Models of directed algebraic topology

Jean Goubault-Larrecq

ENS Cachan

Georgia Southern U. — Jan. 21, 2014

Outline

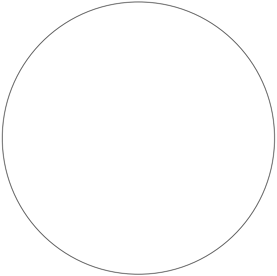
1 Directed algebraic topology
- Space, plus time
- Concurrent processes

2 Prestreams
- Prestreams
- Quotients, colimits, glueing
- Geometric realization

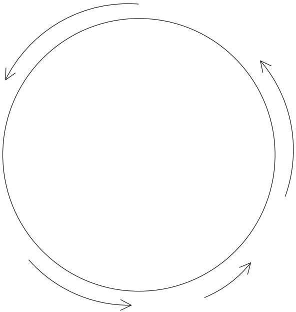
3 Streams
- The issue with prestreams
- Streams
- Stream products
- Geometric realization

4 Conclusion

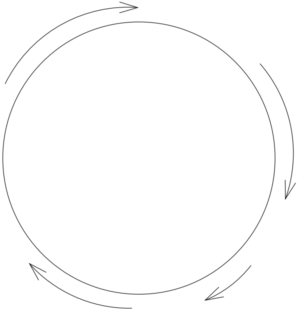
What is this?



What is this?



What is this?



Directed Algebraic Topology

- Originates (mostly) from computer science questions on *concurrent processes* [E. Goubault 95]
- . . . interpreted geometrically [E. Dijkstra 68]
- . . . also in general relativity (R. Penrose, light cones)

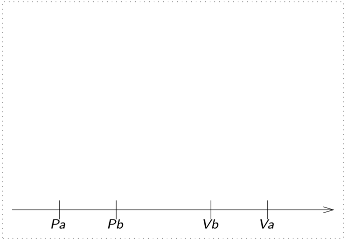
Object of study:

Topological space + notion of **time** (possibly cyclic)

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

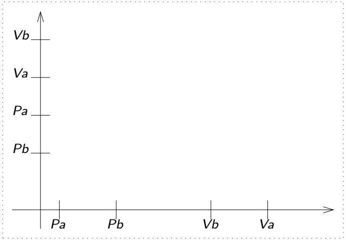


- Process 1: Pa; Pb; Vb; Va

PV Processes

Locks ensure access to resource by *single* process:

- *Pa* blocks until lock *a* taken
- *Va* releases lock *a*

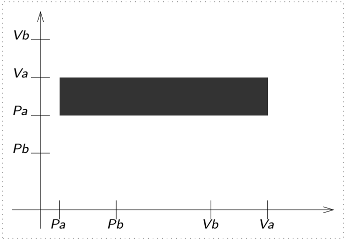


- Process 1: *Pa*; *Pb*; *Vb*; *Va*
- Process 2: *Pb*; *Pa*; *Va*; *Vb*

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

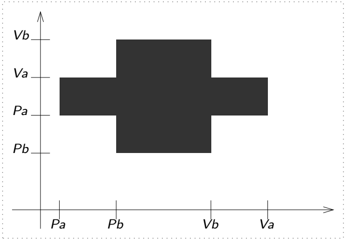


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- $P, V \Rightarrow$ forbidden rectangles

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

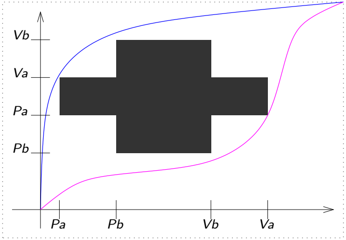


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- P, $V \Rightarrow$ forbidden rectangles

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

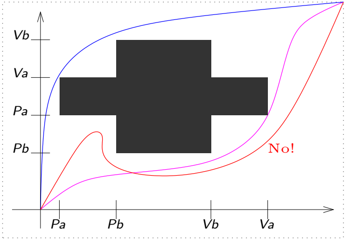


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- $P, V \Rightarrow$ forbidden rectangles
- *D*ipaths must respect order

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

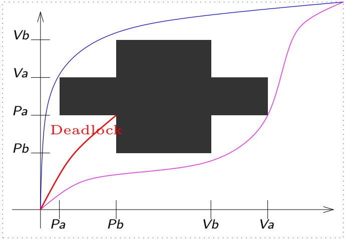


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- $P, V \Rightarrow$ forbidden rectangles
- *D*ipaths must respect order

PV Processes

Locks ensure access to resource by *single* process:

