Category Theory and Functional Programming

Day 3

21 October 2009

Constructions in categories

Categories

- Set of objects $\mathcal{C}_0$
- Set of arrows $\mathcal{C}_1$
- For each arrow $f \in \mathcal{C}_1$, a domain $dom(f) \in \mathcal{C}_0$ and a co-domain $cod(f) \in \mathcal{C}_0$
- (Write $f : A \to B$ if $dom(f) = A$ and $cod(f) = B$)
- For each object $A \in \mathcal{C}_0$, an identity arrow $\mathrm{id}_A \in \mathcal{C}_0$
- For each $f_1 : A \to B$ and $f_2 : B \to C$, a composite $f_2 \circ f_1 : A \to C$,
- 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 $f : A \to B$, $f \circ \mathrm{id}_A = f$ and $\mathrm{id}_B \circ f = f$.
- *That's all folks:* $\mathcal{C}_0, \mathcal{C}_1, dom, cod : \mathcal{C}_1 \to \mathcal{C}_0, \mathrm{id} : \mathcal{C}_0 \to \mathcal{C}_1, \circ : \mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1 \to \mathcal{C}_1$

Initial and terminal objects

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

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

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 $\pi_A : P \to A$, $\pi_B : P \to B$ with the property that:

  for any $C \in \mathcal{C}$ with arrows $f_A : C \to A$ and $f_B : C \to B$, there is exactly one arrow $f_P : C \to P$ for which $\pi_A \circ f_P = f_A$ and $\pi_B \circ f_P = f_B$

$$\begin{tikzcd} & C \arrow[dl, "f_A"'] \arrow[d, dotted, "\exists! f_P"] \arrow[dr, "f_B"] & \\ A & P \arrow[l, "\pi_A"] \arrow[r, "\pi_b"'] & B \end{tikzcd}$$

- Dually: A co-product of A and B consists of an object $U = A \sqcup B$ of $\mathcal{C}$ and ("injection") arrows $\iota_A : A \to U$, $\iota_B : B \to U$ with the property that

$$\begin{tikzcd} A \arrow[r, "\iota_A"] \arrow[dr, "g_A"'] & U \arrow[d, dotted, "\exists! g_U"] & B \arrow[l, "\iota_B"'] \arrow[dl, "g_B"] \\ & C & \end{tikzcd}$$

Products and co-products

Examples:

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

Equalizers and co-equalizers

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

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

$$\begin{tikzcd} C \arrow[drr, "i'"] \arrow[dr, dotted, "\exists! k"] \arrow[ddr, "i'"'] & & \\ & I \arrow[r, "i"'] \arrow[d, "i"] & A \arrow[d, "g"] \\ & A \arrow[r, "f"'] & B \end{tikzcd}$$

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

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "g"'] & B \arrow[d, "p"'] \arrow[ddr, "p'"] & \\ B \arrow[r, "p"] \arrow[drr, "p'"'] & P \arrow[dr, dotted, "\exists! k"] & \\ & & C \end{tikzcd}$$

$$I \xrightarrow[i]{} A \overset{f}{\underset{g}{\rightrightarrows}} B \xrightarrow[p]{} P$$

Equalizers and co-equalizers

Example, in **Set**:

$$I \xrightarrow[i]{} A \overset{f}{\underset{g}{\rightrightarrows}} B \xrightarrow[p]{} P$$

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

Limits

- A (commutative) diagram in a category $\mathcal{C}$ is a functor $F : \mathcal{D} \to \mathcal{C}$ 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 $f_D : X \to F(D)$ for all objects $D \in \mathcal{D}$ such that for all arrows $g : D \to D' \in \mathcal{D}$,

$$\begin{tikzcd} & F(D) \arrow[d, "F(g)"] \\ X \arrow[ur, "f_D"] \arrow[r, "f_{D'}"'] & F(D') \end{tikzcd}$$

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

$$\begin{tikzcd} & & F(D) \arrow[d, "F(g)"] \\ X' \arrow[r, dotted, "\exists! k"] \arrow[urr, bend left, "f'_D"] \arrow[rr, bend right, "f'_{D'}"'] & X \arrow[ur, "f_D"] \arrow[r, "f_{D'}"'] & F(D') \end{tikzcd}$$

Limits

- A (commutative) diagram in a category $\mathcal{C}$ is a functor $F : \mathcal{D} \to \mathcal{C}$ 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 $f_D : X \to F(D)$ for all objects $D \in \mathcal{D}$ such that for all arrows $g : D \to D' \in \mathcal{D}$,

$$\begin{tikzcd} & F(D) \arrow[d, "F(g)"] \\ X \arrow[ur, "f_D"] \arrow[r, "f_{D'}"'] & F(D') \end{tikzcd}$$

- A cone for F consists of an object $X \in \mathcal{C}$ and a natural transformation $f : \tilde{X} \dot{\to} F$, where $\tilde{X} : \mathcal{D} \to \mathcal{C}$ is the constant functor $D \mapsto X$, $f \mapsto \mathrm{id}_X$.
- These form a category of cones over F.
- A limit for F is a terminal object in this category.

