Category Theory in Foundations of Computer Science
2019/2020

Concepts, terminology and notation:

A *weighted graph* $G = \langle N, E, s, t, r, w\rangle$ consists of two sets, N of *nodes* and E of *edges*, with functions $s, t: E \to N$ indicating, respectively, the source and target of each node, a *root* $r \in N$, and a *weight* function $w: E \to \mathcal{N}$, where $\mathcal{N}$ is the set of natural numbers. G is *finite* if $N \cup E$ is a finite set. G is a *weighted tree* if its underlying graph is a tree, i.e., if for each node $n \in N$, there is a unique path from R to n in the graph.

A weighted graph morphism $\theta: G \to G'$, where $G = \langle N, E, s, t, r, w\rangle$ and $G' = \langle N', E', s', t', r', w'\rangle$, consists of two functions $\theta = \langle \theta_{node}: N \to N', \theta_{edge}: E \to E'\rangle$ that preserves sources and targets of edges, the root, and does not increase the weights, i.e., for each $e \in E$, $s'(\theta_{edge}(e)) = \theta_{node}(s(e))$, $t'(\theta_{edge}(e)) = \theta_{node}(t(e))$, and $w(e) \geq w'(\theta_{edge}(e))$, and $\theta_{node}(r) = r'$. With the obvious morphism composition, this yields the category **WGraph** of weighted graphs and their morphisms, and its full subcategory **WTree** of weighted trees and their morphisms. Let $\mathcal{J}: \mathbf{WTree} \to \mathbf{WGraph}$ be the obvious inclusion functor.

We also have their respective full subcategories **FWGraph** of finite weighted graphs and **FWTree** of finite weighted trees, with inclusion functor $\mathcal{FJ}: \mathbf{FWTree} \to \mathbf{FWGraph}$.

To do:

Prove or justify a negative answer to the following questions:

1. Consider categories:
   (a) **WGraph**
   (b) **FWGraph**
   (c) **WTree**
   (d) **FWTree**
   Which of the categories above has all
   FC. i) finite products, ii) equalisers, iii) finite limits
   FCC. i) finite coproducts, ii) coequalisers, iii) finite colimits
   C. i) products, ii) equalisers, iii) limits
   CC. i) coproducts, ii) coequalisers, iii) colimits
2. Consider functors:
   (a) $\mathcal{J}: \mathbf{WTree} \to \mathbf{WGraph}$
   (b) $\mathcal{FJ}: \mathbf{FWTree} \to \mathbf{FWGraph}$
   Which of the functors above
   C. is continuous
   CC. is cocontinuous
   L. has a left adjoint
   R. has a right adjoint

Notes:

The questions above are not independent. For instance, a proof of **1.b.C.i** is likely to be a proof of **1.b.FC.i** as well, and a counterexample to **1.b.FC.ii** is a counterexample to **1.b.FC.iii** and is likely to yield a counterexample to **1.b.C.ii** and **1.b.C.iii**. No need to repeat detailed arguments in such cases, indicating the dependency is enough.

Sketch of a solution:

Limits in WGraph

Consider a family of weighted graphs $\mathcal{G} = \{G_i = \langle N_i, E_i, s_i, t_i, r_i, w_i\rangle \mid i \in \mathcal{I}\}$. Consider the following weigthed graph $G = \langle N, E, s, t, r, w\rangle$, where

- $N = \prod_{i\in\mathcal{I}} N_i$
- $E = \{\langle e_i\rangle_{i\in\mathcal{I}} \in \prod_{i\in\mathcal{I}} E_i \mid \{w_i(e_i) \mid i \in \mathcal{I}\}$ is bounded in $\mathcal{N}\}$
- $s(\langle e_i\rangle_{i\in\mathcal{I}}) = \langle s_i(e_i)\rangle_{i\in\mathcal{I}}$, $t(\langle e_i\rangle_{i\in\mathcal{I}}) = \langle t_i(e_i)\rangle_{i\in\mathcal{I}}$
- $r = \langle r_i\rangle_{i\in\mathcal{I}}$
- $w(\langle e_i\rangle_{i\in\mathcal{I}}) = \max(\{w_i(e_i) \mid i \in \mathcal{I}\})$, where $\max(\emptyset) = 0$ (this is well-defined for $\langle e_i\rangle_{i\in\mathcal{I}} \in E$)

Then G with the obvious morpphisms $\pi_i : G \to G_i$ is the product of $\mathcal{G}$: given a weighted graph $G' = \langle N', E', s', t', r', w'\rangle$ with morphisms $\theta_i : G' \to G_i$, the unique morphism $\theta : G' \to G$ such that $\theta;\pi_i = \theta_i$, $i \in \mathcal{I}$, is given by $\theta(n') = \langle\theta_i(n')\rangle_{i\in\mathcal{I}}$, for $n' \in N'$, and $\theta(e') = \langle\theta_i(e')\rangle_{i\in\mathcal{I}}$, for $e' \in E'$, which is well-defined since $w'(e') \geq w_i(\theta_i(e'))$, $i \in \mathcal{I}$, and so $\{w_i(\theta_i(e')) \mid i \in \mathcal{I}\}$ is bounded in $\mathcal{N}$.