- *Pa* blocks until lock *a* taken
- *Va* releases lock *a*

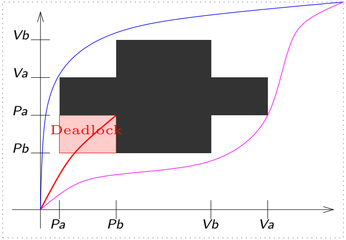


- Process 1: *Pa*; *Pb*; *Vb*; *Va*
- Process 2: *Pb*; *Pa*; *Va*; *Vb*
- *P*, *V* $\Rightarrow$ forbidden rectangles
- *D*ipaths must respect order

PV Processes

Locks ensure access to resource by *single* process:

- *Pa* blocks until lock *a* taken
- *Va* releases lock *a*

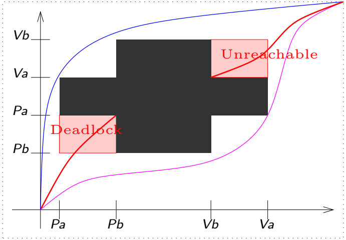


- Process 1: *Pa*; *Pb*; *Vb*; *Va*
- Process 2: *Pb*; *Pa*; *Va*; *Vb*
- *P*, *V* $\Rightarrow$ forbidden rectangles
- *D*ipaths must respect order

PV Processes

Locks ensure access to resource by *single* process:

- *Pa* blocks until lock *a* taken
- *Va* releases lock *a*

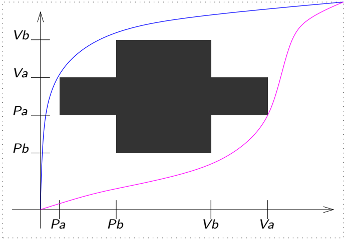


- Process 1: *Pa*; *Pb*; *Vb*; *Va*
- Process 2: *Pb*; *Pa*; *Va*; *Vb*
- *P*, *V* $\Rightarrow$ forbidden rectangles
- *D*ipaths must respect order

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

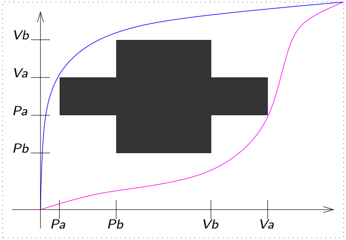


- Process 1: Pa; Pb; Vb; Va
- Process 2: Pb; Pa; Va; Vb
- P, $V \Rightarrow$ forbidden rectangles
- *D*ipaths must respect order
- *D*ihomotopies of *d*ipaths + fixed endpoints

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

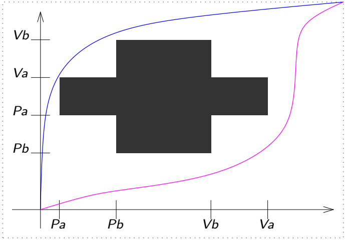


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- $P, V \Rightarrow$ forbidden rectangles
- *Di*paths must respect order
- *Di*homotopies of *di*paths + fixed endpoints

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a

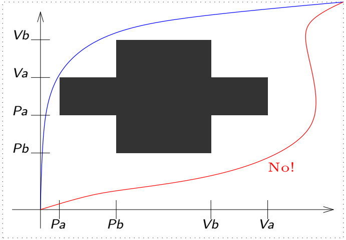


- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- P, $V \Rightarrow$ forbidden rectangles
- *D*ipaths must respect order
- *Di*homotopies of *di*paths + fixed endpoints

PV Processes

Locks ensure access to resource by *single* process:

- Pa blocks until lock a taken
- Va releases lock a



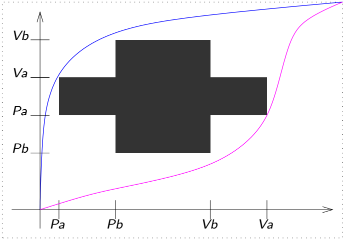
- Process 1: $Pa; Pb; Vb; Va$
- Process 2: $Pb; Pa; Va; Vb$
- P, $V \Rightarrow$ forbidden rectangles
- *D*ipaths must respect order
- *D*ihomotopies of *d*ipaths + fixed endpoints

Dihomotopy $\neq$ Homotopy

Definition

A **di**homotopy $h\colon [0,1]^2 \to X$ from f to g:

- h continuous
- $h(0, \_) = f$, $h(1, \_) = g$
- For every t, $h(t, \_)$ **di**path



Note: deformations $h(\_, u)$ *not* required to preserve order.