Co-limits

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

$$\begin{tikzcd} F(D) \arrow[d, "F(g)"'] \arrow[dr, "f_D"] & \\ F(D') \arrow[r, "f_{D'}"'] & X \end{tikzcd}$$

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

$$\begin{tikzcd} F(D) \arrow[d, "F(g)"'] \arrow[dr, "f_D"] \arrow[drr, bend left, "f'_D"] & & \\ F(D') \arrow[r, "f_{D'}"'] \arrow[rr, bend right, "f'_{D'}"'] & X \arrow[r, dotted, "\exists! k"] & X' \end{tikzcd}$$

Examples

- terminal object $=$ limit of the empty diagram
- initial object $=$ co-limit of the empty diagram
- product $A \times B =$ limit of the diagram $A \quad B$ (no arrows)
- co-product $A \sqcup B =$ co-limit of the diagram $A \quad B$ (no arrows)
- equalizer of $f, g : A \to B =$ limit of the diagram $A \overset{f}{\underset{g}{\rightrightarrows}} B$
- co-equalizer of $f, g : A \to B =$ co-limit of the diagram $A \overset{f}{\underset{g}{\rightrightarrows}} B$

Uniqueness up to isomorphism

Terminal and initial objects are unique up to isomorphism

$$\Downarrow$$

All limits and co-limits are unique up to isomorphism

Pullbacks and pushouts

A pullback is a limit of a diagram

$$\begin{tikzcd} & B \arrow[d, "g"] \\ A \arrow[r, "f"'] & C \end{tikzcd}$$

A pushout is a co-limit of a diagram

$$\begin{tikzcd} C \arrow[r, "f"] \arrow[d, "g"'] & A \\ B & \end{tikzcd}$$

Pullbacks and pushouts

A pullback is a limit of a diagram

$$\begin{tikzcd} A \times_C B \arrow[r, "f'"'] \arrow[d, "g'"] & B \arrow[d, "g"] \\ A \arrow[r, "f"'] & C \end{tikzcd}$$

A pushout is a co-limit of a diagram

$$\begin{tikzcd} C \arrow[r, "f"] \arrow[d, "g"'] & A \arrow[d, "g'"'] \\ B \arrow[r, "f'"] & A \sqcup_C B \end{tikzcd}$$

Pullbacks and pushouts

A pullback is a limit of a diagram

$$\begin{tikzcd} X \arrow[drr, bend left, "j"] \arrow[ddr, bend right, "h"'] & & \\ & A \times_C B \arrow[r, "f'"'] \arrow[d, "g'"] & B \arrow[d, "g"] \\ & A \arrow[r, "f"'] & C \end{tikzcd}$$

A pushout is a co-limit of a diagram

$$\begin{tikzcd} C \arrow[r, "f"] \arrow[d, "g"'] & A \arrow[d, "g'"'] \arrow[ddr, bend left, "h"] & \\ B \arrow[r, "f'"] \arrow[drr, bend right, "j"'] & A \sqcup_C B & \\ & & X \end{tikzcd}$$

Pullbacks and pushouts

A pullback is a limit of a diagram

$$\begin{tikzcd} X \arrow[dr, dotted, "\exists! k"] \arrow[drr, bend left, "j"] \arrow[ddr, bend right, "h"'] & & \\ & A \times_C B \arrow[r, "f'"] \arrow[d, "g'"] & B \arrow[d, "g"] \\ & A \arrow[r, "f"] & C \end{tikzcd}$$

A pushout is a co-limit of a diagram

$$\begin{tikzcd} C \arrow[r, "f"] \arrow[d, "g"] & A \arrow[d, "g'"] \arrow[ddr, bend left, "h"] & \\ B \arrow[r, "f'"] \arrow[drr, bend right, "j"'] & A \sqcup_C B \arrow[dr, dotted, "\exists! k"] & \\ & & X \end{tikzcd}$$

Categorical constructions for transition systems

Adjoints preserve limits

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

- "Right adjoints preserve limits"

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

- "Left adjoints preserve co-limits"

Transition systems

- Recall: Category of transition systems = pointed arrow category $\top \downarrow$ **RGraph** $\to$ **RGraph**[1]
- objects $\top \to T \to G_L$
  – terminal graph $\to$ graph $\to$ one-point graph
  – initial point $\to$ graph $\to$ labeling
- morphisms

$$\begin{tikzcd} & \top \arrow[dl] \arrow[dr] & \\ T \arrow[rr, "\sigma"] \arrow[d] & & T' \arrow[d] \\ G_L \arrow[rr, "\lambda"] & & G'_L \end{tikzcd}$$

Re-labeling

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

$$\begin{tikzcd} \top \arrow[d] & \\ T \arrow[d] \arrow[dr, dotted] & \\ G_L \arrow[r, "\lambda"'] & G'_L \end{tikzcd}$$

