

Categories

- Set of **objects** \mathcal{C}_0
- Set of **arrows** \mathcal{C}_1
- For each arrow $f \in \mathcal{C}_1$, a **domain** $\text{dom}(f) \in \mathcal{C}_0$ and a **co-domain** $\text{cod}(f) \in \mathcal{C}_0$
- (Write $f : A \rightarrow B$ if $\text{dom}(f) = A$ and $\text{cod}(f) = B$)
- For each object $A \in \mathcal{C}_0$, an **identity arrow** $\text{id}_A \in \mathcal{C}_0$
- For each $f_1 : A \rightarrow B$ and $f_2 : B \rightarrow C$, a **composite** $f_2 \circ f_1 : A \rightarrow C$,
- with **associativity**: $f_3 \circ (f_2 \circ f_1) = (f_3 \circ f_2) \circ f_1$ whenever these are defined,
- and **identities**: for all arrows $f : A \rightarrow B$, $f \circ \text{id}_A = f$ and $\text{id}_B \circ f = f$.
- **That's all folks**:
 $\mathcal{C}_0, \mathcal{C}_1, \text{dom}, \text{cod} : \mathcal{C}_1 \rightarrow \mathcal{C}_0, \text{id} : \mathcal{C}_0 \rightarrow \mathcal{C}_1, \circ : \mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1 \rightarrow \mathcal{C}_1$

Initial and terminal objects

Definition: Let \mathcal{C} be a category and $\perp, \top \in \mathcal{C}$ objects.

- \perp is an **initial object** if there is exactly one arrow $\perp \rightarrow A$ for every $A \in \mathcal{C}$.
- \top is a **terminal object** if there is exactly one arrow $A \rightarrow \top$ for every $A \in \mathcal{C}$.

(Note the duality.)

Examples: **Set**, **Graph**, transition systems, poset-as-category, pointed sets

Arrows **from** terminal objects **pick out elements**.

Example: in **Set**, an element of a set A is the same as an arrow $\top \rightarrow A$.

Category Theory and Functional Programming

Day 3

21 October 2009

Constructions in categories

- 1 Categories
- 2 Initial and terminal objects
- 3 Products and co-products
- 4 Equalizers and co-equalizers
- 5 Limits and co-limits
- 6 Uniqueness up to isomorphism
- 7 Pullbacks and pushouts
- 8 Adjoins preserve (co-)limits

Products and co-products

Definition: Let \mathcal{C} be a category and $A, B \in \mathcal{C}$ objects.

- A **product** of A and B consists of an object $P = A \times B$ of \mathcal{C} and (“projection”) arrows $\pi_A : P \rightarrow A, \pi_B : P \rightarrow B$ with the property that:

for any $C \in \mathcal{C}$ with arrows $f_A : C \rightarrow A$

and $f_B : C \rightarrow B$, there is **exactly one**

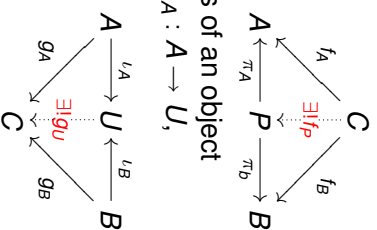
arrow $f_P : C \rightarrow P$ for which

$$\pi_A \circ f_P = f_A \text{ and } \pi_B \circ f_P = f_B$$

- **Dually:** A **co-product** of A and B consists of an object

$U = A \sqcup B$ of \mathcal{C} and (“injection”) arrows $\iota_A : A \rightarrow U,$

$\iota_B : B \rightarrow U$ with the property that



5/31

Products and co-products

Examples:

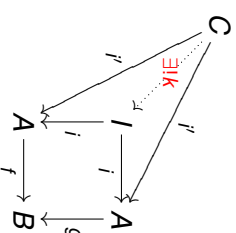
- Products in **Set, Graph, Mon**
- Co-products in **Set, Graph, Mon**
- Co-products in **Set**, $\ast = \top \downarrow \text{Set}$
- Product in **Graph** vs. product in **RGraph**

6/31

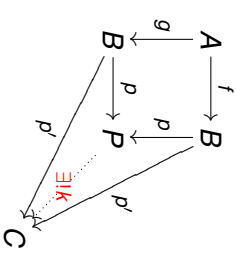
Equalizers and co-equalizers

Definition: Let \mathcal{C} be a category and $f, g : A \rightarrow B \in \mathcal{C}$ arrows.