Note: the construction above works for $\mathcal{I} = \emptyset$, yielding the weigthed graph G_1 with a root and a single edge going from the root to the root, with weight 0.

Given two morphisms $\theta, \theta' : G \to G'$, where $G = \langle N, E, s, t, r, w\rangle$ and $G' = \langle N', E', s', t', r', w'\rangle$, their equaliser is given by the inclusion into G of the following weighted graph: $G_0 = \langle N_0, E_0, s_0, t_0, r_0, w_0\rangle$, where

- $N_0 = \{n \in N \mid \theta(n) = \theta'(n)\}$
- $E_0 = \{n \in E \mid \theta(e) = \theta'(e)\}$
- s_0, t_0 and w_0 coincide with s, t and w, respectively, on their arguements
- $r_0 = r$

Consequently, **WGraph** has all limits (and so all finite limits as well)

Limits in FWGraph

Given a full subcategory $\mathbf{K}'$ of $\mathbf{K}$, if a limit of a diagram in $\mathbf{K}'$ exists in $\mathbf{K}$ and is in $\mathbf{K}'$, then it is also a limit of this diagram in $\mathbf{K}'$.

It is easy to check that the construction of the limits ensures that a limit of a finite diagram of finite weigthed graphs is finite, hence **FWGraph** has all finite products, equalisers, and all finite limits as well.

However, products of infinite famlies of finite weigthed graphs need not exist in **FWGraph**. For instance, consider a weigthed tree T_2, with two edges going out of its root, with both weights being 0. Suppose that for an infinite $\mathcal{I}$, there exists a product P in **FWGraph** of $\mathcal{T} = \langle T_i\rangle_{i\in\mathcal{I}}$, where $T_i = T_2$ for each $i \in \mathcal{I}$. Let T_1 be a weighted tree with a single edge going out of the root, with weight 0. Then there are infinitely many distincts families $\langle\theta_i : T_1 \to T_i\rangle_{i\in\mathcal{I}}$, hence there must be infinitely many morphisms $\theta : T_1 \to P$ – and so P must have infinitely many edges, which yields a contradiction.

Colimits in WGraph

It's easy to check that given a family of weighted graphs, its colimit in **WGraph** is given as the "disjoint union with a new root replacing all old roots".

Then, given two morphisms $\theta, \theta' \colon G \to G'$, where $G = \langle N, E, s, t, r, w \rangle$ and $G' = \langle N', E', s', t', r', w' \rangle$, their equaliser is given by the inclusion into G of the following weighted graph $G'' = \langle N'', E'', s'', t'', r'', w'' \rangle$, where

- $N'' = N' / \equiv_{node}$, where $\equiv_{node}$ is the least equivalence on N' such that $\theta(n) \equiv_{node} \theta'(n)$, for each $n \in N$
- $E'' = E' / \equiv_{edge}$, where $\equiv_{edge}$ is the least equivalence on E' such that $\theta(e) \equiv_{node} \theta'(e)$, for each $e \in E$
- $s''([e']_{\equiv_{edge}}) = [s'(e')]_{\equiv_{node}}$, and $t''([e']_{\equiv_{edge}}) = [t'(e')]_{\equiv_{node}}$ for each $e' \in E'$ (note that the congruence property holds, so this is well defined)
- $w''([e']_{\equiv_{edge}}) = min(\{w'(e_0) \mid e_0 \equiv_{edge} e'\})$ for each $e' \in E'$
- $r'' = [r]_{\equiv_{node}}$

In other words, colimits in **WGraph** are constructed on colimits in the category of graphs, with weights added in the obvious way.

Consequently, all colimits in **WGraph** exist.

Colimits in FWGraph

Coeqaulisers and finite coproducts, hence all finite colimits, carry over from **WGraph** to **FWGraph** (dually to limits).

However, infinite coproducts need not exist in **FWGraph**. To see this, a counterexample may be constructed almost dually to that for infinite products in **FWGraph**: suppose that for an infinite $\mathcal{I}$, there exists a product C in **FWGraph** of $\mathcal{T} = \langle T_i \rangle_{i \in \mathcal{I}}$, where $T_i = T_1$ for each $i \in \mathcal{I}$. There are infinitely many distinct families of morphisms $\theta_i \colon (T_i = T_1) \to T_2$. Hence there must be infinitely many morphisms from C to T_2 — which is impossible when C has finitely many edges.

