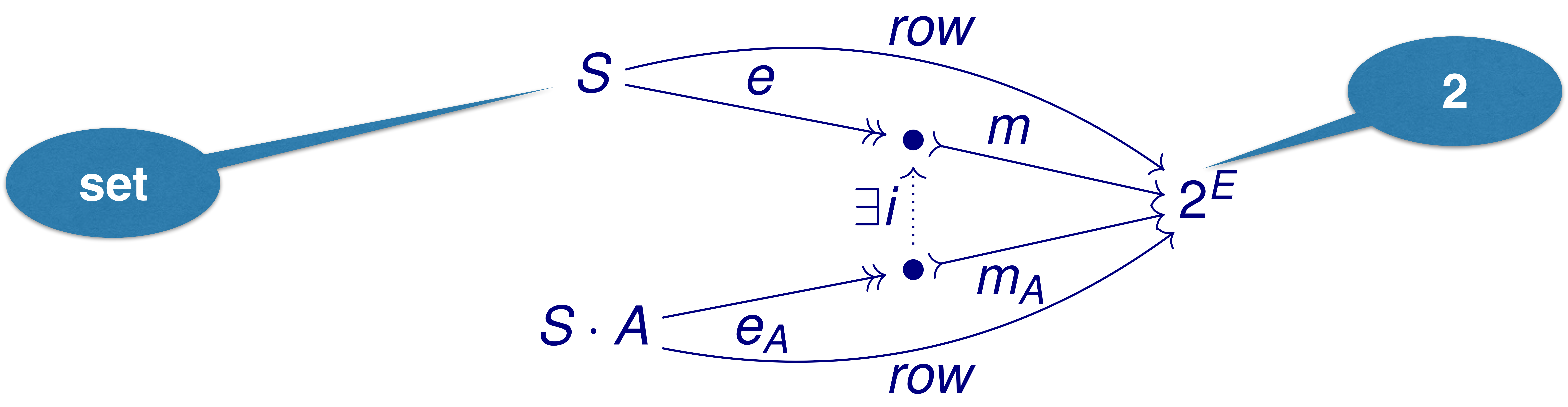
The Power of Convex Algebras (CONCUR’ 17)

Filippo Bonchi, Alexandra Silva, Ana Sokolova

Coinduction up-to in a fibrational setting (CSL-LICS 2014)

Filippo Bonchi, Daniela Petrisan, Damien Pous, Jurriaan Rot

Category Theory in learning



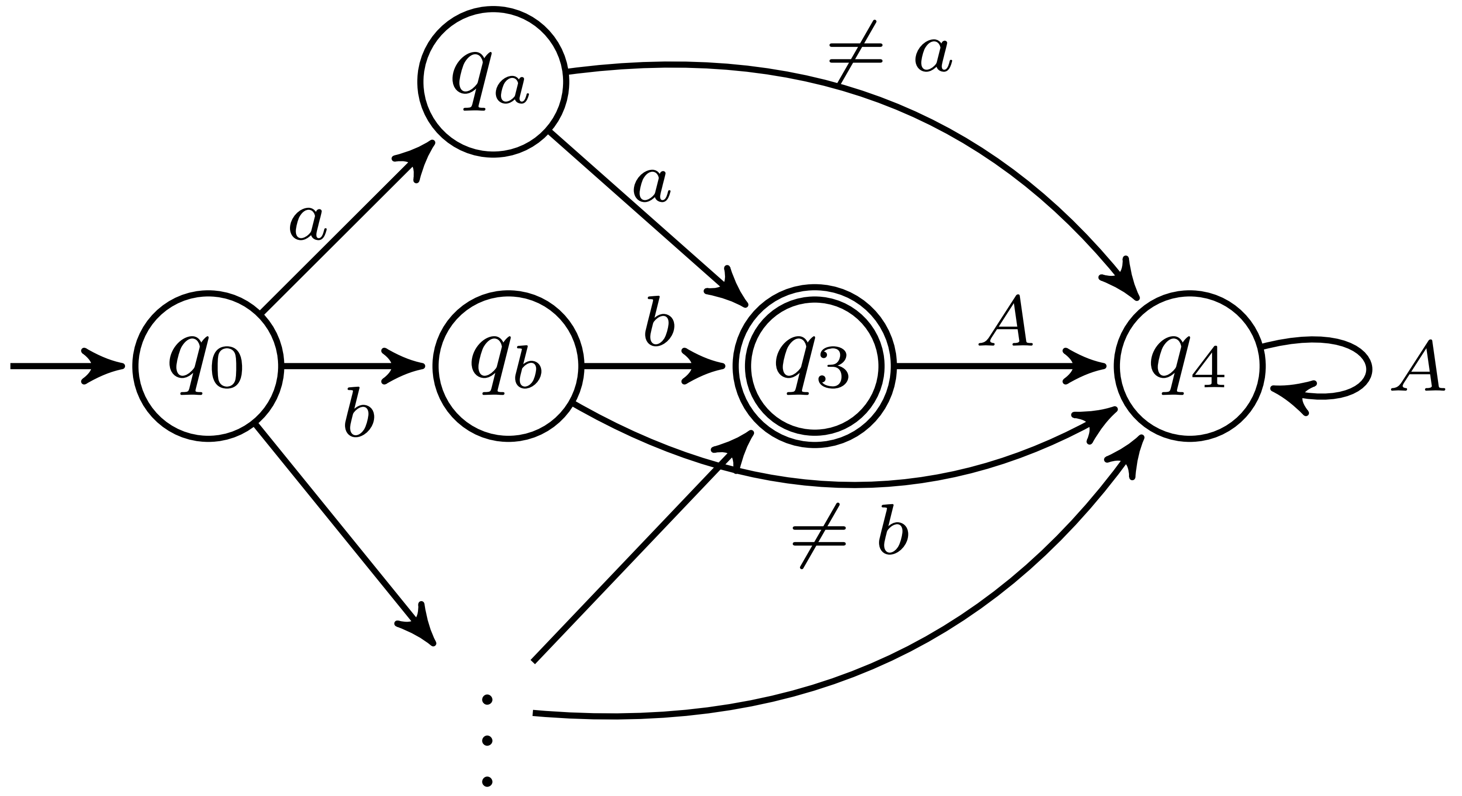
(S, E, row) is *closed* if for all $t \in S \cdot A$ there exists an $s \in S$ such that $row(t) = row(s)$.

Can we develop L* for infinite (nominal) sets?

