2-1993

A Generalization of the Trie Data Structure

Richard H. Connelly

F. Lockwood Morris
Syracuse University, lockwood@ecs.syr.edu

SU-CIS-93-23

A Generalization of the Trie Data Structure

Richard H. Connelly and F. Lockwood Morris

February 1993

School of Computer and Information Science
Syracuse University
Suite 4-116, Center for Science and Technology
Syracuse, NY 13244-4100

A GENERALIZATION OF THE TRIE DATA STRUCTURE

RICHARD H. CONNELLY AND F. LOCKWOOD MORRIS

ABSTRACT. Tries, a form of string-indexed look-up structure, are generalized to permit indexing by terms built according to an arbitrary signature. The construction is parametric with respect to the type of data to be stored as values; this is essential, because the recursion which defines tries appeals from one value type to others. "Trie" (for any fixed signature) is then a functor, and the corresponding look-up function is a natural isomorphism.

The trie functor is in principle definable by the "initial fixed point" semantics of Smyth and Plotkin. We simplify the construction, however, by introducing the "category-cpo", a class of category within which calculations can retain some domain-theoretic flavor. Our construction of tries extends easily to many-sorted signatures.

Section 1. Introduction.

A look-up table—a finite data structure intended for the retrieval of values which have been stored corresponding to "keys"—is naturally regarded as a concrete implementation of what abstractly is a function from keys to values, but just what sort of function deserves some consideration. If the value type is A, which for convenience we shall think of as always containing a distinguished element or *base point* $*_A$—we call such a type a *pointed set*—and the key type is Y, we may say that a table models a function from Y to A whose value is $*_A$ (representing the absence of a genuine value) for all but finitely many arguments; we introduce the notation $A^{[Y]}$ for the set of all such functions.

Tries [4,6] are a form of look-up table suited to the situation where keys are strings over a finite alphabet. Our innovation here will be to extend the possible sets of keys from "strings over any finite alphabet" to "terms built with any finite signature of operators". We begin by giving a description of ordinary string-indexed tries (omitting optimizations found in more practically-oriented treatments) in such a way as to make the generalization to indexing by terms as obvious as possible.

Let H be the set of finite strings over a finite alphabet $\{c_1, \ldots, c_{m-1}\}$ for any $m \geq 1$. (To be sure, if m is one, H contains only the empty string, but there is no reason to forbid that case.) An H-indexed, A-valued *trie* is a finite $(m-1)$-ary tree, its nodes labelled by elements of A. (We shall be wanting to consider tree nodes as m-tuples, with the label as mth component; this accounts for our taking the size of the alphabet to be $m-1$.) A trie is searched by the evident recursive principle: the empty $(m-1)$-ary tree, which we denote by $\bullet$ (pronounced "spot"), represents that function in $A^{[H]}$ whose value is $*_A$ for every string, while an m-tuple

Early stages of the second author's research were supported by the SERC "IPSE 2.5" grant to Manchester University.

Typeset by $\mathcal{A}_{\mathcal{M}}\mathcal{S}$-TEX

$\langle r_1, \ldots, r_{m-1}, a\rangle$ represents the function whose value at the empty string is a and whose value at a non-empty string $c_i h$ is the value at h of the function represented by r_i.

To put this more formally, we may regard H as a *term algebra:* the set of all terms which can be built with $m-1$ unary operators ("prefix c_i") and one nullary operator ("empty string"). [This is more often called a "word algebra"; we go against convention because we shall be wanting "word" in a distinct technical sense in Section 4.] To make a specific construction, we may, following Reynolds [9], define H as the least solution of the set equation

$$H = \sum_{i=1}^{m} (\text{if } i < m \text{ then } H \text{ else } \{\langle\,\rangle\})$$

where by the summation notation we intend the following specific disjoint union of sets:

$$\sum_{i=1}^{m} Z_i \stackrel{\text{def}}{=} \{\langle i, z\rangle \mid 1 \leq i \leq m \text{ and } z \in Z_i\}.$$

Here $\{\langle\,\rangle\}$ is a convenient one-element set—we take the zero-tuple for its element in the same spirit that we say the empty string is a nullary operator. Then, as is well known, H is explicitly given by $H = \bigcup_n H_n$, where $H_0 = \emptyset$, and for $n \geq 0$,

$$H_{n+1} = \sum_{i=1}^{m} (\text{if } i < m \text{ then } H_n \text{ else } \{\langle\,\rangle\}).$$

