An Invitation to Higher-Dimensional Automata Theory

Uli Fahrenberg

LRE & EPITA Rennes, France

10 February 2023

Languages of higher-dimensional automata

- Generating Posets Beyond N. RAMiCS 2020
- Languages of Higher-Dimensional Automata. MSCS 2021
- Posets with Interfaces as a Model for Concurrency. I&C 2022
- A Kleene Theorem for Higher-Dimensional Automata. CONCUR 2022
- A Myhill-Nerode Theorem for Higher-Dimensional Automata. arxiv 2022

Today:

1. What are HDAs (and why should I be interested)?
2. What are languages of HDAs (and why should I be interested)?
3. What can I do with languages of HDAs (that I cannot do with other models)?

Nice people

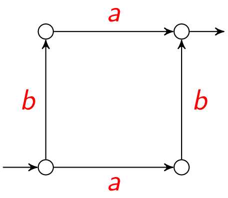
- Christian Johansen, NTNU
- Georg Struth, Sheffield
- Krzysztof Ziemiański, Warsaw

- Amazigh Amrane, Hugo Bazille, EPITA
- Safa Zouari, NTNU
- Eric Goubault, LIX

- See `https://ulifahrenberg.github.io/pomsetproject/` for more

Higher-dimensional automata

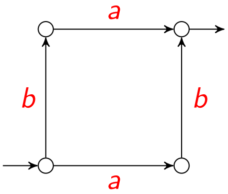
semantics of "*a* parallel *b*":



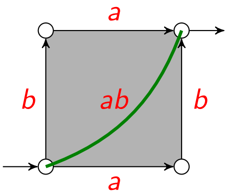
$a.b + b.a$

Higher-dimensional automata

semantics of "*a* parallel *b*":



$a.b + b.a$



a and *b* are independent

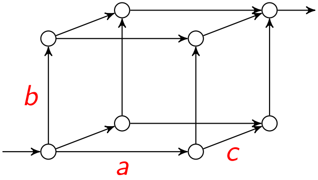
Higher-dimensional automata & concurrency

HDAs as a model for concurrency:

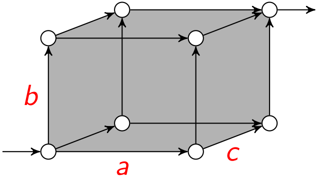
- points: states
- edges: transitions
- squares, cubes etc.: independency relations (concurrently executing events)
- two-dimensional automata $\cong$ asynchronous transition systems [Bednarczyk]

[van Glabbeek 2006, TCS]: Up to history-preserving bisimilarity, HDAs "generalize the main models of concurrency proposed in the literature" (notably, event structures and Petri nets)

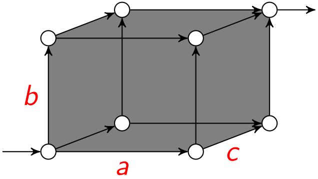
Examples



no concurrency



two out of three



full concurrency

Precubical sets and higher dimensional automata

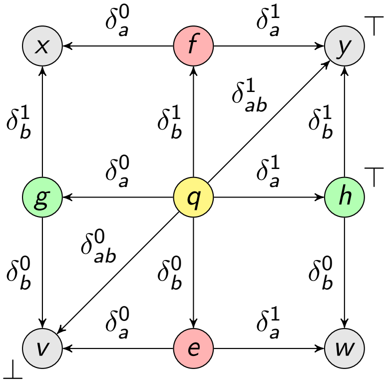
An loset is a finite, ordered and Σ-labelled set. (a list of events)

A precubical set X consists of:

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

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

Example



$X[\emptyset] = \{v, w, x, y\}$

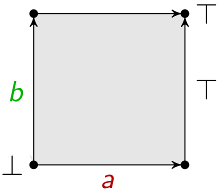
$X[a] = \{e, f\}$

$X[b] = \{g, h\}$

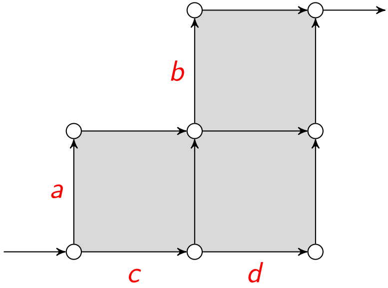
$X[ab] = \{q\}$

$\bot_X = \{v\}$

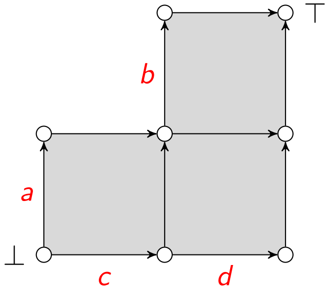
$\top_X = \{h, y\}$



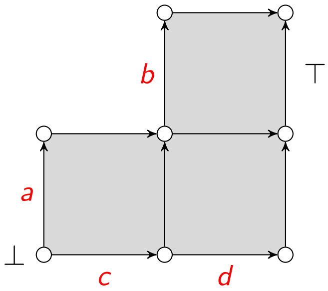
More interesting



More interesting



More interesting



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 (iso classes of) losets as objects.
Morphisms are coface maps $d_{A,B} : U \to V$, where