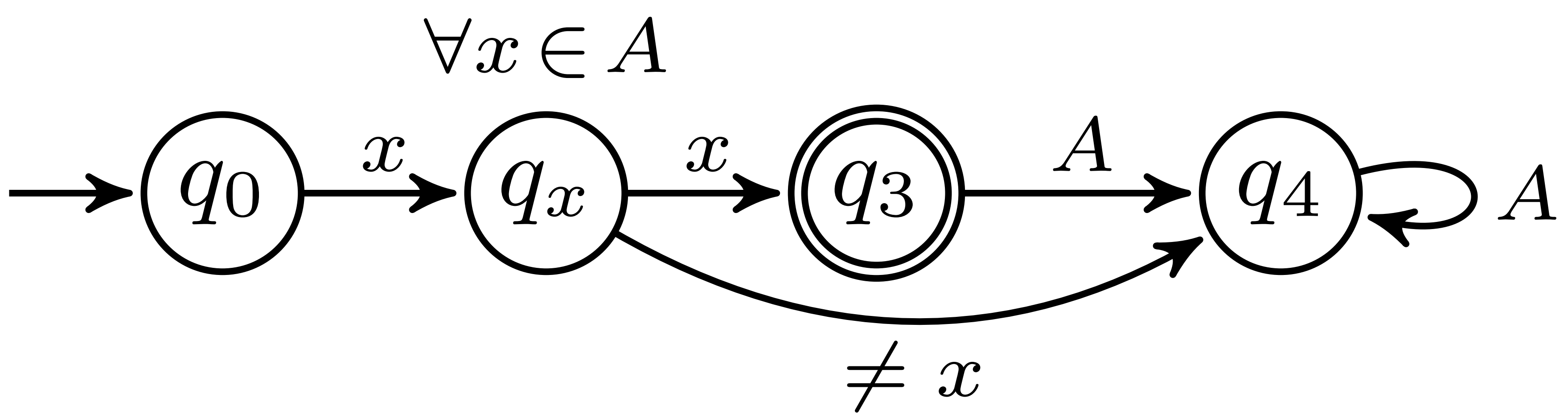
Infinite alphabets

$\mathcal{L}_n = \{ww \mid w \in A^\star, |w| = n\}$ $\quad$ A infinite

$$\mathcal{L}_1 = \{aa, bb, cc, dd, \ldots\}$$

$\mathcal{A}_5 =$ 

infinite automaton



but with a finite representation

Nominal automata

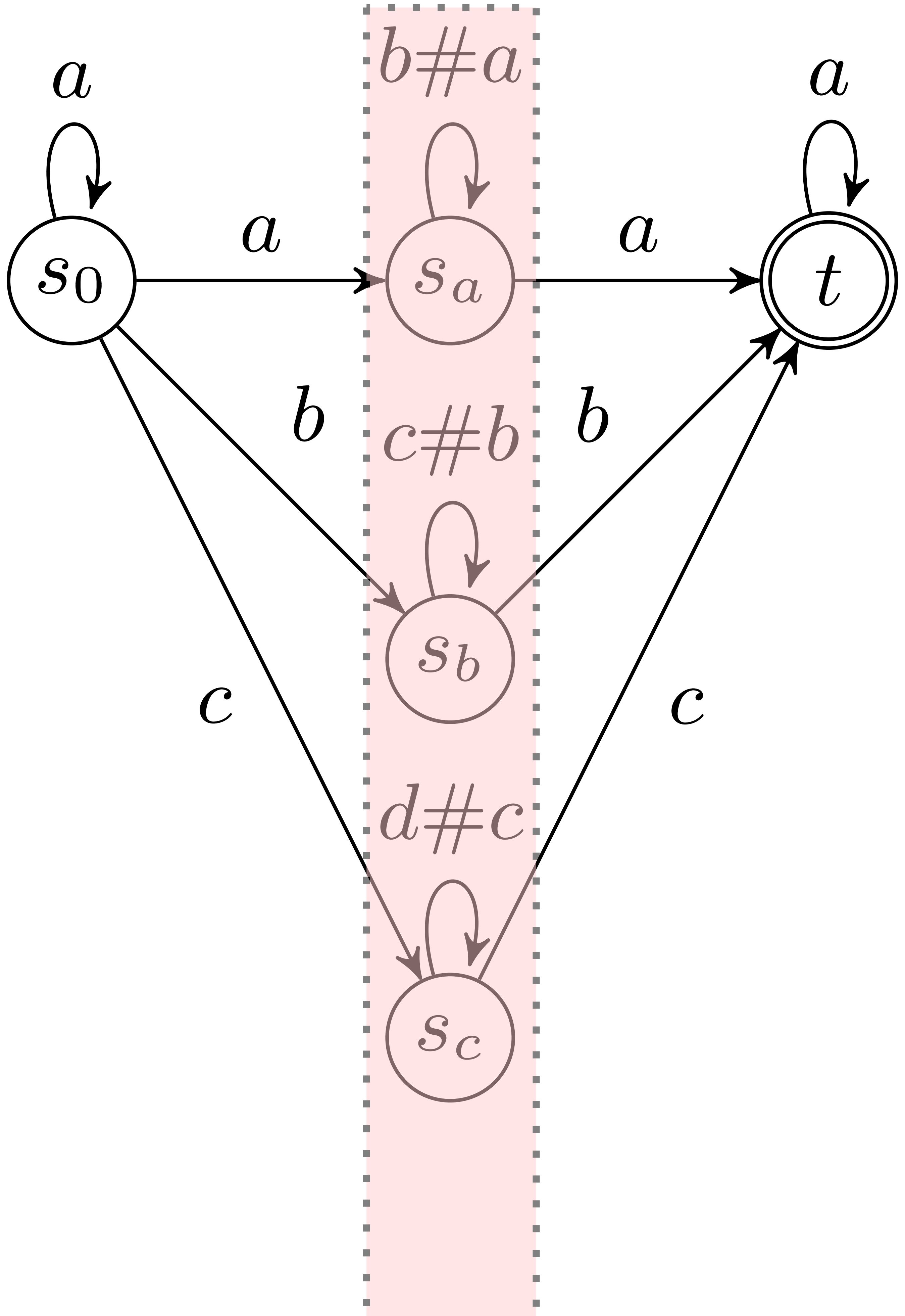
Nominal sets

name binding
alpha-equivalence
.....