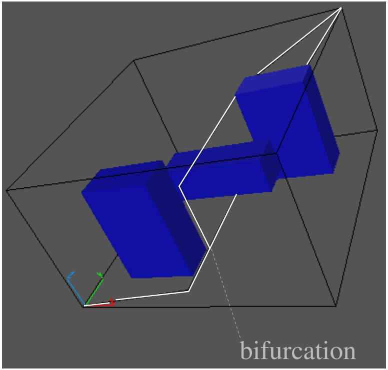
Dihomotopy $\neq$ Homotopy

Definition

A **di**homotopy $h\colon [0,1]^2 \to X$ from f to g:

- h continuous
- $h(0,\_) = f$, $h(1,\_) = g$
- For every t, $h(t,\_)$ **di**path

Note: deformations $h(\_, u)$ *not* required to preserve order.



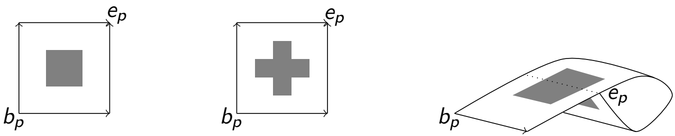
What is a geometric model of time+space?

- Ordered topological spaces. . .

What is a geometric model of time+space?

- Ordered topological spaces. . . do not handle cycles

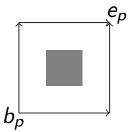
$P_a.V_a|P_a.V_a$ $\quad$ $P_a.P_b.V_b.V_a|P_b.P_a.V_a.V_b$ $\quad$ $P_a.(V_a.P_a)^*|P_a.V_a$

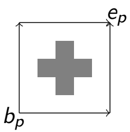


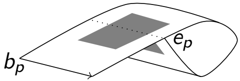
What is a geometric model of time+space?

- Ordered topological spaces. . . do not handle cycles

$P_a.V_a|P_a.V_a$ $\quad$ $P_a.P_b.V_b.V_a|P_b.P_a.V_a.V_b$ $\quad$ $P_a.(V_a.P_a)^*|P_a.V_a$





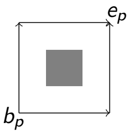


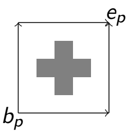
- Local pospaces (manifold-like) [Fajstrup, Goubault, Raussen 08] . . . do not admit colimits [Haucourt 04]
    - Bad: cannot define meaningful *geometric realization* functor

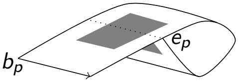
What is a geometric model of time+space?

- Ordered topological spaces. . . do not handle cycles

$P_a.V_a|P_a.V_a$ $\quad$ $P_a.P_b.V_b.V_a|P_b.P_a.V_a.V_b$ $\quad$ $P_a.(V_a.P_a)^*|P_a.V_a$







- Local pospaces (manifold-like) [Fajstrup, Goubault, Raussen 08] . . . do not admit colimits [Haucourt 04]
  - Bad: cannot define meaningful *geometric realization* functor
- Better:
  - d-spaces [Grandis 09]
  - **streams** [Krishnan 08]

Outline

1 Directed algebraic topology
- Space, plus time
- Concurrent processes

2 Prestreams
- Prestreams
- Quotients, colimits, glueing
- Geometric realization

3 Streams
- The issue with prestreams
- Streams
- Stream products
- Geometric realization

4 Conclusion

Prestreams

A prestream is a topological presheaf of preorders:

Definition (Category Prestr of prestreams)

A *prestream* $\mathcal{X}$ = topological space X
+ *precirculation* $(\sqsubseteq_U)_{U \in \mathcal{O}(X)}$:

- $\sqsubseteq_U$ preorder on U
- monotonicity: $U \subseteq V, x \sqsubseteq_U y \Rightarrow x \sqsubseteq_V y$

Prestreams

A prestream is a topological presheaf of preorders:

Definition (Category Prestr of prestreams)

A *prestream* $\mathcal{X}$ = topological space X

+ *precirculation* $(\sqsubseteq_U)_{U \in \mathcal{O}(X)}$:

- $\sqsubseteq_U$ preorder on U
- monotonicity: $U \subseteq V, x \sqsubseteq_U y \Rightarrow x \sqsubseteq_V y$

Prestream *morphisms* $f : (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)}) \to (Y, (\preceq_V)_{V \in \mathcal{O}(Y)})$:

- continuous
- locally monotonic: for each open V of Y,

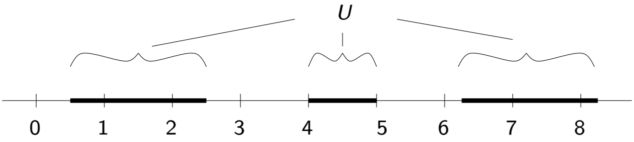
$$x \sqsubseteq_{f^{-1}(V)} y \Rightarrow f(x) \preceq_V f(y)$$

Paradigmatic examples 1

- *Preordered spaces*: $\sqsubseteq_U = \leq_{|U}$ for each U

Paradigmatic examples 1

- *Preordered spaces*: $\sqsubseteq_U = \leq_{|U}$ for each U
- $\overrightarrow{\mathbb{R}}$: $t \sqsubseteq_U^{\mathbb{R}} t'$ iff *whole interval* $[t, t'] \subseteq U$.