- $A, B \subseteq V$ are disjoint subsets,
- $U \simeq V - (A \cup B)$ are isomorphic losets,
- $d_{A,B} : U \to V$ is a unique order and label preserving map with image $V - (A \cup B)$.

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 $\partial : V \to W - (C \cup D)$ is the loset isomorphism.

Intuitively, $d_{A,B}$ terminates events B and “unstarts” events A.

- precubical sets: presheaves over $\square$

Context

augmented presimplex category Δ	large augmented presimplex category $\blacktriangle$
objects $\{1 < \cdots < n\}$ for $n \geq 0$	objects totally ordered sets
morphisms order injections	morphisms order injections
skeletal	isos are unique

$\Delta \hookrightarrow \blacktriangle$ equivalence with unique left inverse

(augmented) precube category $\square$	large (augmented) precube category $\boxdot$
objects $\{0,1\}^n$ for $n \geq 0$	objects totally ordered sets
morphisms 0-1 injections	morphisms distinguished order injections
skeletal	isos are unique

$\square \hookrightarrow \boxdot$ equivalence with unique left inverse

- presimplicial sets: $\mathsf{Set}^{\Delta^{\text{op}}}$ or $\mathsf{Set}^{\blacktriangle^{\text{op}}}$; makes no difference
- precubical sets: $\mathsf{Set}^{\square^{\text{op}}}$ or $\mathsf{Set}^{\boxdot^{\text{op}}}$; makes no difference

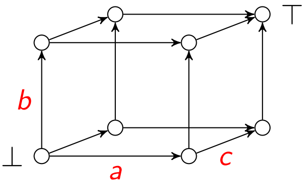
Languages of HDAs

- automata have languages
- HDAs don't (hitherto)
- (focus has been on geometric and topological aspects)

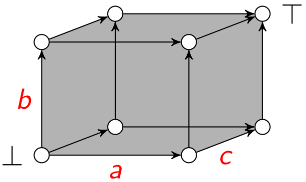
- automata and language theory is the very basis of computer science
- happy mix of operational and algebraic theory
- glue provided by Kleene and Myhill-Nerode theorems (among others)

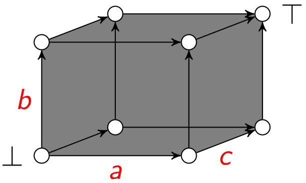
- Let's go!

Examples



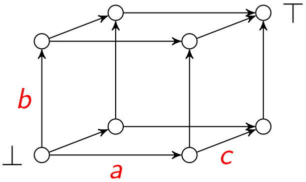
$$L_1 = \{abc, acb, bac, bca, cab, cba\}$$



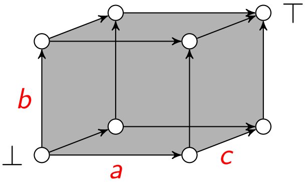


$$L_3 = \left\{ \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \dots \right\}$$

Examples

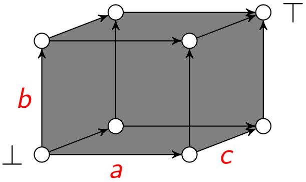


$$L_1 = \{abc, acb, bac, bca, cab, cba\}$$



$$L_2 = \Big\{ \begin{pmatrix} a \\ b \to c \end{pmatrix}, \begin{pmatrix} a \\ c \to b \end{pmatrix}, \begin{pmatrix} b \\ a \to c \end{pmatrix}, \begin{pmatrix} b \\ c \to a \end{pmatrix}, \begin{pmatrix} c \\ a \to b \end{pmatrix}, \begin{pmatrix} c \\ b \to a \end{pmatrix} \Big\} \cup L_1$$