Infinite sets with symmetries → Finitely representable

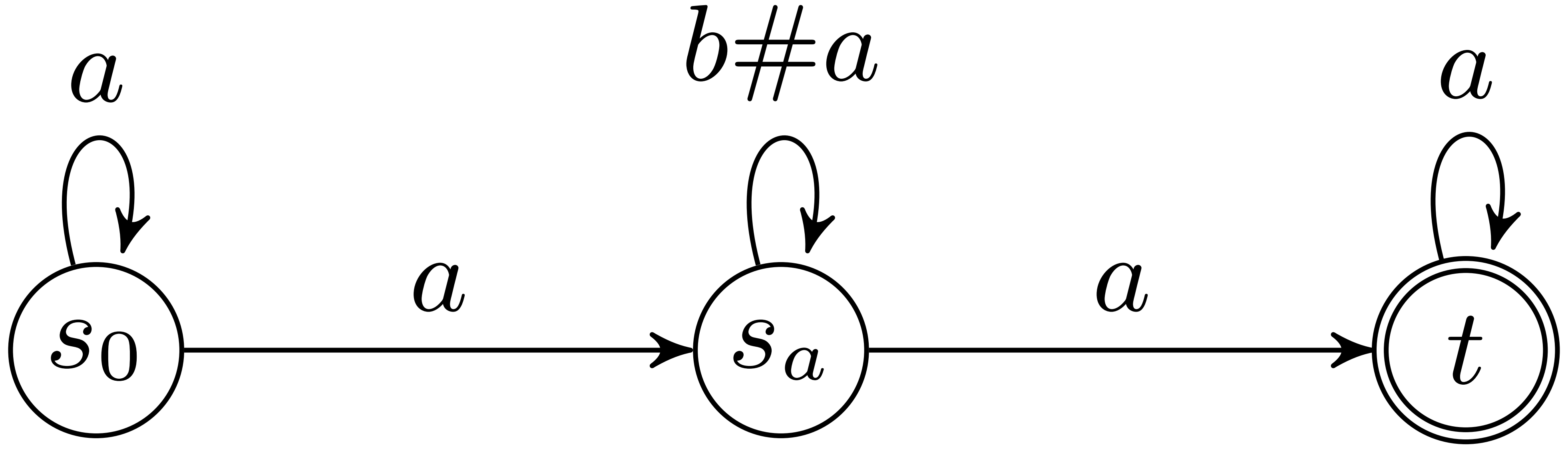
Automata theory
over nominal sets

Nominal automata



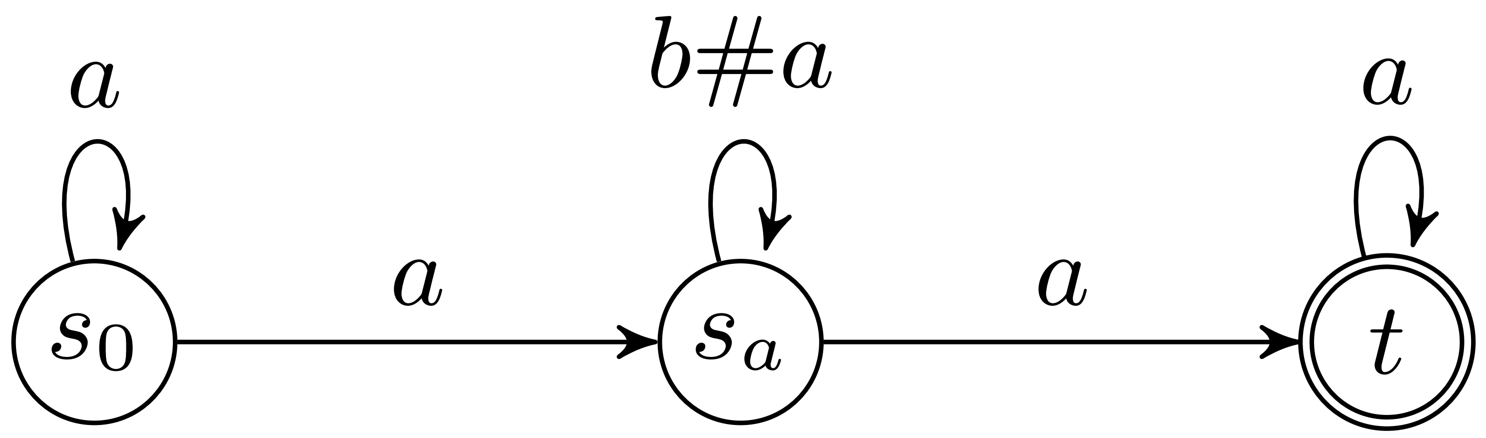
$\mathbb{A}$ infinite

$\{w \in \mathbb{A}^* \mid \exists a.a \text{ occurs twice in } w\}$



finite representation

Nominal automata



$$X \to 2 \times X^A$$

DFA in Nom

$$X = \{s_0\} + \mathbb{A} + \{t\}$$

$$\pi : \mathbb{A} \to \mathbb{A}$$
$$s_a \mapsto s_{\pi a}$$

transition closed under permutations
equivariant

$$s_a \xrightarrow{a} t \Rightarrow s_{\pi a} \xrightarrow{\pi a} t$$

algebraic structure

Challenges

L* LEARNER

$S, E \leftarrow \{\epsilon\}$
repeat
    while (S, E) is not closed or not consistent
    if (S, E) is not closed
        find $s_1 \in S$, $a \in A$ such that
            $row(s_1 a) \neq row(s)$, for all $s \in S$
        $S \leftarrow S \cup \{s_1 a\}$
    if (S, E) is not consistent
        find $s_1, s_2 \in S$, $a \in A$, and $e \in E$ such that
            $row(s_1) = row(s_2)$ and $\mathcal{L}(s_1 ae) \neq \mathcal{L}(s_2 ae)$
        $E \leftarrow E \cup \{ae\}$
    Make the conjecture $M(S, E)$
    if the Teacher replies **no**, with a counter-example t
        $S \leftarrow S \cup \texttt{prefixes}(t)$
until the Teacher replies **yes** to the conjecture $M(S, E)$.
return $M(S, E)$