(A remark on notation: We will be doing a lot of indexing from zero to infinity; we therefore will write things like $\bigcup_n H_n$ as shorthand for $\bigcup_{n\geq 0} H_n$.) Given this construction of H, we can write an explicit recursive program for the look-up or "apply" function:

$$\begin{aligned}
&\text{ap} \bullet h = *_A \\
&\text{ap}\, \langle r_1, \ldots, r_{m-1}, a\rangle \langle m, \langle\,\rangle\rangle = a \\
&\text{ap}\, \langle r_1, \ldots, r_{m-1}, a\rangle \langle i, h\rangle = \text{ap}\, r_i\, h \qquad \text{for } 1 \leq i < m.
\end{aligned}$$

Before leaving the case of strings as keys, we give a more rigorous, and also slightly more restrictive, definition of the set of H-indexed, A-valued tries. Note that our first description, that a trie was any A-labelled $(m-1)$-ary tree, allowed the everywhere-$*_A$ function to be represented not only by $\bullet$, the empty tree, but also by any tree all of whose node labels were $*_A$. We now decide to require that $\bullet$ be the only allowed representation for this function, so as to get a one-to-one correspondence between the functions in $A^{[H]}$ and their representing tries. To this end we introduce a modified Cartesian product $\prod_\bullet$ (called "spot product", if this is not too cutesy) defined for any m pointed sets by

$$\prod_{i=1}^{m}{}_{\bullet} A_i \stackrel{\text{def}}{=} \prod_{i=1}^{m} A_i - \{\langle *_{A_1}, \ldots, *_{A_m}\rangle\} \cup \{\bullet\}$$

with $\bullet$ taken to be the base point of the resulting set.

Now we can say that our set of tries is the least solution to the set equation

$$R = \prod_{i=1}^{m}{}_{\bullet}(\text{if } i < m \text{ then } R \text{ else } A)$$

and is given explicitly by

$$\begin{aligned} R_0 &= \{\bullet\} \\ R_{n+1} &= \prod_{i=1}^{m}{}_{\bullet}(\text{if } i < m \text{ then } R_n \text{ else } A) \\ R &= \bigcup_n R_n. \end{aligned}$$

The reader will have noticed the parallelism between the constructions of R and H, and may foresee that the one-to-one correspondence between R and $A^{[H]}$ will prove to be a consequence of a "law of exponents" which we give as:

Proposition 1.1. *If $X_1, \ldots, X_m$ are sets, $m \geq 0$, and A is a pointed set, then there is a one-to-one correspondence*

$$\mu_m : A^{[X_1]} \times \cdots \times A^{[X_m]} \cong A^{[X_1 + \cdots + X_m]}$$

given by

$$\mu_m : \langle g_1, \ldots, g_m \rangle \mapsto \lambda \langle i, z \rangle . g_i\, z, \qquad \mu_m^{-1} : f \mapsto \big\langle \lambda x_1 . f \langle 1, x_1 \rangle, \ldots, \lambda x_m . f \langle m, x_m \rangle \big\rangle.$$

Proof. We know that the formula for μ_m gives a one-to-one correspondence for unrestricted functions, $\mu_m : A^{X_1} \times \cdots \times A^{X_m} \cong A^{X_1 + \cdots + X_m}$; this is the coproduct property of the m-ary disjoint union in **Set**. We have only to notice that this correspondence cuts down to the almost-everywhere-$*_A$ functions: we have $\langle g_1, \ldots, g_m \rangle \in A^{[X_1]} \times \cdots \times A^{[X_m]}$ if and only if, for $i = 1, \ldots, m$, each $g_i\, x_i \neq *_A$ just for some finite set of values of x_i, say for $x_i \in \{x_{i1}, \ldots, x_{in_i}\}$; this is the same as to say that $\mu_m \langle g_1, \ldots, g_m \rangle \langle i, z \rangle \equiv (\lambda \langle i, z \rangle . g_i\, z) \langle i, z \rangle$ is different from $*_A$ just for $\langle i, z \rangle$ one of the finite set of values $\{\langle 1, x_{11} \rangle, \ldots, \langle 1, x_{1n_1} \rangle, \ldots, \langle m, x_{m1} \rangle, \ldots, \langle m, x_{mn_m} \rangle\}$, i.e., that $\mu_m \langle g_1, \ldots, g_m \rangle \in A^{[X_1 + \cdots + X_m]}$. $\square$