Product

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

$$\begin{tikzcd} & \top \arrow[dl] \arrow[dr] \arrow[d, dotted, "\exists! i"] & \\ T \arrow[d, "\ell"'] & T \times T' \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] \arrow[d, dotted, "\exists! k"] & T' \arrow[d, "\ell'"] \\ G_L & G_L \times G'_L \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & G'_L \end{tikzcd}$$

- Arrows $\top \xrightarrow{i} T \times T' \xrightarrow{k} G_L \times G'_L$ given uniquely because of product.
- The labeling is $G_L \times G'_L = G_{L \sqcup L' \sqcup L \times L'}$, 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 $\top \downarrow$ **RGraph** $\to$ **RGraph**[1]

Restriction

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

Pullback

$$\begin{tikzcd} \top \arrow[dr, dotted] \arrow[drr, bend left] \arrow[ddr, bend right] & & \\ & X' \arrow[r, dashed] \arrow[d, dashed] & X \arrow[d] \\ & G'_L \arrow[r, hook] & G_L \end{tikzcd}$$

Parallel composition

For parallel composition $(\top \to T \to G_L) \,||\, (\top \to T' \to G'_L)$:

1. Form product $(\top \to T \to G_L) \times (\top \to T' \to G'_L)$
   - This is completely synchronized: contains all possible combinations $(a, b), (a, \perp), (\perp, b)$ of labels $\Rightarrow$ all possible synchronizations
2. 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. (!)
3. 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

$\Rightarrow$ All types of parallel composition are combinations of limits and composition.

$\Rightarrow$ All types of parallel composition are preserved by right adjoints.

- Recall: Unfolding from transition systems to synchronization trees is a right adjoint
- Corollary: If $||$ is any type of parallel composition, then the unfolding of a $||$ is the $||$ of the unfoldings.

Solutions to recursive domain equations

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 $\bot \in D$, and
  - in which every increasing chain $x_1 \sqsubseteq x_2 \sqsubseteq \cdots$ has a least upper bound (lub).
- A function $f : D \to D'$ 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 $S \subseteq D$, $f(\operatorname{lub} S) = \operatorname{lub} f(S)$.
- Domains and continuous functions form a category **Dom**.
- A fixed point of an endofunction $f : D \to D$ is an element $x \in D$ for which $f(x) = x$.
- Fixed-point theorem: A continuous endofunction $f : D \to D$ has a least fixed point x^*, and $x^* = \operatorname{lub}\{f^i(\bot) \mid i \in \mathbb{N}\}$.

Recursive domain equations

Recall:

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

$$\mathbf{Env}_P = \mathbf{Pnavne} \rightharpoonup \mathbf{Kom} \times \mathbf{Env}_P$$

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

$$\mathbf{Expr} = \mathbf{Expr} \rightharpoonup \mathbf{Expr}$$

- Or lambda-calculus with constants A:

$$\mathbf{Expr} = A \cup (\mathbf{Expr} \rightharpoonup \mathbf{Expr})$$

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

Recursive domain equations

General question:

- If F is a function from domains to domains: Under what conditions does the equation $D = F(D)$ have a meaningful solution?

Solution by categorification:

- Let $F : \mathbf{Dom} \to \mathbf{Dom}$ be a functor. Find conditions under which the equation $D = F(D)$ has a least fixed point up to isomorphism, and a way to compute it.

Definition (P-3.4.1): A fixed point for a functor $F : \mathbf{Dom} \to \mathbf{Dom}$ is a pair (D, d) of a domain $D \in \mathbf{Dom}$ and an isomorphism $d : F(D) \to D$.
A pre-fixed point is a pair (D, d) with an arrow $d : F(D) \to D$.

- Want to find an initial fixed point.

Generalized fixed-point theorem

- Pre-fixed points and fixed points form categories: arrows:

$$\begin{tikzcd} F(D) \arrow[r, "F(f)"] \arrow[d, "d"'] & F(D') \arrow[d, "d'"] \\ D \arrow[r, "f"'] & D' \end{tikzcd}$$

- We are looking for an initial object in the category of fixed points.
- Lemma (P-3.4.2): An initial pre-fixed point is also an initial fixed point.

Generalized fixed-point theorem

- The one-point domain $\perp = \{\perp\}$ is both initial and terminal in **Dom**.
- Theorem: Let $p : \perp \to F(\perp)$ be the unique arrow, and look at the (infinite) diagram

  $$\perp \xrightarrow{p} F(\perp) \xrightarrow{F(p)} F^2(\perp) \xrightarrow{F^2(p)} F^3(\perp) \xrightarrow{F^3(p)} \cdots$$

  F has an initial pre-fixed point, which is the co-limit of this diagram.

- This looks like

$$\begin{tikzcd} \perp \arrow[r, "p"] \arrow[drrrrr] & F(\perp) \arrow[r, "F(p)"] \arrow[drrrr] & F^2(\perp) \arrow[r, "F^2(p)"] \arrow[drrr] & F^3(\perp) \arrow[r, "F^3(p)"] \arrow[drr] & \cdots & \\ & & & & \cdots & D \end{tikzcd}$$

(this is called a projective limit)