range over infinite sets

finding witnesses potentially requires checking infinite rows

t has only finitely many prefixes, but an infinite S is necessary

Challenges

L* LEARNER

$S, E \leftarrow \{\epsilon\}$
repeat
  while (S, E) is not closed or not consistent
  if (S, E) is not closed
    find $s_1 \in S$, $a \in A$ such that
      $row(s_1 a) \neq row(s)$, for all $s \in S$
    $S \leftarrow S \cup \{s_1 a\}$
  if (S, E) is not consistent
    find $s_1, s_2 \in S$, $a \in A$, and $e \in E$ such that
      $row(s_1) = row(s_2)$ and $\mathcal{L}(s_1 ae) \neq \mathcal{L}(s_2 ae)$
    $E \leftarrow E \cup \{ae\}$
  Make the conjecture $M(S, E)$
  if the Teacher replies **no**, with a counter-example t
    $S \leftarrow S \cup \texttt{prefixes}(t)$
until the Teacher replies **yes** to the conjecture $M(S, E)$.
return $M(S, E)$

range over infinite sets

finding witnesses potentially requires checking infinite rows

t has only finitely many prefixes, but an infinite S is necessary

(P1) the sets S, S·A and E admit a finite representation up to permutations;
(P2) row is such that row(π(s))(π(e)) = row(s)(e), for all s ∊ S and e ∊ E.
Observation table admits a finite symbolic representation.

Nominal L*

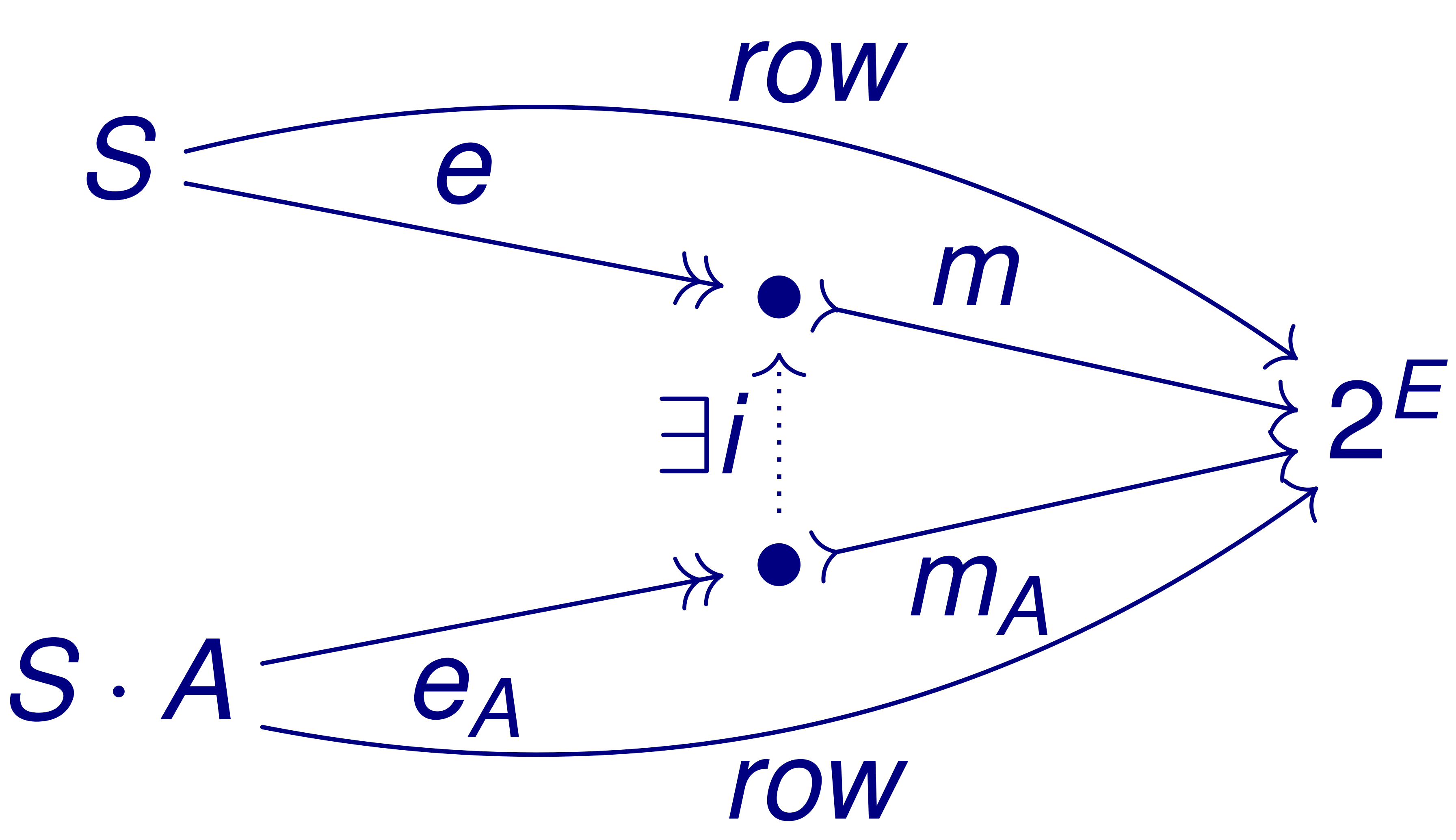
$$
\begin{array}{ll}
6' & S \leftarrow S \cup \texttt{orb}(sa) \\
9' & E \leftarrow E \cup \texttt{orb}(ae) \\
12' & E \leftarrow E \cup \texttt{prefixes}(\texttt{orb}(t))
\end{array}
$$