sets of pomsets

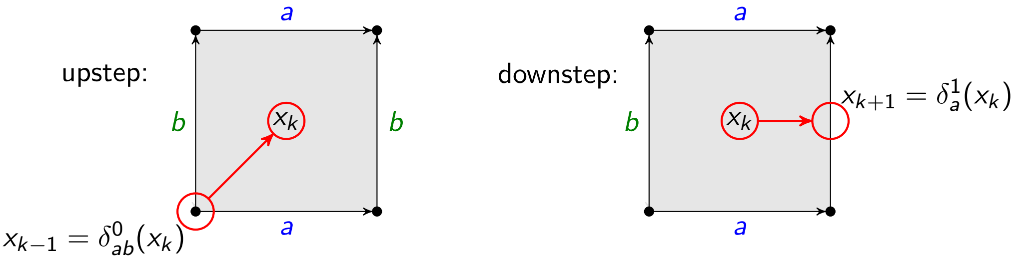


$$L_3 = \Big\{ \begin{pmatrix} a \\ b \\ c \end{pmatrix} \Big\} \cup L_2$$

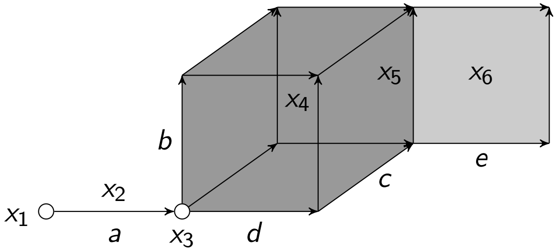
Computations of HDAs

A path on an HDA X is a sequence $(x_0, \phi_1, x_1, \dots, x_{n-1}, \phi_n, x_n)$ such that for every k, (x_{k-1}, ϕ_k, x_k) is either

- $(\delta^0_A(x_k), \nearrow^A, x_k)$ for $A \subseteq \mathrm{ev}(x_k)$ or (upstep: start A)
- $(x_{k-1}, \searrow_B, \delta^1_B(x_{k-1}))$ for $B \subseteq \mathrm{ev}(x_{k-1})$ (downstep: terminate B)

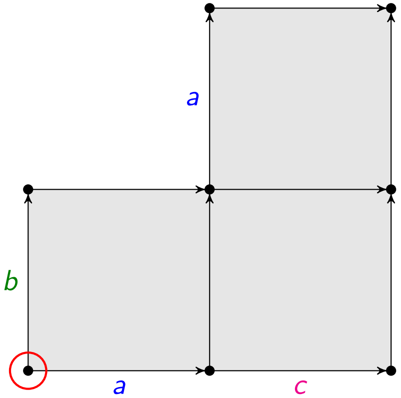


Example

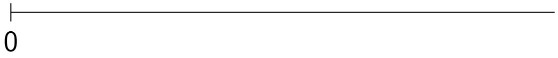


$$\left(x_1 \nearrow^{a} x_2 \searrow_{a} x_3 \nearrow^{\{b,c,d\}} x_4 \searrow_{\{c,d\}} x_5 \nearrow^{e} x_6\right)$$