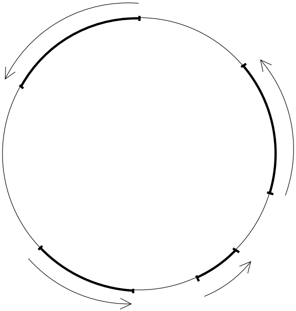
Here, $1 \sqsubseteq_U 2$, but $2 \not\sqsubseteq_U 4.6$
"Islands of order"

Paradigmatic examples 2: the directed circle

Definition

$\overrightarrow{S}^1 = \overrightarrow{\mathbb{R}}/\mathbb{Z}$

- Quotient by $x \equiv y$ iff $x - y \in \mathbb{Z}$
- **Prestr** is complete and cocomplete, in particular has quotients (see next slide)

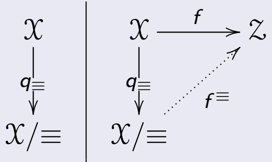


Prestream quotients

Theorem

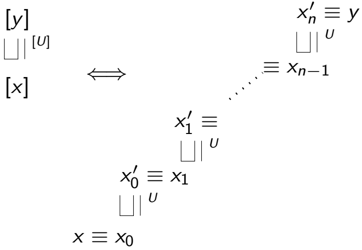
Let $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$,
$\equiv$ *equivalence relation on* X.

There is a
quotient prestream $\mathcal{X}/\equiv$:



(if f *compatible with* $\equiv$*)*

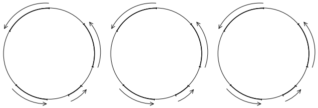
Proof. Build
topological quotient $X/\equiv$,
quotient map $q_\equiv(x) = [x]$,
and precirculation:



Colimits, glueing

Fact: coproducts $\coprod_{i \in I} \mathcal{X}_i$ exist.

Ex: coproduct of three copies of $\vec{S}^1$:



Theorem (Colimits)

Prestr *is cocomplete (has all colimits).*

Proof. (standard) Colimit of diagram $(\mathcal{X}_i \xrightarrow{f_{ij}} \mathcal{X}_j)_{i,j}$:

- Build big coproduct $\coprod_i \mathcal{X}_i$
- **Glue** every $x_i \in \mathcal{X}_i$ with $f_{ij}(x_i) \in \mathcal{X}_j$, i.e. build quotient by smallest equivalence relation $\equiv$ such that $x_i \equiv f_{ij}(x_i)$

Cubical sets

Cubical sets will replace simplicial sets in our directed world.

Definition (Cubical set)

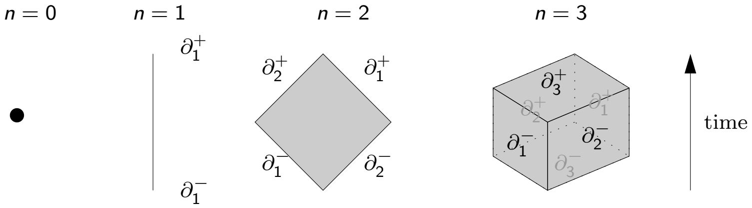
Collection of:

- sets K_n "n-dimensional cubes"
- *forward* ∂_i^+ and *backward* ∂_i^- face maps : $K_n \to K_{n-1}$
- degeneracies $s_i : K_{n-1} \to K_n$

satisfying:

$$\begin{array}{rcll} \partial_i^\alpha \partial_j^\beta & = & \partial_{j-1}^\beta \partial_i^\alpha & (i < j) \\ s_i \partial_j^\beta & = & \partial_{j+1}^\beta s_i & (i \leq j) \end{array} \qquad \partial_i^\alpha s_j = \begin{cases} s_{j-1} \partial_i^\alpha & (i < j) \\ \mathrm{id} & (i = j) \\ s_j \partial_{i-1}^\alpha & (i > j) \end{cases}$$