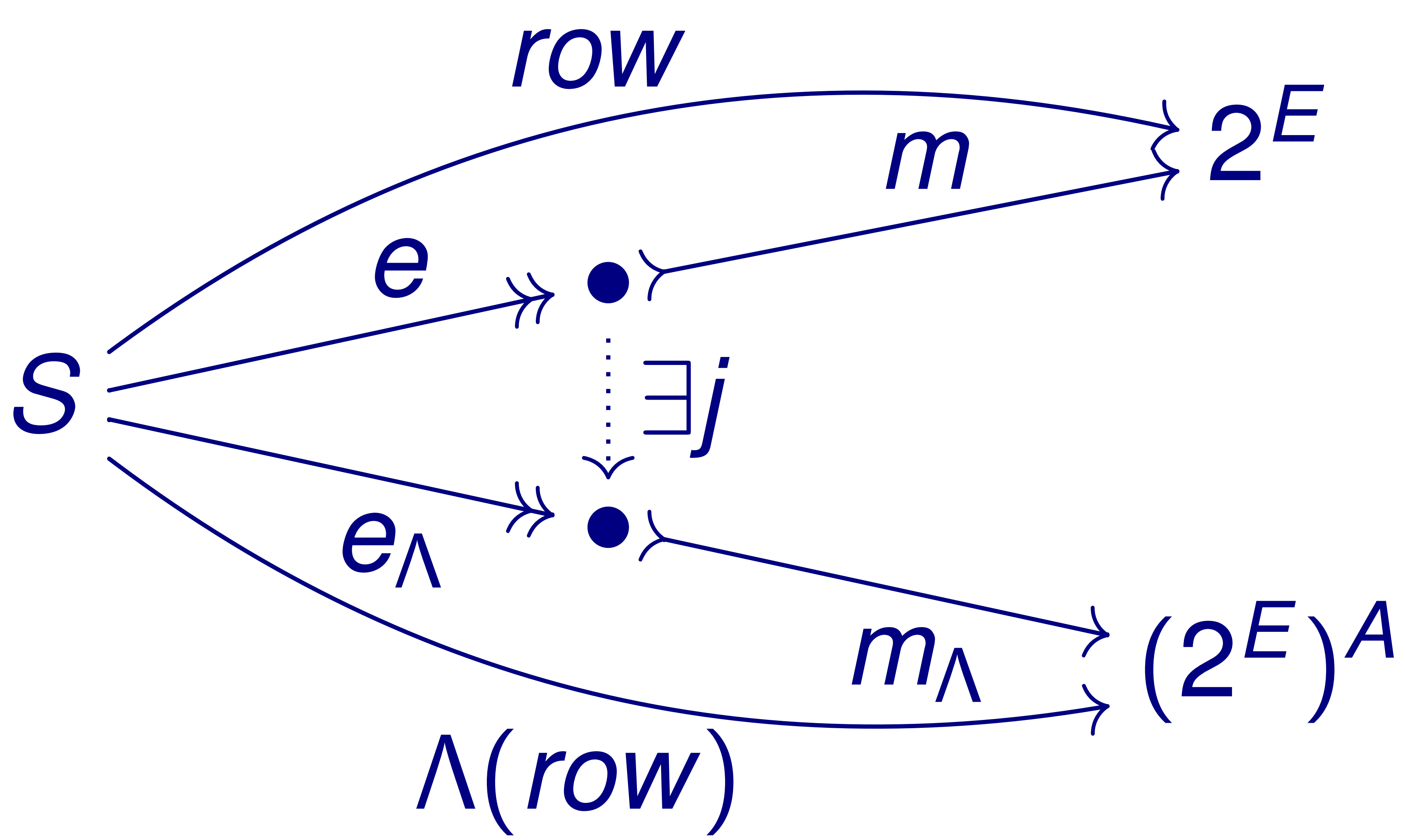
only 3 lines changed!

not really… all definitions have to be adapted to nominal/equivariant.

Correctness, termination, … have to be re-proved!

Categorical glasses





(S, E, row) is *closed* if for all $t \in S \cdot A$ there exists an $s \in S$ such that $row(t) = row(s)$.

Pretty.... but is it useful?

(S, E, row) is *consistent* if whenever $s_1, s_2 \in S$ are such that $row(s_1) = row(s_2)$, for all $a \in A$, $row(s_1 a) = row(s_2 a)$.