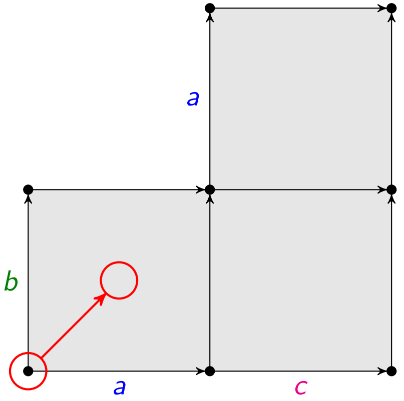
Event ipomset of a path



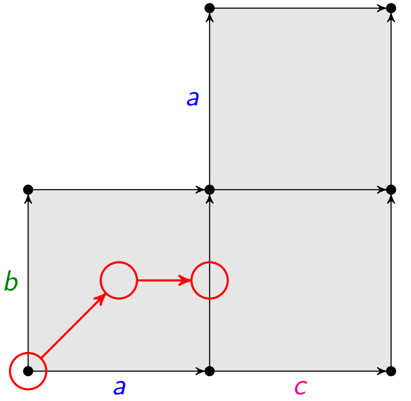
Lifetimes of events

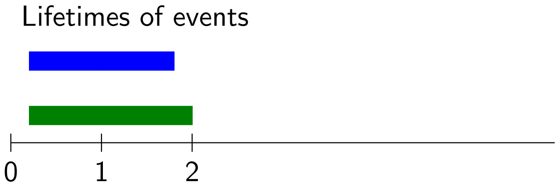


Event ipomset of a path

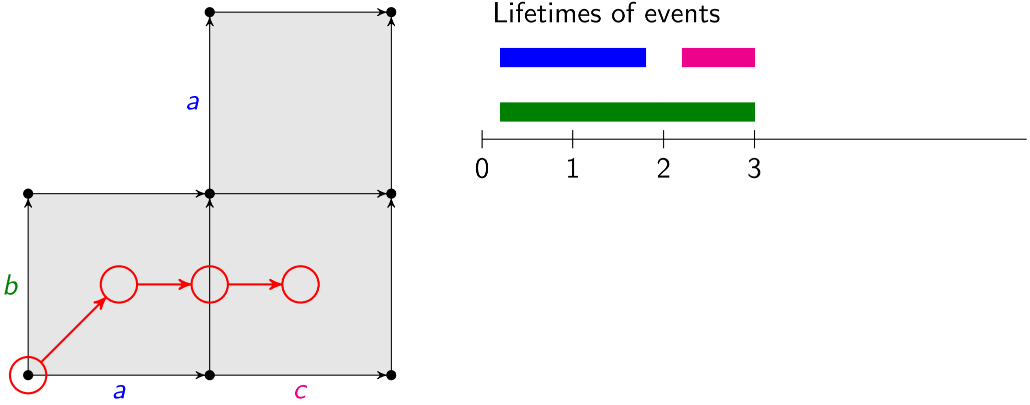


Event ipomset of a path

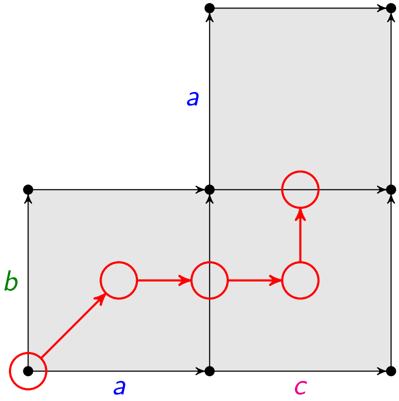




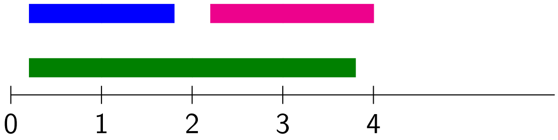
Event ipomset of a path



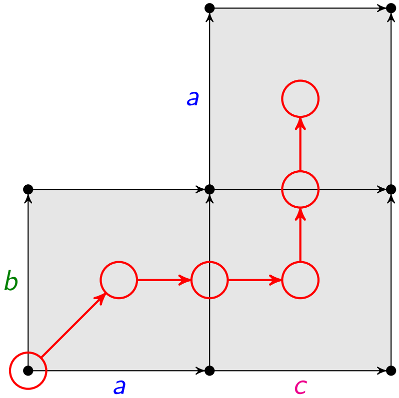
Event ipomset of a path

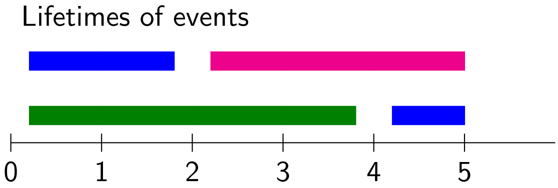


Lifetimes of events

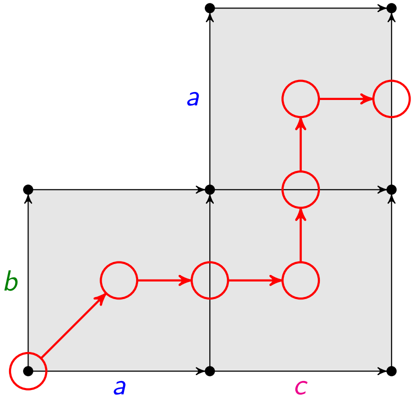


Event ipomset of a path

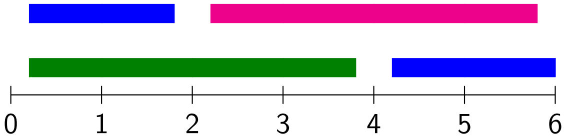




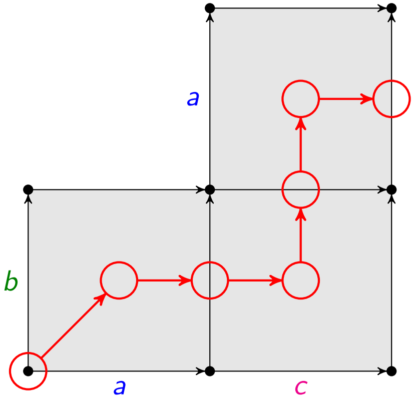
Event ipomset of a path



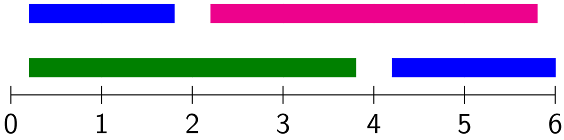
Lifetimes of events



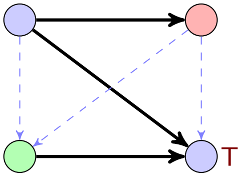
Event ipomset of a path



Lifetimes of events



Event ipomset

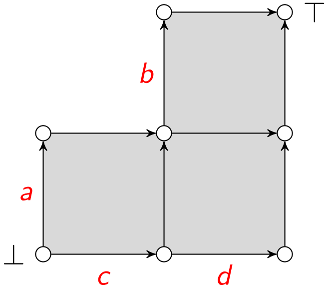


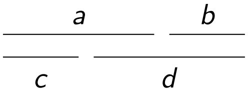
not series-parallel!

Are all pomsets generated by HDAs?

No, only (labeled) interval orders

- Poset $(P, \leq)$ is an interval order iff it has an interval representation:
  - a set $I = \{[l_i, r_i]\}$ of real intervals
  - with order $[l_i, r_i] \preceq [l_j, r_j]$ iff $r_i \leq l_j$
  - and an order isomorphism $(P, \leq) \leftrightarrow (I, \preceq)$