Geometric intuition

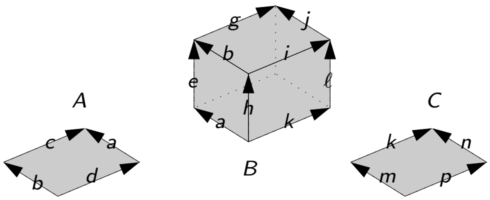


Geometric realization

Definition (Geometric realization functor Geom: Cub → Prestr)

$\mathbf{Geom}((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n) = (\coprod_n K_n \cdot [0,1]^n)/\equiv$ where:

$$\begin{array}{rcl} (a, (x_1, \cdots, 1, \cdots, x_n)) & \equiv & (\partial_i^+ a, (x_1, \cdots, x_n)) \\ (a, (x_1, \cdots, 0, \cdots, x_n)) & \equiv & (\partial_i^- a, (x_1, \cdots, x_n)) \\ (a, (x_1, \cdots, \hat{x}_i, \cdots x_n)) & \equiv & (s_i a, (x_1, \cdots, x_{n+1})) \end{array}$$

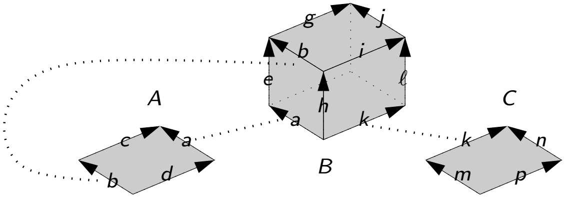


Geometric realization

Definition (Geometric realization functor $\mathbf{Geom}\colon \mathbf{Cub} \to \mathbf{Prestr}$)

$\mathbf{Geom}((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n) = (\coprod_n K_n \cdot [0,1]^n)/\equiv$ where:

$$
\begin{array}{rcl}
(a, (x_1, \cdots, 1, \cdots, x_n)) & \equiv & (\partial_i^+ a, (x_1, \cdots, x_n)) \\
(a, (x_1, \cdots, 0, \cdots, x_n)) & \equiv & (\partial_i^- a, (x_1, \cdots, x_n)) \\
(a, (x_1, \cdots, \hat{x}_i, \cdots x_n)) & \equiv & (s_i a, (x_1, \cdots, x_{n+1}))
\end{array}
$$

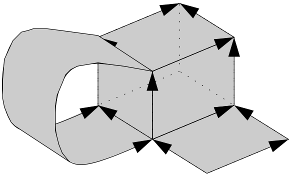


Geometric realization

Definition (Geometric realization functor **Geom**: **Cub** → **Prestr**)

$\mathbf{Geom}((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n) = (\coprod_n K_n \cdot [0,1]^n)/\equiv$ where:

$$\begin{aligned}(a,(x_1,\cdots,1,\cdots,x_n)) &\equiv (\partial_i^+ a,(x_1,\cdots,x_n))\\ (a,(x_1,\cdots,0,\cdots,x_n)) &\equiv (\partial_i^- a,(x_1,\cdots,x_n))\\ (a,(x_1,\cdots,\hat{x}_i,\cdots x_n)) &\equiv (s_i a,(x_1,\cdots,x_{n+1}))\end{aligned}$$



All is well

Conversely, there is a *singular cube* functor $\mathbf{Sing}\colon \mathbf{Prestr} \to \mathbf{Cub}$.
$\mathbf{Sing}(\mathcal{X}) = ((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n)$ where:

- $K_n = \{\text{prestream morphisms } \gamma\colon ([0,1]^n, \leq) \to \mathcal{X}\}$
- $\partial_i^+ \gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, 1, \cdots, x_n)$
- $\partial_i^- \gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, 0, \cdots, x_n)$
- $s_i \gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, \hat{x}_i, \cdots, x_n)$

Theorem

Geom *is left adjoint to* **Sing**

Outline

1 Directed algebraic topology
- Space, plus time
- Concurrent processes

2 Prestreams
- Prestreams
- Quotients, colimits, glueing
- Geometric realization

3 Streams
- The issue with prestreams
- Streams
- Stream products
- Geometric realization

4 Conclusion

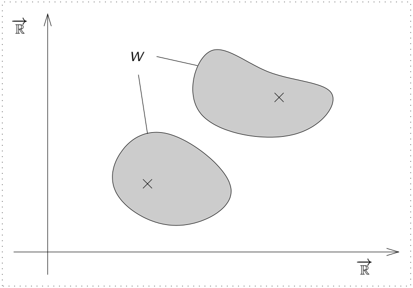
The issue with prestreams

Binary products in **Prestr** are *weird*.