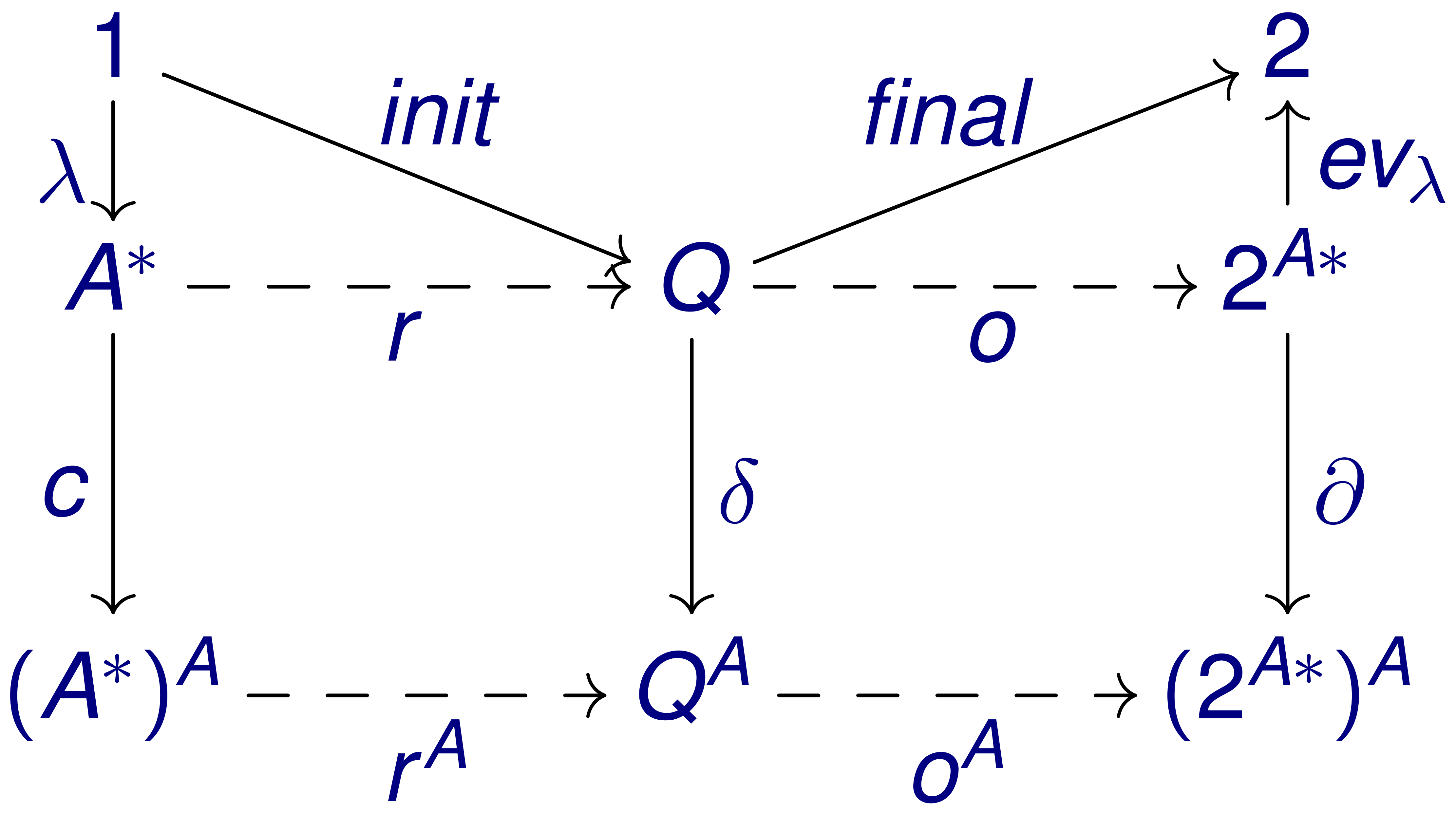
The power of abstraction

$$X \to 2 \times X^A$$

DFA in Nom

Definitions are the *same*

Proof of correctness is the *same*



Abstract automata

Category $\mathbf{C}$ **= universe of state-spaces**

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ **= automaton type**

$$
\begin{array}{ccccc}
 & & FQ & & \\
 & & \big\downarrow{\scriptstyle \delta_Q} & & \\
 & \mathsf{init}_Q & Q & \mathsf{out}_Q & \\
 & \nearrow & & \searrow & \\
I & & & & Y
\end{array}
$$

Abstract automata

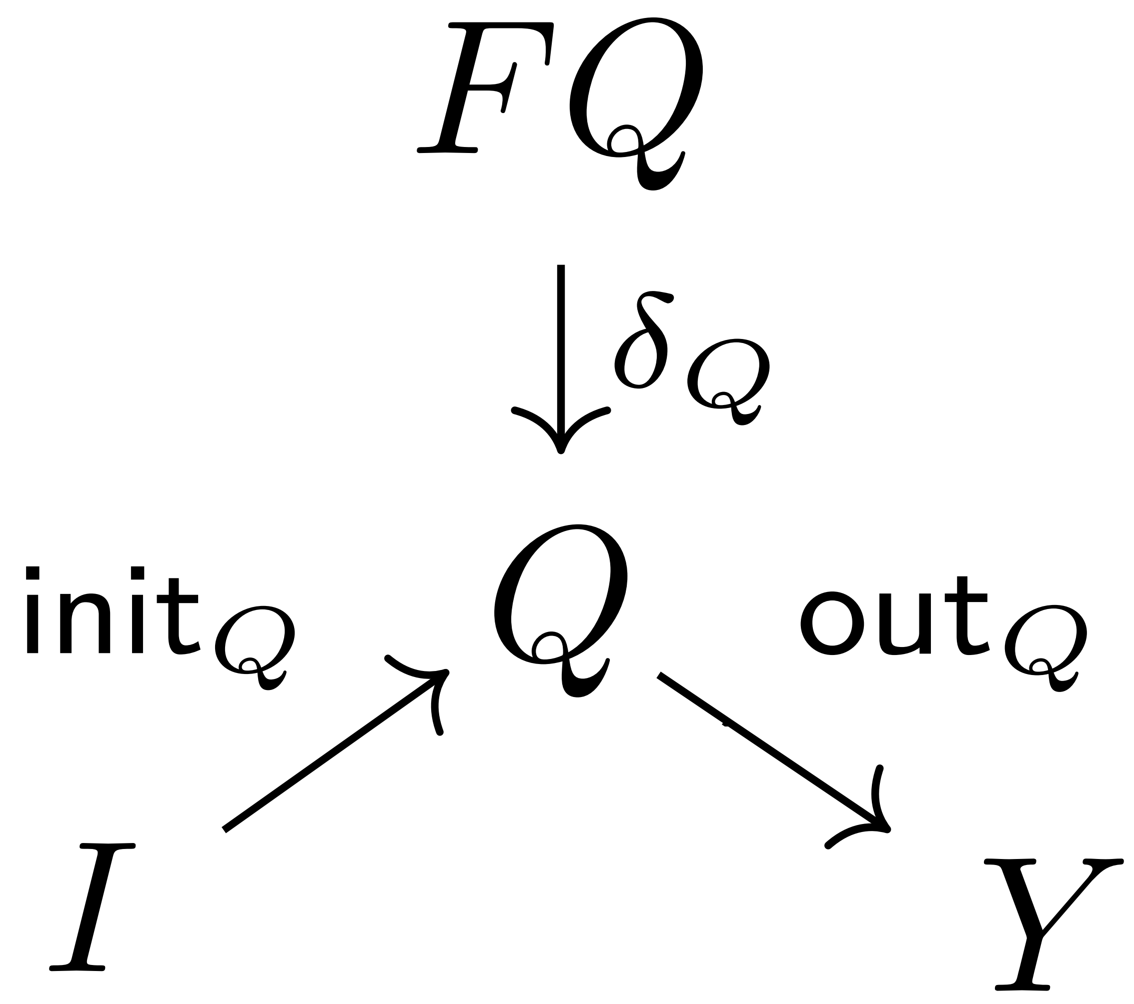
Category $\mathbf{C}$ **= universe of state-spaces**

