Discrete and Continuous Models for Concurrent Systems

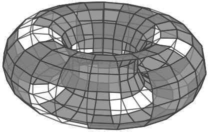
3. Languages of Higher-Dimensional Automata

Uli Fahrenberg

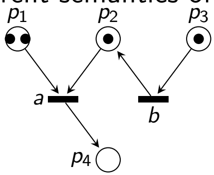
EPITA Research Laboratory (LRE), Rennes/Paris, France

EWSCS 27, Viinistu, March 2025

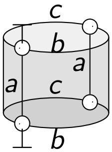
1. The geometry of concurrency



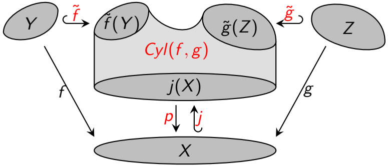
2. Concurrent semantics of Petri nets



3. Languages of higher-dimensional aut.



4. Geometry of higher-dimensional aut.



1. Introduction
2. Higher-Dimensional Automata
3. Languages of Higher-Dimensional Automata
4. Properties

Higher-Dimensional Automata

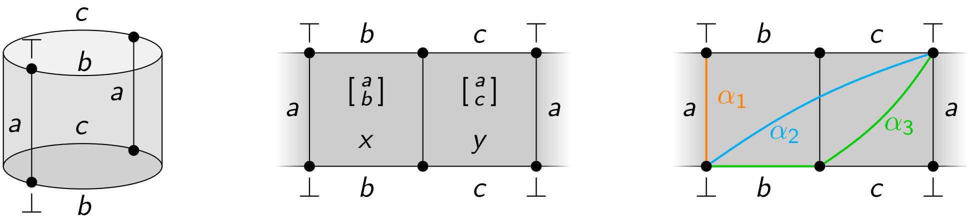
A conclist is a finite, totally ordered, Σ-labeled set. (a list of labeled events)

A precubical set X consists of:

- A set of cells X (cubes)
- Every cell $x \in X$ has a conclist $\mathrm{ev}(x)$ (list of events active in x)
- We write $X[U] = \{x \in X \mid \mathrm{ev}(x) = U\}$ for a conclist U (cells of type U)
- For every conclist U and $A \subseteq U$ there are:
  upper face map $\delta^1_A : X[U] \to X[U \setminus A]$ (terminating events A)
  lower face map $\delta^0_A : X[U] \to X[U \setminus A]$ ("unstarting" events A)
- Precube identities: $\delta^\mu_A \delta^\nu_B = \delta^\nu_B \delta^\mu_A$ for $A \cap B = \emptyset$ and $\mu, \nu \in \{0, 1\}$

A higher dimensional automaton (HDA) is a precubical set X with initial cells $\bot \subseteq X$ and accepting cells $\top \subseteq X$ (not necessarily vertices)

Example

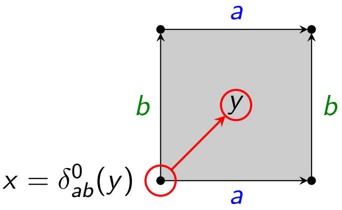


$$a \parallel (bc)^*$$

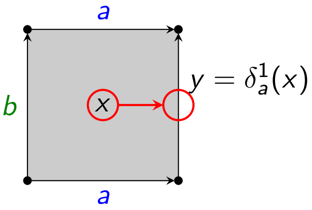
Computations of HDAs

An HDA computes by starting and terminating events in sequence:

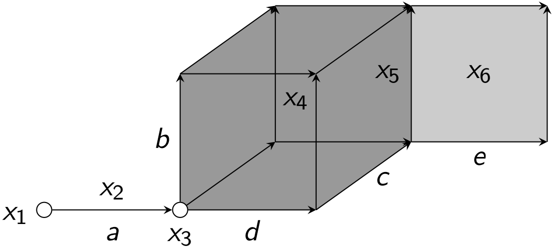
upstep $x \nearrow y$, starting $\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$:



downstep $x \searrow y$, terminating a:



