Bisimulation for Higher-Dimensional Automata

A Geometric Interpretation

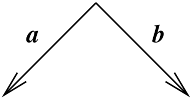
Uli Fahrenberg

Department of Mathematical Sciences
Aalborg University

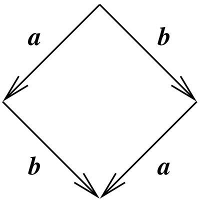
Fields-Ottawa Workshop on the Geometry of Very Large Data Sets

Outline

Parallelism vs. Mutual Exclusion



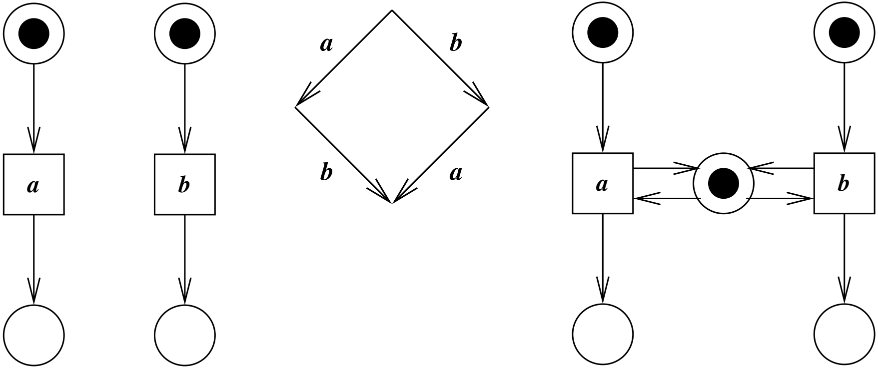
Parallelism vs. Mutual Exclusion



Parallelism vs. Mutual Exclusion

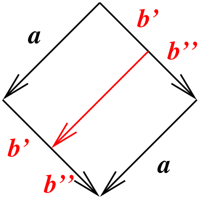
$a||b$

$a.b + b.a$



Parallelism vs. Mutual Exclusion

$a||b$



Action refinement

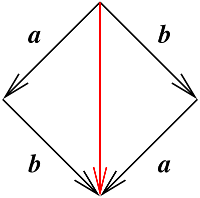
$a.b + b.a$

Parallelism vs. Mutual Exclusion

$a||b$

$a.b + b.a$

$D(a||b) =$
$\max(D(a), D(b))$

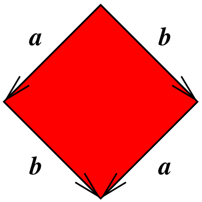


$D(a.b + b.a) =$
$D(a) + D(b)$

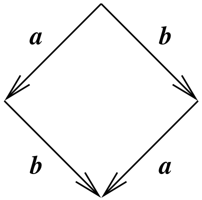
Real-time systems

Parallelism vs. Mutual Exclusion

$a||b$



$a.b + b.a$



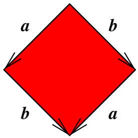
Higher-Dimensional Automata

So a higher-dimensional automaton is a pointed precubical set

$$A = \{A_n\}$$
$$\delta_i^0, \delta_i^1 : A_n \to A_{n-1} \quad (i = 1, \dots, n)$$

(The point $* \in A_0$ is the *initial state*.)

Note: For *labeled* HDA, the easiest is to work in a comma category of pointed precubical sets over a category of certain special *alphabet precubical sets* (which are ∞-tori), but we shall not need this here.



(Pratt, Van Glabbeek 1991)

Higher-Dimensional Automata

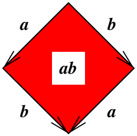
So a higher-dimensional automaton is a pointed precubical set

$$A = \{A_n\}$$

$$\delta_i^0, \delta_i^1 : A_n \to A_{n-1} \quad (i = 1, \dots, n)$$

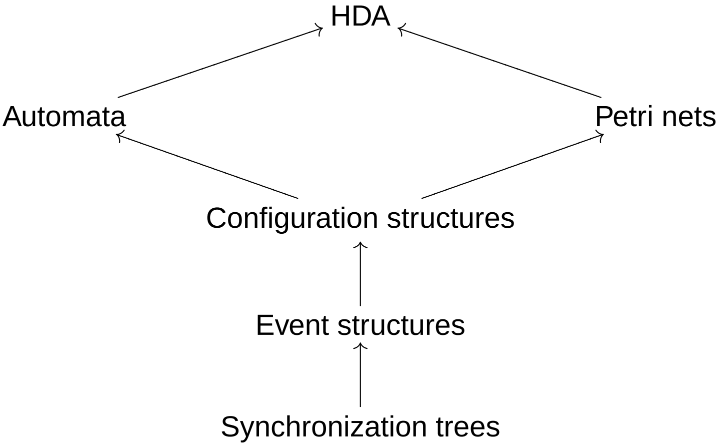
(The point $* \in A_0$ is the *initial state*.)