- [Fishburn 1970]





$$\begin{pmatrix} a \to b \\ c \to d \end{pmatrix}$$

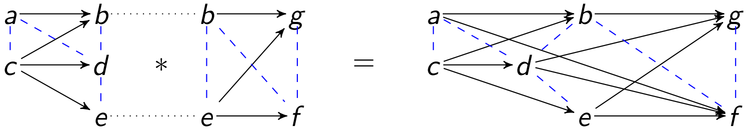
Pomsets with interfaces

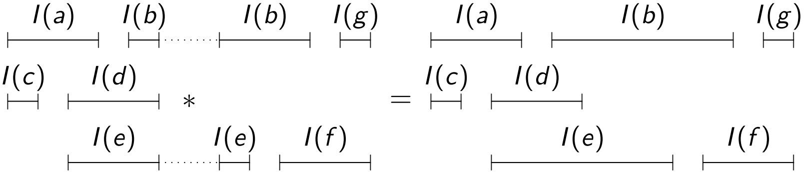
Definition (Ipomset)

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

- 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, T is $<$-maximal.

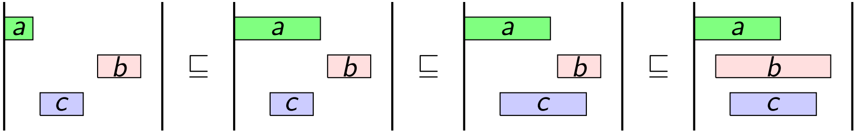
Composition of ipomsets

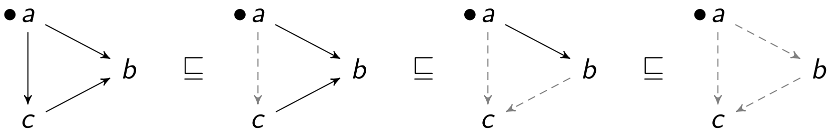




- Gluing $P * Q$: P before Q, except for interfaces (which are identified)
- Parallel composition $P \parallel Q$: P above Q (disjoint union)

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:

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

- $L(X)$ contains only interval-order ipomsets
- and is closed under subsumption

Path objects

Important tool:

Proposition

For any interval-order ipomset P there exists an HDA $\square^P$ for which $L(\square^P) = \{P\}\downarrow$.

Lemma

For any HDA X and ipomset P, $P \in L(X)$ iff $\exists f : \square^P \to X$.

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) $^+$

Theorem (à la Kleene)

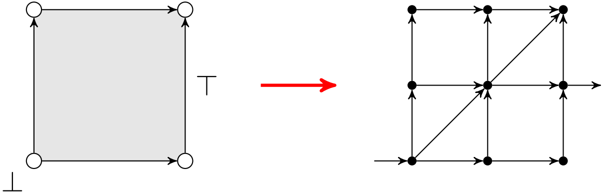
A language is rational iff it is recognized by an HDA.

Theorem (à la Myhill-Nerode)

A language is rational iff it has finite prefix quotient.