Example



$$\left(x_1 \nearrow^{a} x_2 \searrow_{a} x_3 \nearrow^{\{b,c,d\}} x_4 \searrow_{\{c,d\}} x_5 \nearrow^{e} x_6\right)$$

Precubical Sets As Presheaves

A presheaf over a category $\mathcal{C}$ is a functor $\mathcal{C}^{\text{op}} \to \text{Set}$ (contravariant functor on $\mathcal{C}$)

The precube category $\square$ has conclists as objects.
Morphisms are coface maps $d_{A,B} : U \to V$, where

- $A, B \subseteq V$ are disjoint subsets,
- $U \simeq V \setminus (A \cup B)$ are isomorphic conclists,
- $d_{A,B} : U \to V$ is the unique label-preserving monotonic map with image $V \setminus (A \cup B)$.

Composition of coface maps $d_{A,B} : U \to V$ and $d_{C,D} : V \to W$ is

$$d_{\partial(A) \cup C, \partial(B) \cup D} : U \to W,$$

where $\partial : V \to W \setminus (C \cup D)$ is the unique conclist isomorphism.

- precubical sets: presheaves over $\square$
- HDAs: precubical sets with initial and accepting cells

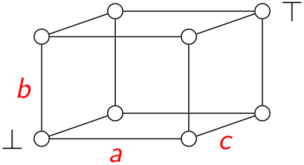
1 Introduction

2 Higher-Dimensional Automata

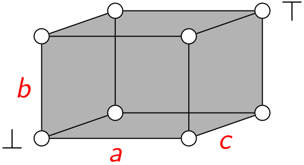
3 Languages of Higher-Dimensional Automata

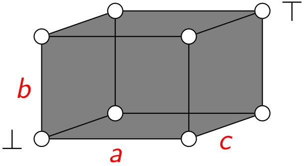
4 Properties

Languages of HDAs: Examples



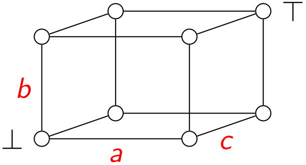
$$L_1 = \{abc, acb, bac, bca, cab, cba\}$$



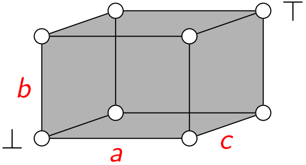


$$L_3 = \left\{ \begin{bmatrix} a \\ b \\ c \end{bmatrix}, \dots \right\}$$

Languages of HDAs: Examples

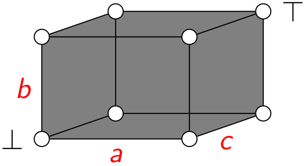


$$L_1 = \{abc, acb, bac, bca, cab, cba\}$$



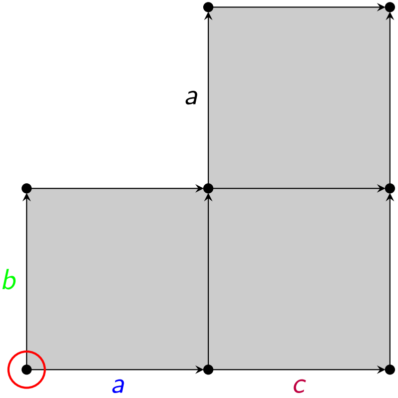
$$L_2 = \left\{ \begin{bmatrix} a & \\ b \to c \end{bmatrix}, \begin{bmatrix} a & \\ c \to b \end{bmatrix}, \begin{bmatrix} b & \\ a \to c \end{bmatrix}, \begin{bmatrix} b & \\ c \to a \end{bmatrix}, \begin{bmatrix} c & \\ a \to b \end{bmatrix}, \begin{bmatrix} c & \\ b \to a \end{bmatrix} \right\} \cup L_1 \cup \dots$$

sets of pomsets

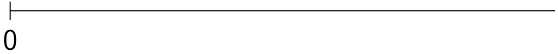


$$L_3 = \left\{ \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\} \cup L_2$$