Lemma

$(X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)}) \times (Y, (\preceq_V)_{V \in \mathcal{O}(Y)})$ *is* $X \times Y$ *with precirculation:*

$$(x, y) \leq_W (x', y') \text{ iff } x \sqsubseteq_{\pi_1[W]} x' \text{ and } y \preceq_{\pi_2[W]} y'$$



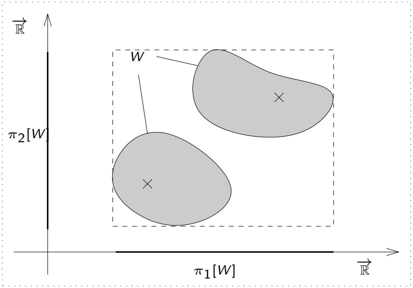
The issue with prestreams

Binary products in **Prestr** are *weird*.

Lemma

$(X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)}) \times (Y, (\preceq_V)_{V \in \mathcal{O}(Y)})$ is $X \times Y$ with precirculation:

$$(x, y) \leq_W (x', y') \text{ iff } x \sqsubseteq_{\pi_1[W]} x' \text{ and } y \preceq_{\pi_2[W]} y'$$



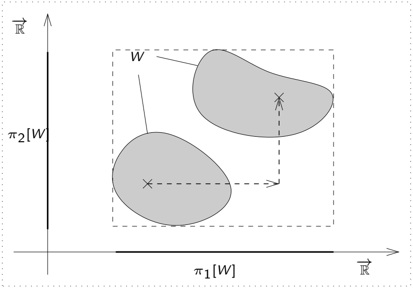
The issue with prestreams

Binary products in **Prestr** are *weird*.

Lemma

$(X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)}) \times (Y, (\preceq_V)_{V \in \mathcal{O}(Y)})$ is $X \times Y$ with precirculation:

$$(x, y) \leq_W (x', y') \text{ iff } x \sqsubseteq_{\pi_1[W]} x' \text{ and } y \preceq_{\pi_2[W]} y'$$

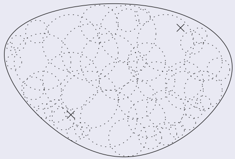


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathbin{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

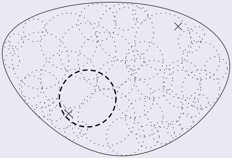


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

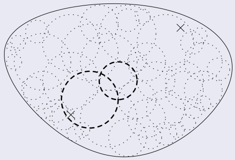


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \;\widehat{\sqsubseteq}_U\; y$ iff
for every open cover $(U_i)_{i \in I}$ of $U \ldots$

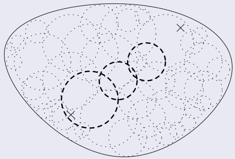


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

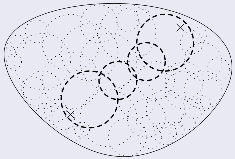


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

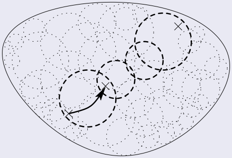


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

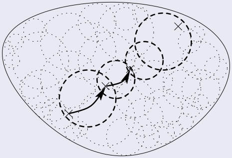


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

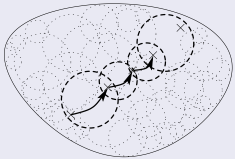


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U...

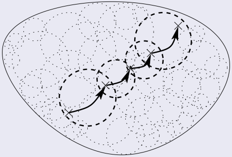


Streams

Definition (One-step cosheafification)

For $\mathcal{X} = (X, (\sqsubseteq_U)_{U \in \mathcal{O}(X)})$, $Sh^1(\mathcal{X}) = (X, (\widehat{\sqsubseteq}_U)_{U \in \mathcal{O}(X)})$ where:

$x \mathrel{\widehat{\sqsubseteq}_U} y$ iff
for every open cover $(U_i)_{i \in I}$ of U:



Note: $Sh^1(\mathcal{X})$ always finer than $\mathcal{X}$ on X.

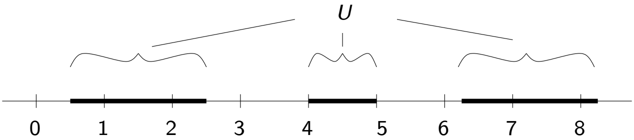
Definition (Stream)

$\mathcal{X}$ stream iff $Sh^1(\mathcal{X}) = \mathcal{X}$

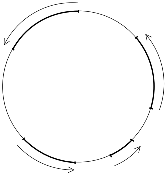
Defines a full subcategory **Str** of **Prestr**.