Note: For *labeled* HDA, the easiest is to work in a comma category of pointed precubical sets over a category of certain special *alphabet precubical sets* (which are ∞-tori), but we shall not need this here.



(Pratt, Van Glabbeek 1991)

The "van Glabbeek Hierarchy"



The Link to Geometry

Geometric realisation:

$$\text{precubical sets} \longrightarrow \text{topological spaces}$$

Holes in the space $\Longleftrightarrow$ Mutual exclusion in the HDA

Has been employed by various people, Goubault, Fajstrup, Raussen, . . .

My contribution: Bisimilarity of HDA is related to a path-lifting property of their geometric realisations

Morphisms of HDA

Morphisms of HDA should be *simulations*:
$A \to B$ iff whatever A can compute, B can compute, too.

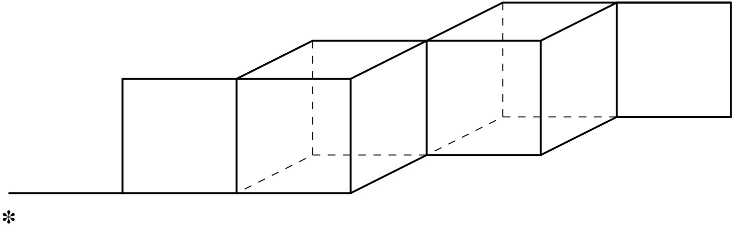
So what is a computation?

So simulations are just *morphisms of pointed precubical sets*.

Morphisms of HDA

Morphisms of HDA should be *simulations*:
$A \to B$ iff whatever A can compute, B can compute, too.

So what is a computation?

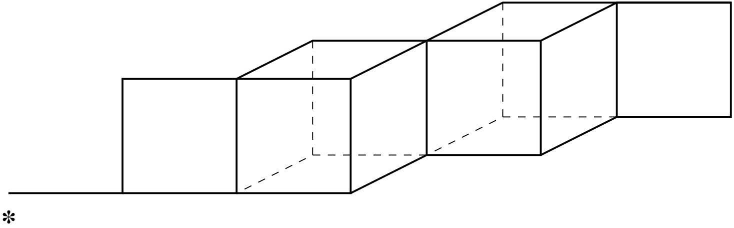


So simulations are just *morphisms of pointed precubical sets*.

Morphisms of HDA

Morphisms of HDA should be *simulations*:
$A \to B$ iff whatever A can compute, B can compute, too.

So what is a computation?



So simulations are just *morphisms of pointed precubical sets*.

Morphisms of HDA

Fine, so the category of HDA (over a fixed alphabet L) is just the category of pointed precubical sets.

Note: For the category of HDA over varying alphabets, things are more complicated. One needs to introduce "*idle transitions*" and to work with *cubical* sets instead (i.e. precubical sets with degeneracies).

To be precise: The full category of HDA consists of diagrams like this one, with A, A', L, L' precubical sets, black arrows precubical morphisms, and red arrows *cubical* morphisms.

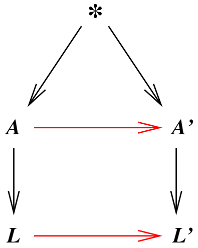
We will not need this here.

Morphisms of HDA

Fine, so the category of HDA (over a fixed alphabet L) is just the category of pointed precubical sets.

Note: For the category of HDA over varying alphabets, things are more complicated. One needs to introduce "*idle transitions*" and to work with *cubical* sets instead (i.e. precubical sets with degeneracies).

To be precise: The full category of HDA consists of diagrams like this one, with A, A', L, L' precubical sets, black arrows precubical morphisms, and red arrows *cubical* morphisms.



We will not need this here.

Bisimulation

Two HDA A, B are *bisimilar* if whatever A can compute, B can also compute, and *vice versa*.

So a morphism $f : A \to B$ is a bisimulation if for any $a \in A$ and for any computation starting in $f(a)$, there is a computation starting in a which maps to the computation in B.