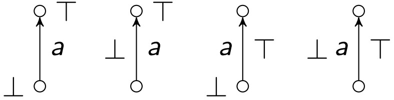
Kleene theorem: easy parts

- regular $\Rightarrow$ rational: by reduction to automata



- rational $\Rightarrow$ regular: generators:

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



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

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

Kleene theorem: gluing of HDAs

- 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
- uses inspiration from topology
- showing here: only gluing, no $^+$

Gluing composition: naive attempt

Assumptions:

- X, Y: HDAs,
- X, Y are simple, i.e., have one start and one accept cell each
- $\text{ev}(x^\top) = \text{ev}(y_\bot) =: U$.

The gluing composition of X and Y is the HDA

$$X * Y = \operatorname{colim}\left(X \xleftarrow{x^\top} \square^U \xrightarrow{y_\bot} Y \right)$$

(identifying the accept cell of X with the start cell of Y)
with $(X * Y)_\bot = X_\bot$, $(X * Y)^\top = Y^\top$.

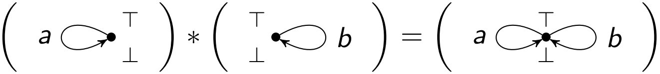
Lemma

$$L(X) * L(Y) \subseteq L(X * Y).$$

Gluing composition: problems

Do we have $L(X * Y) = L(X) * L(Y)$? No.

Problem 1:

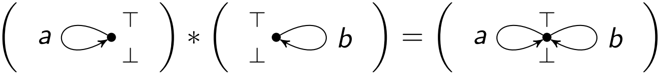


but $a^* * b^* \neq (a+b)^*$.

Gluing composition: problems

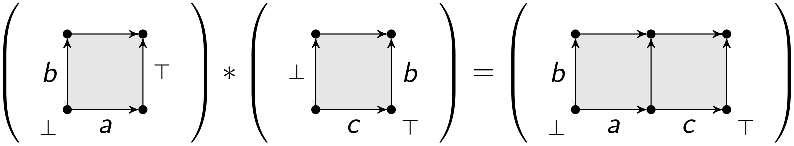
Do we have $L(X * Y) = L(X) * L(Y)$? No.

Problem 1:



but $a^* * b^* \neq (a+b)^*$.

Problem 2:



$\emptyset$ $\ni ac$

We need to prepare X and Y to avoid these problems

Tools: HDAs with interfaces

An loset with interfaces (iloset) is an loset U with subsets $S \subseteq U \supseteq T$ (notation: ${}_S U_T$).
(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:

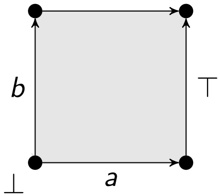
- Every cell $x \in X$ has an iloset $\mathrm{ev}(x)$
- We write $X[{}_S U_T] = \{x \in X \mid \mathrm{ev}(x) = {}_S U_T\}$.
- For every $A \subseteq U - S$ there is a lower face map $\delta^0_A : X[U] \to X[{}_S U_T - A]$.
- For every $B \subseteq U - T$ there is an upper face map $\delta^1_B : X[U] \to X[{}_S U_T - b]$.
- Precubical identities: $\delta^\mu_A \delta^\nu_B = \delta^\nu_B \delta^\mu_A$ for $A \cap B = \emptyset$ and $\mu, \nu \in \{0, 1\}$

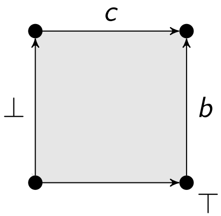
An HDA with interfaces (iHDA) is a finite ipc-set with start and accept cells.

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 $T = U$.

Example

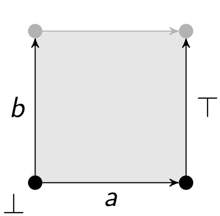
Slogan: iHDAs are consistent partial HDAs

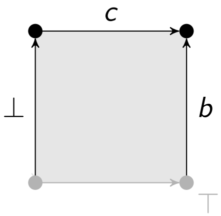




Example

Slogan: iHDAs are consistent partial HDAs





About iHDAs

- ipc-sets are presheaves over a category $\mathrm{I}\square$.
- paths on iHDAs and their event ipomsets are defined the same way as for HDAs
- There is a pair of adjoint functors

$$\mathsf{Res} : \mathsf{HDA} \to \mathsf{iHDA} \qquad \mathsf{Cl} : \mathsf{iHDA} \to \mathsf{HDA}.$$

- (induced by the geometric morphism $\mathsf{Res} : \mathsf{Set}^{\square^{\mathrm{op}}} \leftrightarrows \mathsf{Set}^{\mathrm{I}\square^{\mathrm{op}}} : \mathsf{Cl}$)

Lemma

For $X \in \mathsf{HDA}$ and $Y \in \mathsf{iHDA}$, $L(\mathsf{Res}(X)) = L(X)$ and $L(\mathsf{Cl}(Y)) = L(Y)$.