Limits in WTree

Non-empty products in **WGraph** may be adjusted to yield products in **WTree** as follows. Consider a family of weighted trees $\mathcal{T} = \{T_i = \langle N_i, E_i, s_i, t_i, r_i, w_i \rangle \mid i \in \mathcal{I}\}$, and let $G = \langle N, E, s, t, r, w \rangle$ be its products in **WGraph**, with projections $\pi_i \colon G \to T_i$. Let P be a reachable part of G. Then each π_i restricted to P is a weighted graph morphism. Hence, P is a weighted tree (since for any $i \in \mathcal{I}$, T_i is a weighted tree). Then it is easy to check that P with such restricted projections is a product of $\mathcal{T}$ in **WTree**.

Then, the "single infinite line with weighths 0" tree $T_l = \langle N_l, E_l, s_l, t_l, r_l, w_l \rangle$ is a terminal object in **WTree**, where $N_l = \mathcal{N}$, $E_l = \mathcal{N}$, $s_l(k) = k$, $t_l(k) = k + 1$, $r_l = 0$, $w_l(k) = 0$, for all $k \in Nat$.

The construction of equalisers in **WGraph** works for **WTree** as well.

Hence, **WTree** has all limits (and so all finite limits as well).

Limits in FWTree

The construction of equalisers in **WTree** works for **FWTree** as well. So does the construction of non-empty products of finite families.

However, there is no terminal object in **FWTree**: suppose $T_?$ is a terminal object in **FWTree**. Then it is easy to check, that from each node in $T_?$ there may be at most one outgoing edge. Hence $T_?$ is a finite prefix of T_l. None of them is terminal in **FWTree** though, since there are no morphisms to any such prefix from "longer" prefixes of T_l.

Moreover, products of infinite families in **FWTree** need not exist: the counterexample for infinite products in **WGraph** works here as well.

Colimits in WTree

The construction of coproducts carries over from **WGraph** to **WTree**. So does the construction of coequalisers — given $\theta, \theta' : T \to T'$, where $T = \langle N, E, s, t, r, w \rangle$ and $T' = \langle N', E', s', t', r', w' \rangle$, it is enough to notice here that for any node $n \in N$, the path from r to n in T is mapped by θ to the path from r' to $\theta(n)$, and by θ' to the path from r' to $\theta'(n)$, and so the construction of colimit of θ and θ' in **WGraph** yieds a tree when both T and T' are trees.

Consequently, all colimits in **WTree** exist.

Colimits in FWTree

Coeqaulisers and finite coproducts, hence all finite colimits, carry over from **WTree** to **FWTree**.

However, infinite coproducts need not exist in **FWTree**: the counterexample for infinite coproducts in **FWGraph** applies here as well.

Continuity and cocontinuity of $\mathcal{J}$ and $\mathcal{FJ}$

$\mathcal{J}$: **WTree** $\to$ **WGraph** does not preserve the terminal object, hence is not continuous. The constructions of colimits in **WTree** coincide with those for **WGraph**, hence $\mathcal{J}$ is cocontinuous.

$\mathcal{FJ}$: **FWTree** $\to$ **FWGraph** does not preserve products, hence is not continuous. The constructions of finite colimits in **FWTree** coincide with those for **FWGraph**, hence $\mathcal{FJ}$ is finitely cocontinuous. Those infinite colimits in **FWTree** that exist are preserves by $\mathcal{FJ}$ as well.

Adjoints to $\mathcal{J}$ and $\mathcal{FJ}$

Since neither $\mathcal{J}$ nor $\mathcal{FJ}$ is continuous, neither has a left adjoint.

If $\mathcal{FJ}$: **FWTree** $\to$ **FWGraph** had a right adjoint, then this right eadjoint would have to map the terminal object in **FWGraph**, which exists, to a terminal object in **FWTree**, which does not exist – hence $\mathcal{FJ}$ does not have a right adjoint.

$\mathcal{J}$: **WTree** $\to$ **WGraph** does have the right adjoint: this is the "unfolding functor" $\mathcal{U}$: **WGraph** $\to$ **WTree**, where for any weigthed graph $G = \langle N, E, s, t, r, w \rangle$, $\mathcal{U}(G)$ is the weithed tree of paths in G starting in r: such paths are nodes in $\mathcal{U}(G)$, edges in $\mathcal{U}(G)$ expand the paths by one edge from G, and weights in $\mathcal{U}(G)$ are inherited from G.