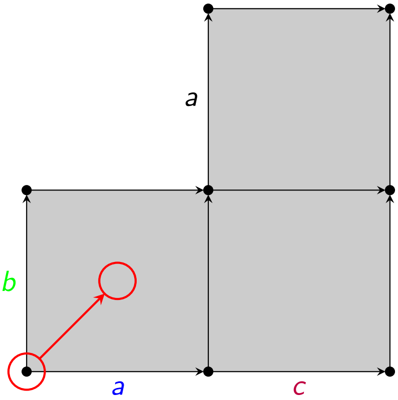
Event Ipomset of a Path



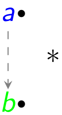
Lifetimes of events



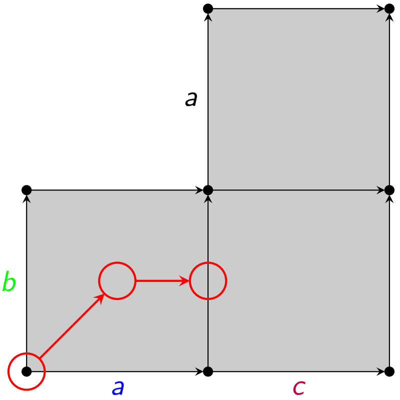
Event Ipomset of a Path



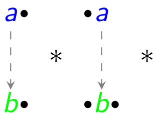
Lifetimes of events



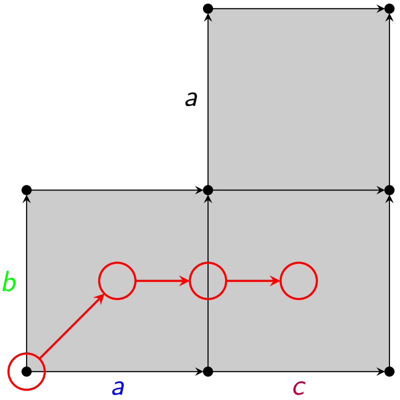
Event Ipomset of a Path



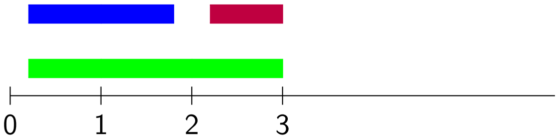
Lifetimes of events



Event Ipomset of a Path

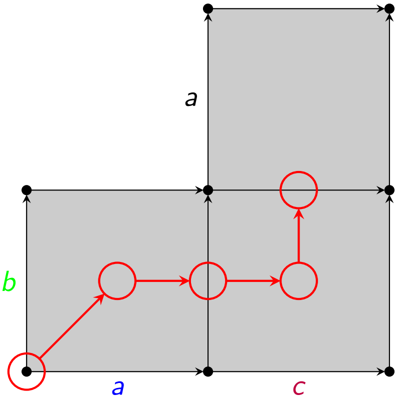


Lifetimes of events

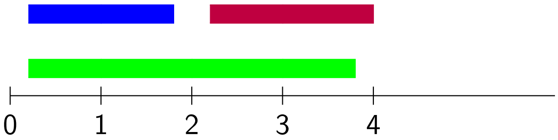


$$\begin{matrix} a\bullet \\ b\bullet \end{matrix} \quad * \quad \begin{matrix} \bullet a \\ \bullet b\bullet \end{matrix} \quad * \quad \begin{matrix} c\bullet \\ \bullet b\bullet \end{matrix} \quad *$$