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ **= automaton type**

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$



Abstract automata

Category $\mathbf{C}$ **= universe of state-spaces**

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ **= automaton type**

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$

$$Q \times A \xrightarrow{\delta_Q} Q$$

$$I \xrightarrow{\mathsf{init}_Q} Q \xrightarrow{\mathsf{out}_Q} Y$$

Abstract automata

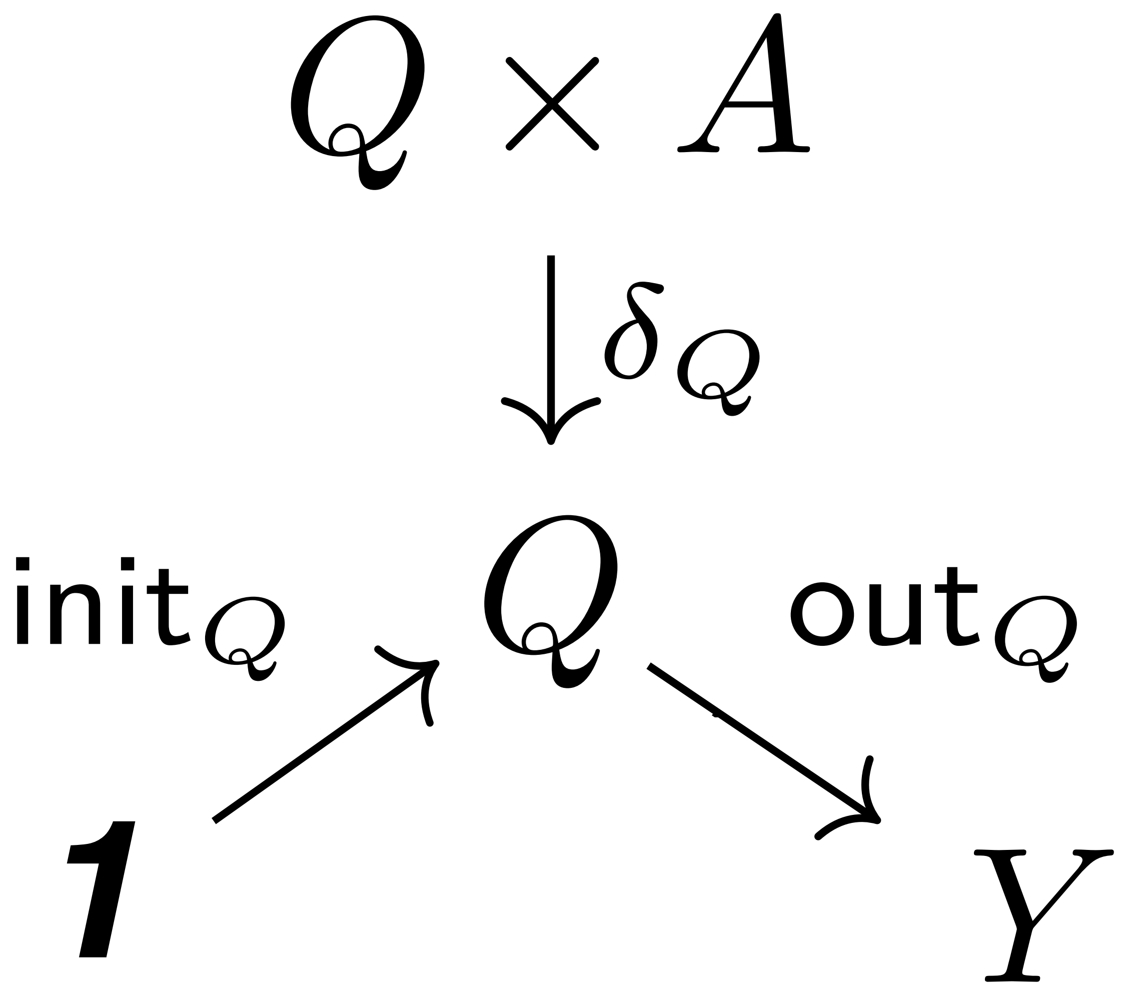
Category $\mathbf{C}$ = universe of state-spaces

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ = automaton type

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$



Abstract automata

Category $\mathbf{C}$ **= universe of state-spaces**

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ **= automaton type**

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$

$$Q \times A$$

$$\downarrow \delta_Q$$

$$\mathbf{1} \xrightarrow{\mathsf{init}_Q} Q \xrightarrow{\mathsf{out}_Q} Y$$

$q_0 \in Q$

Abstract automata

Category $\mathbf{C}$ = universe of state-spaces

Endofunctor $F\colon \mathbf{C} \to \mathbf{C}$ = automaton type

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$

$$Q \times A \xrightarrow{\delta_Q} Q$$

$$\mathbf{1} \xrightarrow{\mathsf{init}_Q} Q \xrightarrow{\mathsf{out}_Q} \mathbf{2}$$

$q_0 \in Q$

Abstract automata

Category $\mathbf{C}$ = universe of state-spaces

Endofunctor $F : \mathbf{C} \to \mathbf{C}$ = automaton type

DFAs

$\mathbf{C} = \mathbf{Set}$

$F = (-) \times A$

$$Q \times A$$

$$\downarrow \delta_Q$$

$$\mathbf{1} \xrightarrow{\mathsf{init}_Q} Q \xrightarrow{\mathsf{out}_Q} \mathbf{2}$$

$q_0 \in Q$ $\qquad$ $F \subseteq Q$

Abstract learning

Abstract observation data structure

approximates

Target minimal automaton

$$\begin{array}{ccc} & FQ & \\ & \big\downarrow \delta_Q & \\ & Q & \\ \text{init}_Q \nearrow & & \searrow \text{out}_Q \\ I & & Y \end{array}$$

abstract closedness and consistency

Hypothesis automaton

$$\begin{array}{ccc} & FH & \\ & \big\downarrow \delta_H & \\ & H & \\ \text{init}_H \nearrow & & \searrow \text{out}_H \\ I & & Y \end{array}$$

General correctness theorem

Guidelines for implementation

CALF: Categorical Automata Learning Framework **(arXiv:1704.05676)**

Gerco van Heerdt, Matteo Sammartino, Alexandra Silva

Other automata & optimizations

Change base category

Learning Nominal Automata (POPL '17)

Joshua Moerman, Matteo Sammartino, Alexandra Silva, Bartek Klin, Michal Szynwelski