Or equivalently, if

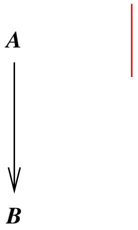
$$\forall\, a \in A, \forall\, c' \in B : f(a) = \delta_i^0 c',$$
$$\exists\, c \in A : c' = f(c), a = \delta_i^0 c$$

And two HDA B, C are bisimilar if there are bisimulations $B \leftarrow A \to C$. ("Bisimulation through open maps," Winskel, Nielsen 1995)

Bisimulation

Two HDA A, B are *bisimilar* if whatever A can compute, B can also compute, and *vice versa*.

So a morphism $f : A \to B$ is a bisimulation if for any $a \in A$ and for any computation starting in $f(a)$, there is a computation starting in a which maps to the computation in B.



Or equivalently, if

$$\forall\, a \in A, \forall\, c' \in B : f(a) = \delta_i^0 c',$$
$$\exists\, c \in A : c' = f(c), a = \delta_i^0 c$$

And two HDA B, C are bisimilar if there are bisimulations $B \leftarrow A \to C$. ("Bisimulation through open maps," Winskel, Nielsen 1995)

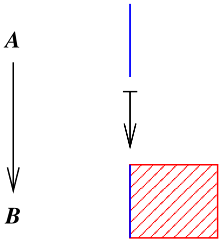
Bisimulation

Two HDA A, B are *bisimilar* if whatever A can compute, B can also compute, and *vice versa*.

So a morphism $f : A \to B$ is a bisimulation if for any $a \in A$ and for any computation starting in $f(a)$, there is a computation starting in a which maps to the computation in B.

Or equivalently, if

$$\forall\, a \in A, \forall\, c' \in B : f(a) = \delta_i^0 c',$$
$$\exists\, c \in A : c' = f(c), a = \delta_i^0 c$$



And two HDA B, C are bisimilar if there are bisimulations $B \leftarrow A \to C$. ("Bisimulation through open maps," Winskel, Nielsen 1995)

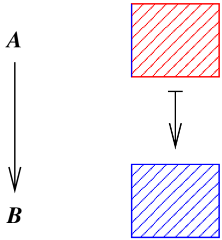
Bisimulation

Two HDA A, B are *bisimilar* if whatever A can compute, B can also compute, and *vice versa*.

So a morphism $f : A \to B$ is a bisimulation if for any $a \in A$ and for any computation starting in $f(a)$, there is a computation starting in a which maps to the computation in B.

Or equivalently, if

$$\forall a \in A, \forall c' \in B : f(a) = \delta_i^0 c',$$
$$\exists c \in A : c' = f(c), a = \delta_i^0 c$$

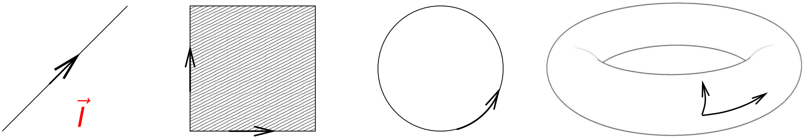


And two HDA B, C are bisimilar if there are bisimulations $B \leftarrow A \to C$. ("Bisimulation through open maps," Winskel, Nielsen 1995)

Local po-spaces

The geometric realisation of a precubical set is a local po-space; a topological space with a relation $\leq$ which is

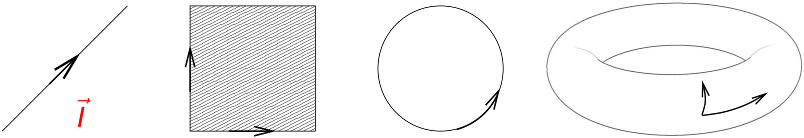
- reflexive,
- antisymmetric,
- *locally* transitive, and *locally* closed.
  (i.e. we have a cover $\mathcal{U} = \{U_\alpha\}$ of X such that $\leq$ is transitive and closed in each U_α.)



Local po-spaces

The geometric realisation of a precubical set is a local po-space; a topological space with a relation $\leq$ which is

- reflexive,
- antisymmetric,
- *locally* transitive, and *locally* closed.
  (i.e. we have a cover $\mathcal{U} = \{U_\alpha\}$ of X such that $\leq$ is transitive and closed in each U_α.)