Event Ipomset of a Path

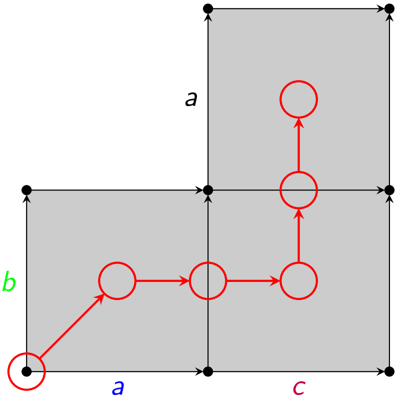


Lifetimes of events

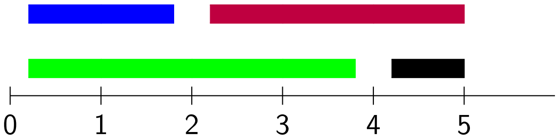


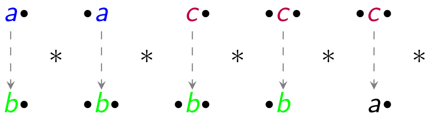
$$\begin{array}{c} a\bullet \\ \downarrow \\ b\bullet \end{array} * \begin{array}{c} \bullet a \\ \downarrow \\ \bullet b\bullet \end{array} * \begin{array}{c} c\bullet \\ \downarrow \\ \bullet b\bullet \end{array} * \begin{array}{c} \bullet c\bullet \\ \downarrow \\ \bullet b \end{array} *$$

Event Ipomset of a Path

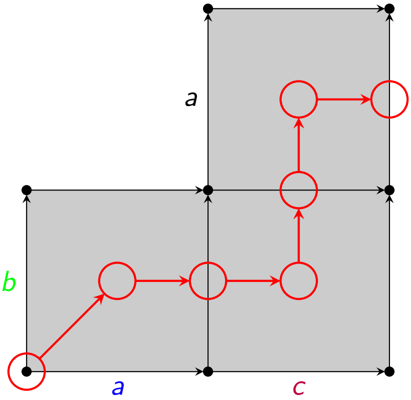


Lifetimes of events

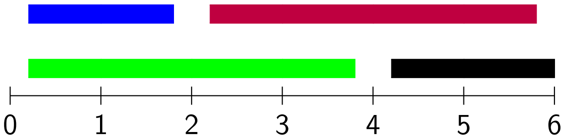


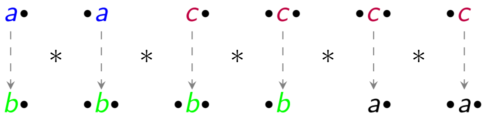


Event Ipomset of a Path

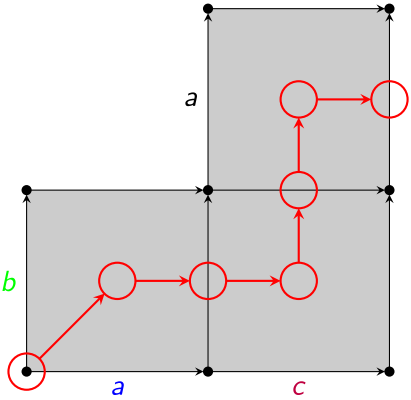


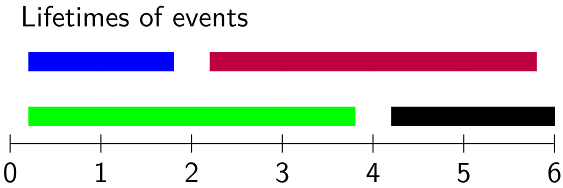
Lifetimes of events

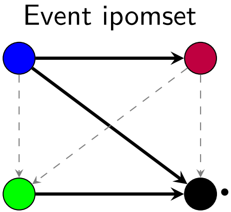




Event Ipomset of a Path







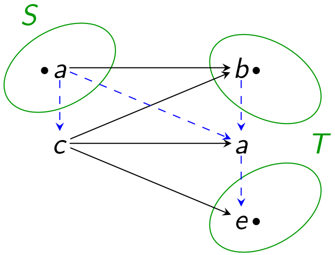
(not series-parallel!)

Pomsets with Interfaces

Definition

A pomset with interfaces (ipomset): $(P, <, \dashrightarrow, S, T, \lambda)$:

- finite set P;
- two partial orders $<$ (precedence order), $\dashrightarrow$ (event order)
  - s.t. $< \cup \dashrightarrow$ is a *total relation*;
- $S, T \subseteq P$ source and target interfaces
  - s.t. S is $<$-minimal and T is $<$-maximal.