Theorem

Finite HDAs and finite iHDAs recognize the same class of languages.

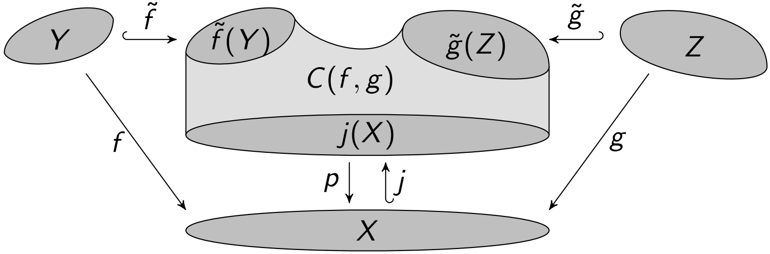
Preparing iHDAs

Definitions:

- an ipc-subset $Y \subseteq X$ is initial if no path in X may enter Y
- an ipc-subset $Y \subseteq X$ is final if no path in X may leave Y
- $f : Y \to X$ is an initial/final inclusion if it is injective and $f(Y) \subseteq X$ is initial/final.
- an iHDA X is start proper if the canonical ipc-map $\coprod_{x \in \bot_X} \mathrm{I}\square^{\mathrm{ev}(x)} \to X$ is an initial inclusion
- (all start cells are initial, disjoint and non-self-linked)
- (accept proper: defined similarly)

Cylinders

Let X, Y, Z be ipc-sets and $f : Y \to X$, $g : Z \to X$ ipc-maps such that $f(Y) \cap g(Z) = \emptyset$. 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 $C(f, g)$.

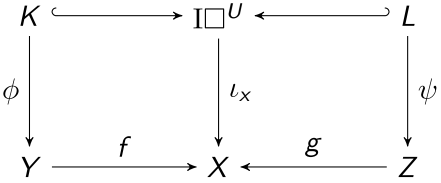
Cylinders: construction

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

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

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;
- $\phi : K \to Y$, $\psi : L \to Z$ are ipc-maps satisfying $f \circ \phi = \iota_x|_K$ and $g \circ \psi = \iota_x|_L$:



Proper iHDAs for simple languages

Definition:

- A language L is simple if it is recognized by an iHDA with one start and one accept cell.

Lemma

Any regular language is a finite union of simple languages.

Proposition

If L is simple, then

- *L is recognized by a start proper iHDA with one start cell.*
- *L is recognized by an accept proper iHDA with one accept cell.*

(usually, one cannot have both)

Theorem

If X is an accept proper iHDA with one accept cell, and Y is a start proper iHDA with one start cell, then $L(\mathsf{Cl}(X) * \mathsf{Cl}(Y)) = L(X) * L(Y)$.

Collecting the pieces

Theorem

Gluing compositions of regular languages are regular.

Proof: Let L and M be regular languages.

1. We may assume that L and M are simple, i.e., $L = L(X)$, $M = L(Y)$ for iHDAs X and Y having one start and one accept cell each.
2. We may replace X and Y by X' and Y', such that X' is accept proper, Y' is start proper, $L(X') = L(X)$, and $L(Y') = L(Y)$.
3. Go back to HDAs and glue:

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

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

Myhill-Nerode

Prefix quotients:

- $P \backslash L := \{Q \in \text{iiPoms} \mid PQ \in L\}$
- $\text{suff}(L) := \{P \backslash L \mid P \in \text{iiPoms}\}$

Theorem

L is rational iff suff(L) is finite.

Proof $\Rightarrow$**:** Let $L = L(X)$ be rational.