Set	**DFAs**
Nom	**Nominal automata**
Vect	**Weighted automata**

Change main data structure

Discrimination trees

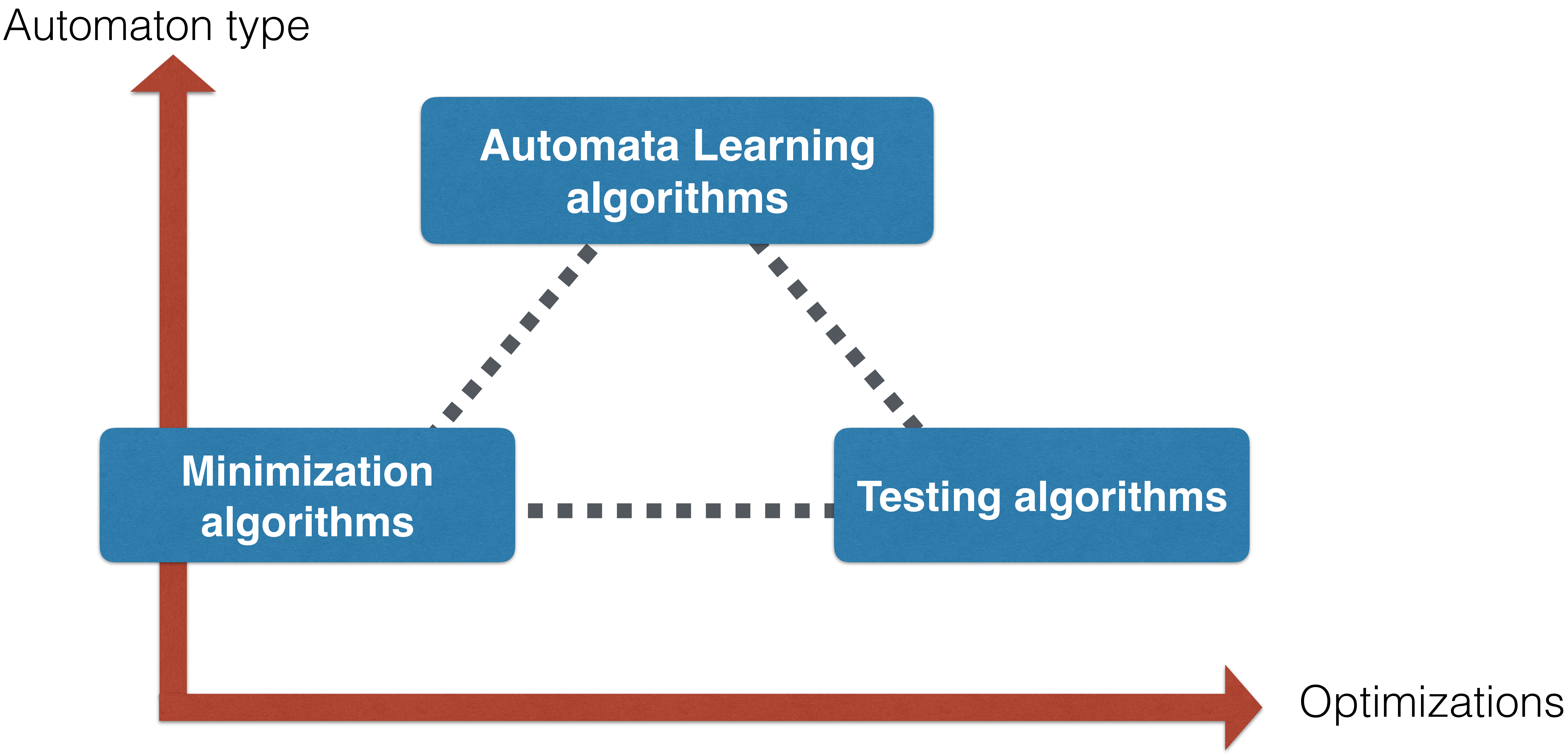
Learning Automata with Side-effects (arXiv:1704.08055)

Gerco van Heerdt, Matteo Sammartino, Alexandra Silva

Side-effects (via monads)

Powerset	**NFAs**
Powerset with intersection	**Universal automata**
Double powerset	**Alternating automata**
Maybe monad	**Partial automata**

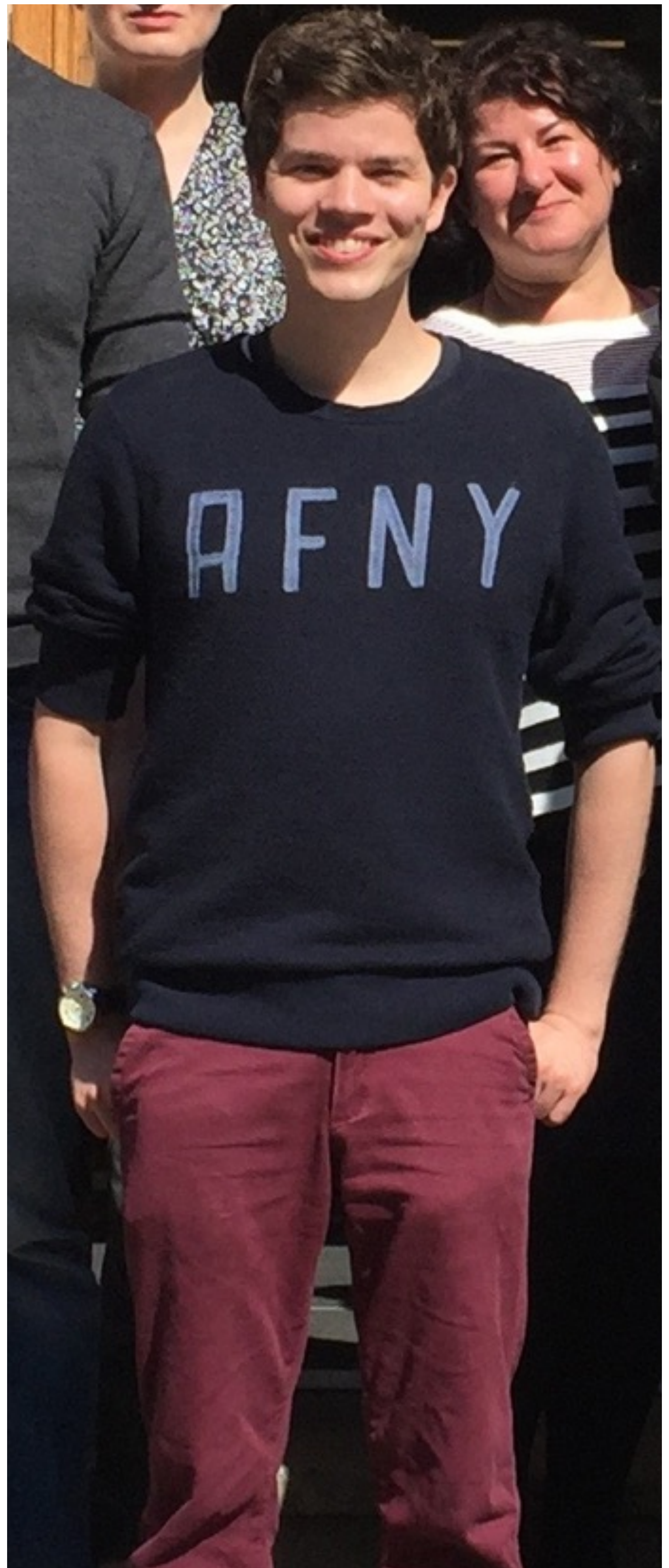
Connections with other algorithms



Ongoing and future work

- **Library & tool** to learn control + data-flow models (as **nominal automata**)
- Applications:
  - Specification mining
  - Network verification, with amazon
  - Verification of cryptographic protocols
  - Ransomware detection

Ongoing and future work



Learning convex automata

Rich algebraic structure

Challenging analytical properties

Conclusions

Category theory is a good playground to understand and generalise algorithms

Unveils connections and sets the scene

—

No free lunch

Questions?