- An **equalizer** of f and g consists of an object $I \in \mathcal{C}$ and an arrow $i : I \rightarrow A$ for which



- A **co-equalizer** of f and g consists of an object $P \in \mathcal{C}$ and an arrow $p : A \rightarrow P$ for which



$$I \xrightarrow{i} A \xrightleftharpoons[f]{g} B \xrightarrow{p} P$$

7/31

Equalizers and co-equalizers

Example, in **Set**:

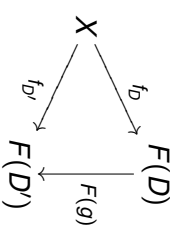
$$I \xrightarrow{i} A \xrightleftharpoons[f]{g} B \xrightarrow{p} P$$

- $I = \{x \in A \mid f(x) = g(x)\}$
- $i : I \hookrightarrow A$ **inclusion**
- $P =$ set of equivalence classes B/\sim , where \sim is the smallest equivalence relation for which $f(x) \sim g(x)$ for all $x \in A$
- $p : x \mapsto [x]$ **projection**

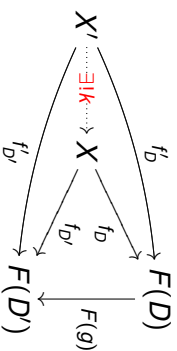
8/31

Limits

- A **(commutative) diagram** in a category \mathcal{C} is a functor $F : \mathcal{D} \rightarrow \mathcal{C}$ from a (usually quite small) category \mathcal{D} .
- A **cone** for such a diagram F consists of an object $X \in \mathcal{C}$ and arrows $f_D : X \rightarrow F(D)$ for all objects $D \in \mathcal{D}$ such that for all arrows $g : D \rightarrow D' \in \mathcal{D}$,



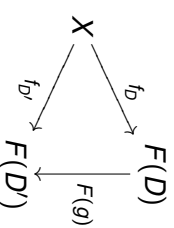
- A **limit** for such a diagram F is a cone $(X, \{f_D\})$ such that for all cones $(X', \{f'_{D'}\})$,



9/31

Limits

- A **(commutative) diagram** in a category \mathcal{C} is a functor $F : \mathcal{D} \rightarrow \mathcal{C}$ from a (usually quite small) category \mathcal{D} .
- A **cone** for such a diagram F consists of an object $X \in \mathcal{C}$ and arrows $f_D : X \rightarrow F(D)$ for all objects $D \in \mathcal{D}$ such that for all arrows $g : D \rightarrow D' \in \mathcal{D}$,



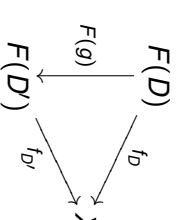
- A **cone** for F consists of an object $X \in \mathcal{C}$ and a **natural transformation** $f : X \rightarrow F$, where $X : \mathcal{D} \rightarrow \mathcal{C}$ is the **constant functor** $D \mapsto X, f \mapsto \text{id}_X$.

- These form a **category of cones over F** .
- A **limit** for F is a **terminal object** in this category.

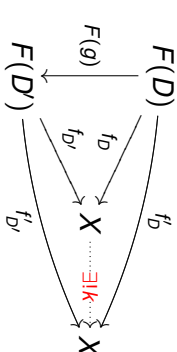
10/31

Co-limits

- A **co-cone** for a diagram $F : \mathcal{D} \rightarrow \mathcal{C}$ consists of an object $X \in \mathcal{C}$ and a natural transformation $f : F \rightarrow X$, where $X : \mathcal{D} \rightarrow \mathcal{C}$ is the constant functor $D \mapsto X, f \mapsto \text{id}_X$:



- A **co-limit** is a terminal object in the category of co-cones over F :



11/31

Examples

- **terminal object** = limit of the empty diagram
- **initial object** = co-limit of the empty diagram
- **product** $A \times B$ = limit of the diagram $A \ B$ (no arrows)
- **co-product** $A \sqcup B$ = co-limit of the diagram $A \ B$ (no arrows)
- **equalizer** of $f, g : A \rightarrow B$ = limit of the diagram $A \begin{matrix} \xrightarrow{f} \\ \xrightarrow{g} \end{matrix} B$
- **co-equalizer** of $f, g : A \rightarrow B$ = co-limit of the diagram $A \begin{matrix} \xrightarrow{f} \\ \xrightarrow{g} \end{matrix} B$

12/31

Uniqueness up to isomorphism

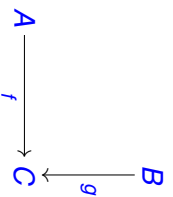
Terminal and initial objects are **unique up to isomorphism**