1. For $x \in X$ denote $\text{Pre}(x) = L(X_\perp^x)$ and $\text{Post}(x) = L(X_x^\top)$.
2. Lemma: for all P, $P \backslash L = \bigcup \{\text{Post}(x) \mid x \in X, P \in \text{Pre}(x)\}$.
3. And then $\{P \backslash L \mid P \in \text{iiPoms}\} \subseteq \{\bigcup_{x \in Y} \text{Post}(x) \bigm| Y \subseteq X\}$ which is finite. $\square$

Myhill-Nerode $\Longleftarrow$

Assume suff(L) finite. Construct HDA $M(L)$:

- Write $P \sim_L Q$ if $P\backslash L = Q\backslash L$
  - standard Myhill-Nerode equivalence: doesn't work for us
  - but implies $S_P = S_Q$ and $T_P = T_Q$
- Write $P \approx_L Q$ if $P \sim_L Q$ and $\forall A \subseteq T_P - S_P : (P - A)\backslash L = (Q - A)\backslash L$
- cells of $M(L)$: $M(L)[U] = \text{iiPoms}_U / \approx_L \cup \{w_U\}$ $\longleftarrow$ subsidiary "completion" cells
- face maps:
  - $\delta^1_A(\langle P\rangle) = \langle P * U{\downarrow_A}\rangle$ (terminate A)
  - $\delta^0_A(\langle P\rangle) = \langle P - A\rangle$ if $A \subseteq T_P - S_P$ (unstart A)
  - $\delta^0_A(\langle P\rangle) = w_{U-A}$ otherwise; $\delta^0_A(w_U) = \delta^1_A(w_U) = w_{U-A}$
- $\bot_{M(L)} = \{\langle \text{id}_U\rangle\}_{U\in\Box}$ and $\top_{M(L)} = \{\langle P\rangle \mid P \in L\}$

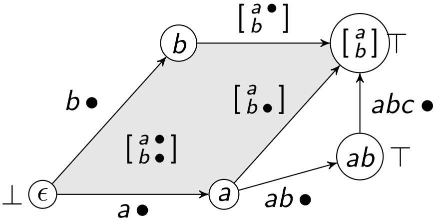
Proposition

The essential part of $M(L)$ is finite and $L(M(L)) = L$.

- essential part: reachable and co-reachable cells plus all their faces

Example

$$L = \{\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right], ab, ba, abc\}$$



$M(L)[\emptyset]$

P	$P\backslash L$
ϵ	L
a	$\{b, bc\}$
b	$\{a\}$
ab	$\{\epsilon, c\}$
$\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$	$\{\epsilon\}$

$M(L)[a]$

P	$P\backslash L$
$a\bullet$	$\{\left[\begin{smallmatrix} \bullet a \\ b \end{smallmatrix}\right], \bullet ab, \bullet abc\}$
$ba\bullet$	$\{\bullet a\}$

$M(L)[b]$

P	$P\backslash L$
$b\bullet$	$\{\left[\begin{smallmatrix} a \\ \bullet b \end{smallmatrix}\right], \bullet ba\}$
$ab\bullet$	$\{\bullet b, \bullet bc\}$
$\left[\begin{smallmatrix} a \\ b\bullet \end{smallmatrix}\right]$	$\{\bullet b\}$

$M(L)[c]$

P	$P\backslash L$
$abc\bullet$	$\{\bullet c\}$

$M(L)[\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]]$

P	$P\backslash L$
$\left[\begin{smallmatrix} a\bullet \\ b\bullet \end{smallmatrix}\right]$	$\{\left[\begin{smallmatrix} \bullet a \\ \bullet b \end{smallmatrix}\right]\}$

Properties

- $M(L)$ may be non-deterministic
- if L is determinizable, then $M(L)$ is deterministic (and minimal (?))
- but there exist non-determinizable ipomset languages
- in fact, there are languages of unbounded ambiguity
  - for example $L = ([\,^a_b\,]\, cd + ab\, [\,^c_d\,])^+$

Further:

- regular languages are closed under ($\cup$, $*$, $\|$, $^+$, and) $\cap$
- but not under complement
  - L regular $\Rightarrow$ L has finite width $\Rightarrow$ $(\text{iiPoms} - L)\!\downarrow$ has infinite width
- width-bounded complement: $\overline{L}^k = \{P \in \text{iiPoms} - L \mid \text{wid}(P) \le k\}\!\downarrow$
- regular languages are closed under $\overline{}^k$ (for all k)

Further:

- emptiness and inclusion of regular languages are decidable

Conclusion & Further Work

Higher-Dimensional Automata Theory for Fun and Profit!

- Kleene and Myhill-Nerode: a good start
- are HDAs learnable?
- trouble with determinization and non-ambiguity: history-determinism to the rescue?

- logical characterization? Büchi-Elgot theorem?
- relation to trace theory?
- languages vs homotopy?
- presheaf automata?
- coalgebra?

- higher-dimensional timed automata
- Distributed Hybrid Systems

Merci à Krzysztof pour des planches et à Amazigh pour les relire