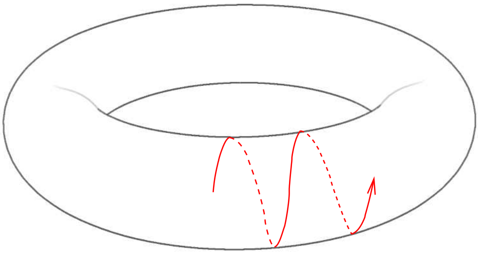


The Main Result

A dimap $f : X \to Y$ is a continuous mapping which is *locally increasing*:

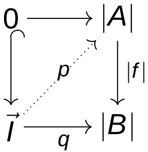
$$\forall\, x \in X, \exists\, U \ni x : \forall\, x_1 \leq x_2 \in U, f(x_1) \leq f(x_2) \in Y$$

A dipath in X is a dimap $\vec{I} \to X$.



The Main Result

Theorem: $f : A \to B$ is a bisimulation if and only if $|f| : |A| \to |B|$ has the dipath-lifting property



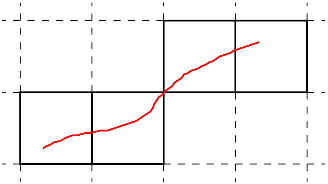
$$q = |f| \circ p$$

– connection to (directed) fibrations, obstruction theory

The Main Result

Theorem: $f : A \to B$ is a bisimulation if and only if $|f| : |A| \to |B|$ has the dipath-lifting property

Key of proof: For any dipath, there is a unique computation through (the geometric realisation of) which it "runs." [Fajstrup 2003]

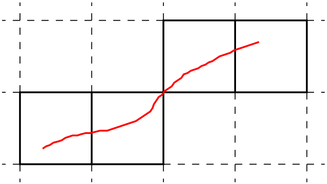


Only holds for geometric and locally finite HDA.

The Main Result

Theorem: $f : A \to B$ is a bisimulation if and only if $|f| : |A| \to |B|$ has the dipath-lifting property

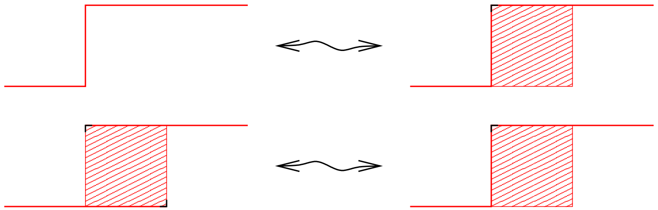
Key of proof: For any dipath, there is a unique computation through (the geometric realisation of) which it "runs." [Fajstrup 2003]



Only holds for geometric and locally finite HDA.

Equivalence of Computations

Adjacency of computations:

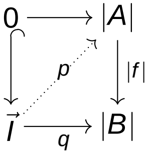


Equivalence of computations $:=$ equivalence generated by adjacency. [van Glabbeek 1991]

Bisimulation up to Equivalence

A morphism $f : A \to B$ is called a bisimulation up to equivalence if for any $a \in A$ and for any computation starting in $f(a)$, there is a computation starting in a which maps to a computation in B that is *equivalent* to the given one.

Conjecture: $f : A \to B$ is a bisimulation up to equivalence if and only $|f| : |A| \to |B|$ lifts dipaths *up to dihomotopy*:



$$q \simeq |f| \circ p$$

An Application to Topology

Conjecture: For any geometric, locally finite precubical set A, there exists a precubical set B such that $|B|$ is the universal (directed) covering space of $|A|$.

Key idea: The cubes of B are the equivalence classes of computations of A.

Thank You!

Uli Fahrenberg
Ph.D. student, Aalborg University, Denmark
(currently looking for job opportunities . . .)

uli@math.aau.dk
http://www.math.aau.dk/∼uli

Selected Bibliography I

- V. Pratt, *Modeling Concurrency with Geometry*. Proc. 18th ACM Symposium on Principles of Programming Languages, 1991.
- R. van Glabbeek, *Bisimulations for Higher Dimensional Automata*. Email message, 1991.
- R. van Glabbeek, *On the Expressiveness of Higher Dimensional Automata*. Proc. EXPRESS 2004, to be published.
- E. Goubault, *The Geometry of Concurrency*. Ph.D. thesis, 1995.
- L. Fajstrup, E. Goubault, M. Raussen, *Algebraic Topology and Concurrency*, 1999. Accepted for publication in Theor.Comp.Sci.

Selected Bibliography II

- L. Fajstrup, *Dipaths and Dihomotopies in a Cubical Complex*. Adv.Appl.Math., to appear.
- U. Fahrenberg, *A Category of Higher-Dimensional Automata*. Proc. FOSSACS 2005, to appear.

Advertisement

Geometry and Topology in Concurrency 2005
San Francisco
21 August 2005

Workshop on Algebraic Topology in Concurrency
Aalborg, Denmark
August 2005 (planned)