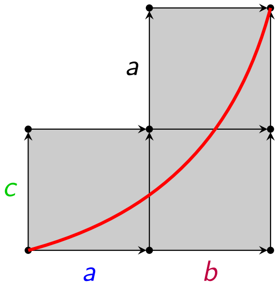


Interval Orders

Definition

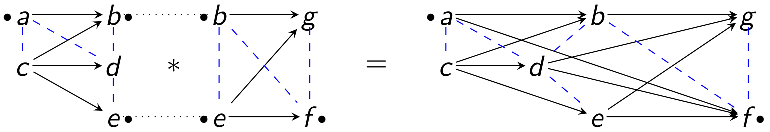
An ipomset $(P, <_P, \dashrightarrow, S, T, \lambda)$ is interval if $(P, <_P)$ has an interval representation: functions $b, e : P \to \mathbb{R}$ s.t.

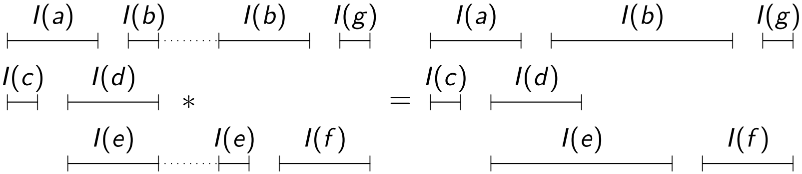
- $\forall x \in P : b(x) \leq_{\mathbb{R}} e(x)$;
- $\forall x, y \in P : e(x) <_{\mathbb{R}} b(y) \iff x <_P y$



$$\begin{array}{ccc} a & \longrightarrow & b \\ & \searrow & \\ c & \longrightarrow & a \end{array}$$

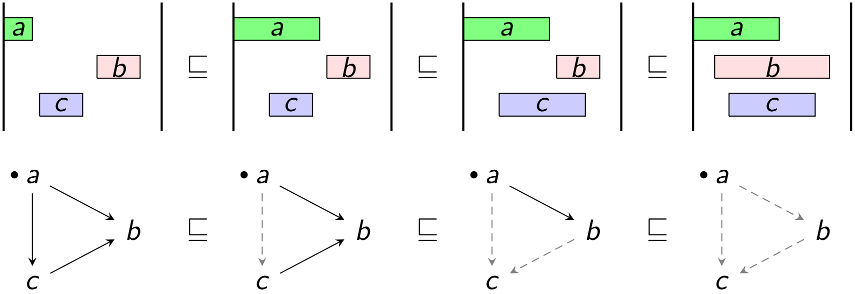
Gluing Composition





- Gluing $P * Q$: P before Q, except for interfaces (which are identified)
- (also have parallel composition $P \parallel Q$: disjoint union)

Subsumption



P refines Q / Q subsumes P / $P \sqsubseteq Q$ iff

- P and Q have same interfaces
- P has more $<$ than Q
- Q has more $\dashrightarrow$ than P

Languages of HDAs

Definition

The language of an HDA X is the set of event ipomsets of all accepting paths:

$$L(X) = \{\mathrm{ev}(\pi) \mid \pi \in \mathrm{Paths}(X), \mathrm{src}(\pi) \in \bot_X, \mathrm{tgt}(\pi) \in \top_X\}$$

- $L(X)$ contains only interval ipomsets,
- is closed under subsumption,
- and has finite width

Definition

A language $L \subseteq \mathrm{iiPoms}$ is regular if there is an HDA X with $L = L(X)$.

1 Introduction

2 Higher-Dimensional Automata

3 Languages of Higher-Dimensional Automata

4 Properties

Theorems

Definition (Rational Languages over Σ)

- Generated by $\emptyset$, $\{\epsilon\}$, and all $\{[a]\}$, $\{[\bullet\, a]\}$, $\{[a\, \bullet]\}$, $\{[\bullet\, a\, \bullet]\}$ for $a \in \Sigma$
- under operations $\cup$, $*$, $\|$ and (Kleene plus) $^+$
- (these need to take subsumption closure into account)