Paradigmatic examples

- *Not* preordered spaces in general, but their cosheafification OK (see next slide)
- $\overrightarrow{\mathbb{R}}$ = cosheafification of $(\mathbb{R}, \leq)$:



- $\overrightarrow{S}^1 = \overrightarrow{\mathbb{R}}/\mathbb{Z}$



Cosheafification

Make a stream out of a prestream $\mathcal{X}$: Refine

$$\mathcal{X} \to Sh^1(\mathcal{X}) \to Sh^1(Sh^1(\mathcal{X})) \to \cdots$$

Iterate Sh^1 transfinitely (!),
$\Rightarrow$ obtain $Sh^\infty(X)$, coarsest stream finer than $\mathcal{X}$.

Definition

$Sh^\infty(\mathcal{X})$ is the *cosheafification* of $\mathcal{X}$.

Yuck.

Str is nice (but complex?)

Theorem

Str *is cocomplete: has all colimits,*
and they are computed just as in **Prestr**.

Good!

Str is nice (but complex?)

Theorem

Str *is cocomplete: has all colimits,*
and they are computed just as in **Prestr***.*

Good!

Theorem

Str *is complete: has all limits. . .*
which are cosheafifications of the corresponding limits in **Prestr***.*

Yuck.

All this follows from: Sh^{∞} coreflects **Prestr** onto **Str**.

Understanding stream products

Still, some limits can be made simpler:

Proposition

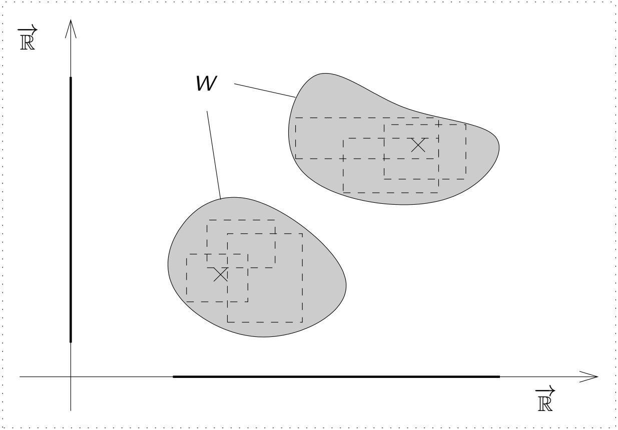
The product of two streams $\mathcal{X}$, $\mathcal{Y}$ is $Sh^1(\mathcal{X} \times \mathcal{Y})$

I.e., take prestream product
then *one-step* cosheafification (not Sh^∞).

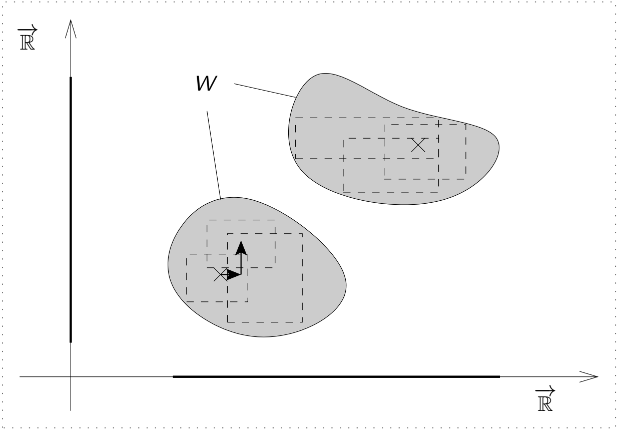
In fact, binary products are particularly nice (see next slide).

(explaining substreams [equalizers] seems almost hopeless)