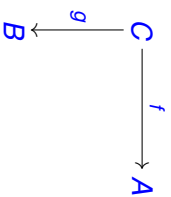
All limits and co-limits are unique up to isomorphism

Pullbacks and pushouts

A **pullback** is a limit of a diagram

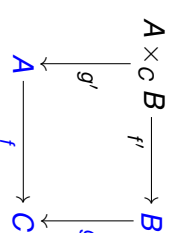


A **pushout** is a co-limit of a diagram

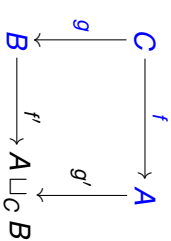


Pullbacks and pushouts

A **pullback** is a limit of a diagram

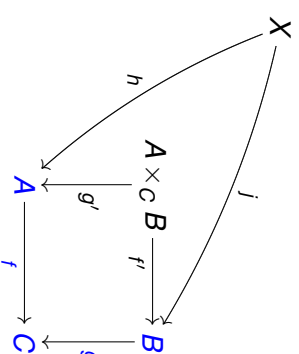


A **pushout** is a co-limit of a diagram

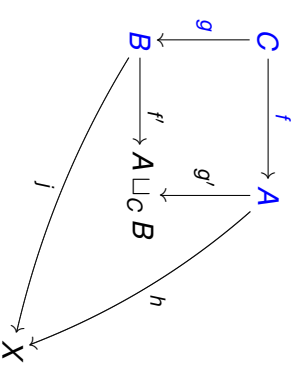


Pullbacks and pushouts

A **pullback** is a limit of a diagram

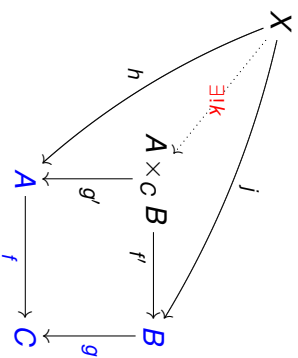


A **pushout** is a co-limit of a diagram

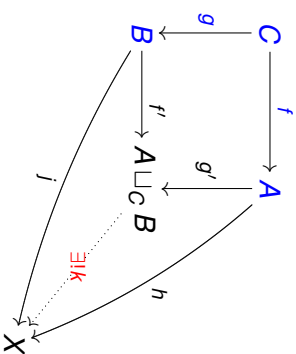


Pullbacks and pushouts

A **pullback** is a limit of a diagram



A **pushout** is a co-limit of a diagram



17/31

Adjoint preserve limits

Theorem: If $G : \mathcal{E} \rightarrow \mathcal{C}$ has a **left adjoint** and $D : \mathcal{D} \rightarrow \mathcal{E}$ has a **limit** (X, f) , then $G \circ D : \mathcal{D} \rightarrow \mathcal{C}$ has limit $(G(X), G \circ f)$.

- “Right adjoints preserve limits”

Dual theorem: If $F : \mathcal{C} \rightarrow \mathcal{E}$ has a **right adjoint** and $D : \mathcal{D} \rightarrow \mathcal{C}$ has a **co-limit** (X, f) , then $G \circ D : \mathcal{D} \rightarrow \mathcal{E}$ has co-limit $(G(X), G \circ f)$.

- “Left adjoints preserve co-limits”

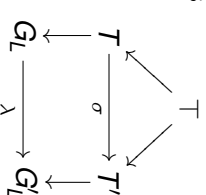
18/31

Categorical constructions for transition systems

- 9 Transition systems
- 10 Re-labeling
- 11 Product
- 12 Restriction
- 13 Parallel composition

Transition systems

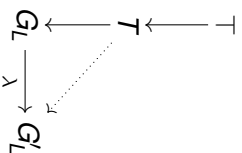
- Recall: Category of transition systems = **pointed arrow category** $\mathbf{T} \downarrow \mathbf{RGraph} \rightarrow \mathbf{RGraph}$
- objects $T \rightarrow T \rightarrow G_L$
 - terminal graph \rightarrow graph \rightarrow one-point graph
 - initial point \rightarrow graph \rightarrow labeling
- morphisms



20/31

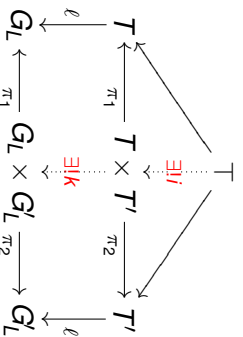
Re-labeling

Re-labeling of a transition system $T \rightarrow T \rightarrow G_L$ by a label morphism $\lambda : L \rightarrow L'_\perp$:



Product

Product of transition systems $T \rightarrow T \rightarrow G_L, T \rightarrow T' \rightarrow G'_L$:



- Arrows $T \xrightarrow{l} T \times T' \xrightarrow{k} G_L \times G'_L$ given uniquely because of product.