Definition (Monadic Second-Order Logics over Ipomsets)

$$\psi ::= a(x) \mid \mathsf{s}(x) \mid \mathsf{t}(x) \mid x < y \mid x \dashrightarrow y \mid x \in X \mid \exists x.\, \psi \mid \forall x.\, \psi \mid \exists X.\, \psi \mid \forall X.\, \psi \mid \psi_1 \wedge \psi_2 \mid \psi_1 \vee \psi_2 \mid \neg\psi$$

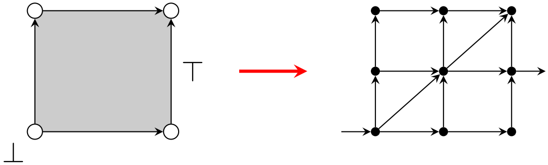
Theorem (à la Kleene): regular $\iff$ rational

Theorem (à la Myhill-Nerode): regular $\iff$ finite prefix quotient

Theorem (à la Büchi-Elgot-Trakhtenbrot):
regular $\iff$ MSO-definable, of finite width, and subsumption-closed

Kleene Theorem: Easy Parts

- regular $\Longrightarrow$ rational: by reduction to ST-automata



- rational $\Longrightarrow$ regular: generators:

$L(X)$	$\emptyset$	$\{\epsilon\}$	$\{[a]\}$	$\{[\bullet\, a]\}$	$\{[a\, \bullet]\}$	$\{[\bullet\, a\, \bullet]\}$
X	$\emptyset$	$\perp \circ \top$	$\top$ a $\perp$	$\top$ $\perp$ a	a $\top$ $\perp$	$\perp$ a $\top$

- rational $\Longrightarrow$ regular: $\cup$ and $\|$

$$L(X) \cup L(Y) = L(X \sqcup Y) \qquad L(X) \parallel L(Y) = L(X \otimes Y)$$

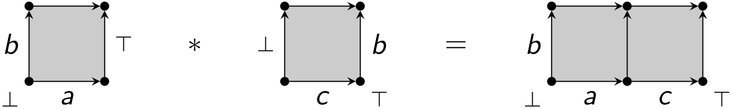
Kleene Theorem: Difficult Parts

- miss to see: gluings and iterations of regular languages are regular:

$$L(X) * L(Y) = L(X * Y) \qquad L(X)^+ = L(X^+)$$

- much more difficult: higher-dimensional gluings identify too much

- for example:



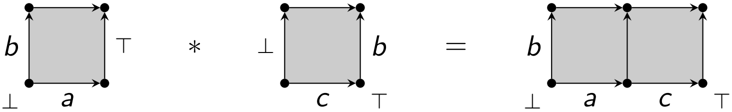
$L_2 = \emptyset$ $\qquad$ $L_1 * L_2 \ni ac$

Kleene Theorem: Difficult Parts

- miss to see: gluings and iterations of regular languages are regular:

$$L(X) * L(Y) = L(X * Y) \qquad L(X)^+ = L(X^+)$$

- much more difficult: higher-dimensional gluings identify too much
- for example:



$L_2 = \emptyset$ $\qquad$ $L_1 * L_2 \ni ac$

- use HDAs with interfaces and cylinder objects

HDAs with Interfaces

A conclist with interfaces (iconclist) is a conclist U with subsets $S \subseteq U \supseteq T$, denoted ${}_S U_T$
(events in T cannot be terminated; events in S cannot be "unstarted")

A precubical set with interfaces (ipc-set) X consists of a set of cells X such that:

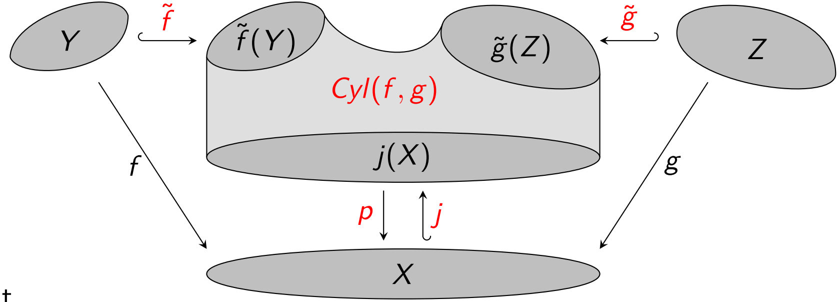
- Every cell $x \in X$ has an iconclist $\text{ev}(x)$
- We write $X[{}_S U_T] = \{x \in X \mid \text{ev}(x) = {}_S U_T\}$.
- For every $A \subseteq U - S$ there is a lower face map $\delta_A^0 : X[U] \to X[{}_S U_T - A]$.
- For every $B \subseteq U - T$ there is an upper face map $\delta_B^1 : X[U] \to X[{}_S U_T - b]$.
- Precubical identities: $\delta_A^\mu \delta_B^\nu = \delta_B^\nu \delta_A^\mu$ for $A \cap B = \emptyset$ and $\mu, \nu \in \{0, 1\}$
- (presheaves over a category $\mathrm{I}\square$)

An HDA with interfaces (iHDA) is a finite ipc-set with start and accept cells.

Extra conditions:
If $x \in X[{}_S U_T]$ is a start cell, then $S = U$.
If $x \in X[{}_S U_T]$ is an accept cell, then $T = U$.

Cylinders

Let X, Y, Z be ipc-sets and $f : Y \to X$, $g : Z \to X$ ipc-maps with $f(Y) \cap g(Z) = \emptyset$
There is a diagram of ipc-sets



such that

- $\tilde f$ is an initial inclusion;
- $\tilde g$ is a final inclusion;
- all paths in X from $f(Y)$ to $g(Z)$ lift to paths in $Cyl(f, g)$.

Cylinders: Construction

X, Y, Z: ipc-sets, $f : Y \to X$, $g : Z \to X$: ipc-maps with $f(Y) \cap g(Z) = \emptyset$.
For ${}_S U_T \in \mathrm{I}\square$ let

$$Cyl(f,g)[{}_S U_T] = \{(x, K, L, \varphi, \psi)\}$$

such that

- $x \in X[{}_S U_T]$;
- $K \subseteq \mathrm{I}\square^U$ is an initial subset;
- $L \subseteq \mathrm{I}\square^U$ is a final subset;
- $\varphi : K \to Y$, $\psi : L \to Z$ are ipc-maps satisfying $f \circ \varphi = \iota_x|_K$ and $g \circ \psi = \iota_x|_L$:

$$\begin{tikzcd}
K \arrow[r, hook] \arrow[d, "\varphi"'] & \mathrm{I}\square^U \arrow[d, "\iota_x"] & L \arrow[l, hook'] \arrow[d, "\psi"] \\
Y \arrow[r, "f"] & X & Z \arrow[l, "g"']
\end{tikzcd}$$

Gluing Composition of Regular Languages Is Regular

Proposition

Gluing composition of regular languages is regular.

Proof sketch: Let L and M be regular languages.

1. We may ssume that L, M are simple, i.e., $L = L(X)$, $M = L(Y)$ for iHDAs X, Y having one initial and one accepting cell each.
2. Now replace X by $X' = Cyl(X \leftarrow \top_X : j)$ and Y by $Y' = Cyl(i : \bot_Y \to Y)$, then $L(X') = L(X)$ and $L(Y') = L(Y)$.
3. Go back to HDA and glue:

$$L(\mathbf{Cl}(X') * \mathbf{Cl}(Y')) = L(X') * L(Y') = L * M.$$

(closure $\mathbf{Cl}$: iHDA $\to$ HDA "adds missing cells")

4. So $L * M$ is recognized by a finite HDA, hence regular.

Selected Bibliography

- U.F., C.Johansen, G.Struth, K.Ziemiański: *Kleene theorem for higher-dimensional automata*. Logical Meth.Comput.Sci. 2024
- U.F., K.Ziemiański: *Myhill-Nerode theorem for higher-dimensional automata*. Fund.Inf. 2024
- A.Amrane, H.Bazille, U.F., K.Ziemiański: *Closure and decision properties for higher-dimensional automata*. Theor.Comput.Sci. 2025
- A.Amrane, H.Bazille, U.F., M.Fortin: *Logic and languages of higher-dimensional automata*. DLT 2024