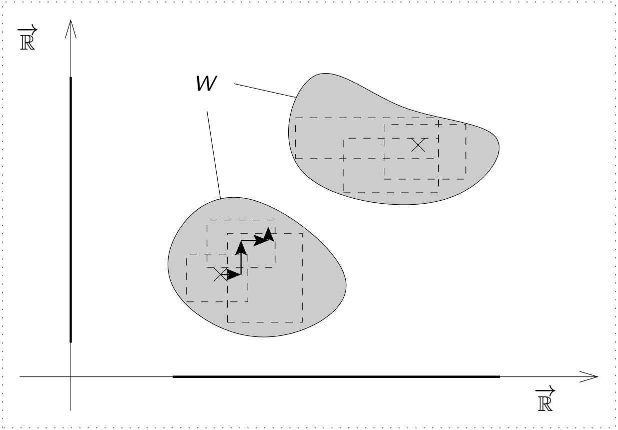
Stream products



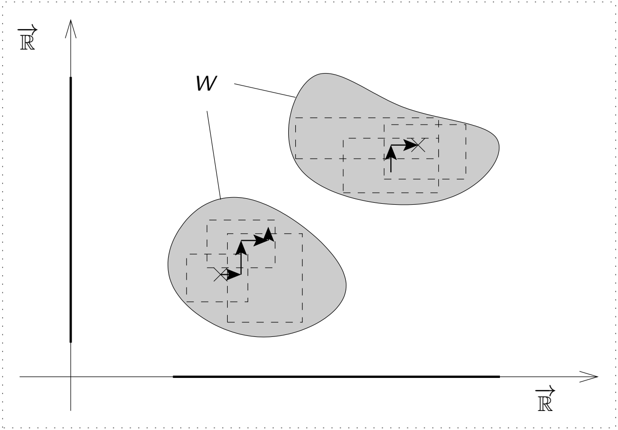
Stream products



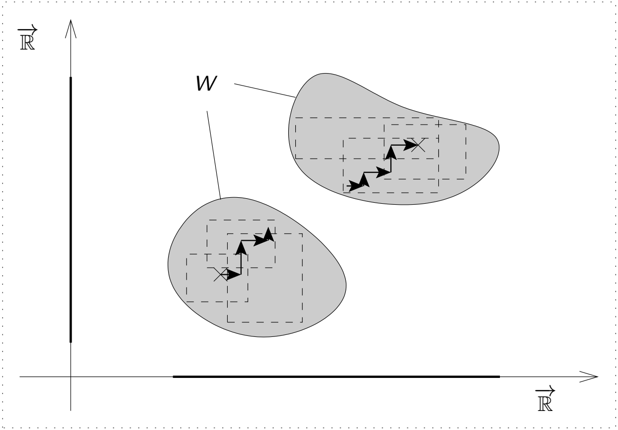
Stream products



Stream products



Stream products



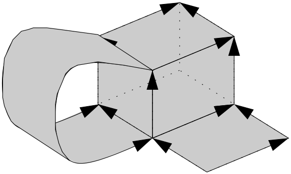
Geometric realization revisited

As with prestreams... only replace $[0,1]^n$ by $\overrightarrow{[0,1]}^n$

Definition (Geometric realization functor **Geom**: **Cub** → **Str**)

$\mathbf{Geom}((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n) = (\coprod_n K_n \cdot \overrightarrow{[0,1]}^n)/\equiv$ where:

$$
\begin{array}{rcl}
(a, (x_1, \cdots, 1, \cdots, x_n)) & \equiv & (\partial_i^+ a, (x_1, \cdots, x_n)) \\
(a, (x_1, \cdots, 0, \cdots, x_n)) & \equiv & (\partial_i^- a, (x_1, \cdots, x_n)) \\
(a, (x_1, \cdots, \hat{x}_i, \cdots x_n)) & \equiv & (s_i a, (x_1, \cdots, x_{n+1}))
\end{array}
$$



All is well

Conversely, there is a *singular cube* functor $\textbf{Sing}\colon \textbf{Str} \to \textbf{Cub}$.
$\textbf{Sing}(\mathcal{X}) = ((K_n)_n, (\partial_n^\alpha)_{\alpha,n}, (s_n)_n)$ where:

- $K_n = \{\text{prestream morphisms } \gamma\colon \overrightarrow{[0,1]}^n \to \mathcal{X}\}$
- $\partial_i^+\gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, 1, \cdots, x_n)$
- $\partial_i^-\gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, 0, \cdots, x_n)$
- $s_i\gamma\colon (x_1, \cdots, x_n) \mapsto \gamma(x_1, \cdots, \hat{x}_i, \cdots, x_n)$

Theorem

Geom *is left adjoint to* **Sing**

Only the beginning of the theory... much remains to be done

Outline

1 Directed algebraic topology
- Space, plus time
- Concurrent processes

2 Prestreams
- Prestreams
- Quotients, colimits, glueing
- Geometric realization

3 Streams
- The issue with prestreams
- Streams
- Stream products
- Geometric realization

4 Conclusion

Conclusion

- Streams are a (the?) *right* model for space+(cyclic) **time**
- Other nice model: Grandis's d-spaces...
  close to streams: adjunction [Haucourt 09]
  ...restricting to an equivalence ⇒ *Haucourt streams*
- Agenda: adapt whatever can be
  from (undirected) algebraic topology
  - Seems much harder than expected
  - E.g., van Kampen is hard [Goubault, Haucourt 07]
  - My own current itch (with E. Goubault, J. Dubut):
    good theories of directed homology