- The labeling is $G_L \times G'_L = G_{L \cup L' \cup L \times L'}$, or

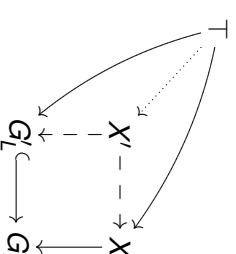
$$L_\perp \times L'_\perp = \{(a, b), (a, \perp), (\perp, b), (\perp, \perp) \mid a \in L, b \in L'\}$$

- This is the **product** in the category $T \downarrow \mathbf{RGraph}^1$

Restriction

Restriction of a transition system $T \rightarrow T \rightarrow G_L$ to a subset $L' \hookrightarrow L$:

Pullback



Parallel composition

For **parallel composition** $(T \rightarrow T \rightarrow G_L) \parallel (T \rightarrow T' \rightarrow G'_L)$:

- Form product $(T \rightarrow T \rightarrow G_L) \times (T \rightarrow T' \rightarrow G'_L)$
 - This is **completely synchronized**: contains **all** possible combinations $(a, b), (a, \perp), (\perp, b)$ of labels \Rightarrow all possible synchronizations

- Restrict by an inclusion $S \hookrightarrow L_\perp \times L'_\perp$

Specifies **which synchronizations are allowed**

- For **CCS**: $S = \{(a, \bar{a}), (b, \bar{b}), \dots\}$
- For **CSP**: $S = \{(a, a), (b, b), \dots\}$
- etc.** (!)

- Re-label

- For **CCS**: $(a, \bar{a}) \mapsto \tau, (b, \bar{b}) \mapsto \tau, \dots$
- For **CSP**: $(a, a) \mapsto a, (b, b) \mapsto b, \dots$
- etc.**

Parallel composition

- Theorem: All types of parallel composition can be expressed using product, restriction, and re-labeling.
- Product: limit. Restriction: pullback – limit. Re-labeling: composition
- ⇒ All types of parallel composition are combinations of **limits** and **composition**.
- ⇒ All types of parallel composition are **preserved by right adjoints**.
- Recall: **Unfolding** from transition systems to synchronization trees is a **right adjoint**
- Corollary: If \parallel is any type of parallel composition, then the unfolding of a \parallel is the \parallel of the unfoldings.

28 / 31

Solutions to recursive domain equations

- 14 Domains; fixed-point theorem
- 15 Recursive domain equations
- 16 Generalized fixed-point theorem

Domains; fixed-point theorem

Recall:

- A **domain** is a set D together with a partial order $\sqsubseteq \subseteq D \times D$
 - which contains a **least element** $\perp \in D$, and
 - in which every **increasing chain** $x_1 \sqsubseteq x_2 \sqsubseteq \dots$ has a **least upper bound** (lub).
- A function $f : D \rightarrow D'$ of domains is **continuous** if
 - f is monotone: $x \sqsubseteq_D y \Rightarrow f(x) \sqsubseteq_{D'} f(y)$, and
 - f preserves lub's: for any increasing chain $S \subseteq D$, $f(\text{lub } S) = \text{lub } f(S)$.
- Domains and continuous functions form a **category Dom**.
- A **fixed point** of an endofunction $f : D \rightarrow D$ is an element $x \in D$ for which $f(x) = x$.
- **Fixed-point theorem**: A continuous endofunction $f : D \rightarrow D$ has a **least fixed point** x^* , and $x^* = \text{lub}\{f^i(\perp) \mid i \in \mathbb{N}\}$.

27 / 31

Recursive domain equations

Recall:

- In operational semantics, we need **recursively defined sets**.
For example

$$\text{Env}_P = \text{Phavne} \multimap \text{Kom} \times \text{Env}_P$$

- This is actually a recursively defined **domain** (with subset (“specialization”) ordering $\sqsubseteq = \subseteq$)
- This is quite common. For example untyped lambda-calculus:

$$\text{Expr} = \text{Expr} \multimap \text{Expr}$$

- Or lambda-calculus with constants A :

$$\text{Expr} = A \cup (\text{Expr} \multimap \text{Expr})$$

- **Problematic**, because **this does not work for general sets!**

28 / 31