For any set Y and pointed set A we take $*_{A[Y]}$ to be the constant function $\lambda y . *_A$. We may then observe further that the correspondence μ_m is base point preserving. Also, there is an evident base-point-preserving, one-to-one correspondence between the modified and the ordinary Cartesian products of $A^{[X_1]}, \ldots, A^{[X_m]}$. Composing these correspondences, we may record for later reference:

Definition 1.2. *Denote by $\mu_m^{\bullet} : \prod_{i=1}^{m}{}_{\bullet} A^{[X_i]} \cong A^{[\Sigma_{i=1}^{m} X_i]}$ the base-point-preserving, one-to-one correspondence*

$$\begin{aligned} \bullet &\mapsto \lambda \langle i, z \rangle . *_A \\ \langle g_1, \ldots, g_m \rangle &\mapsto \lambda \langle i, z \rangle . g_i\, z\, . \end{aligned}$$

We have just seen string-indexed tries presented with strings taken to be the elements of a particular term algebra. We now generalize to keys which are the elements of an arbitrary term algebra, first one-sorted and then, in Section 4, many-sorted. A term algebra is characterized by its operators, say m of them; the only significant property of each operator is the number of operands it expects (its *arity*), a non-negative integer k_i for $i = 1, \ldots, m$. Thus, following Reynolds [9], we may take our generic one-sorted term algebra to be the least set T solving the equation

$$T = \sum_{i=1}^{m} T^{k_i},$$

namely

$$T = \bigcup_n T_n, \qquad \text{where} \quad T_0 = \emptyset \quad \text{and} \quad T_{n+1} = \sum_{i=1}^{m} T_n^{k_i}.$$

The key to generalizing the trie idea, still obtaining a set in one-to-one correspondence with $A^{[T]}$, in the presence of operations of unrestricted arity is another law of exponents.

Proposition 1.3. *If $X_1, \ldots, X_k$ are sets and A is a pointed set, then there is a one-to-one correspondence*

$$\nu_k : A^{[X_k]\cdots[X_1]} \cong A^{[X_1 \times \cdots \times X_k]}$$

given by

$$\nu_k : g \mapsto \lambda\langle x_1, \ldots, x_k\rangle . g\, x_1 \cdots x_k, \qquad \nu_k^{-1} : f \mapsto \lambda x_1 \cdots \lambda x_k . f\langle x_1, \ldots, x_k\rangle.$$

Proof. We treat first the case $k = 2$, that is $\nu_2 : A^{[Y]^{[X]}} \cong A^{[X\times Y]}$. For functions without restriction, this is the "uncurrying" isomorphism, part of the Cartesian closed structure of **Set**. Again, we have only to check that it cuts down to the almost-everywhere-$*$ functions. Suppose $f = \nu_2\, g$, and suppose that $f \in A^{[X\times Y]}$, i.e., that for some N we have $f\langle x, y\rangle \neq *_A$ only for $\langle x, y\rangle \in \{\langle x_1, y_1\rangle, \ldots, \langle x_N, y_N\rangle\}$. Then $g\, x \neq *_{A^{[Y]}}$ only for x in the finite set (possibly with fewer than N elements) $\{x_1, \ldots, x_N\}$, and even when x is one of these, $g\, x\, y \neq *_A$ only for $\langle x, y\rangle$ in the set of N pairs, so that $g\, x \in A^{[Y]}$ in all cases, and $g \in A^{[Y]^{[X]}}$. Conversely, if $g \in A^{[Y]^{[X]}}$, then $g\, x \neq *_{A^{[Y]}}$ only for x in some finite set $\{x_1, \ldots x_M\}$, and for $i = 1, \ldots, M$, $g\, x_i\, y \neq *_A$ only for, say, $y \in \{y_{i1}, \ldots, y_{in_i}\}$, which is to say that $\nu_2\, g\langle x, y\rangle \neq *_A$ only for the $n_1 + \cdots + n_M$ pairs $\langle x_1, y_{11}\rangle, \ldots, \langle x_M, y_{Mn_M}\rangle$, so that $\nu_2\, g \in A^{[X\times Y]}$.

Now we may treat general $k \geq 0$ by induction. For $k = 0$, it is immediate that $\nu_0 : A \cong A^{[\{\langle\rangle\}]}$, where $\nu_0\, a = \lambda\langle\,\rangle.a$, and $\nu_0^{-1}\, f = f\langle\,\rangle$. Then supposing inductively that any ν_k has been shown to be a one-to-one correspondence, we may observe that ν_{k+1} is the composition

$$A^{[X_{k+1}]\cdots[X_2]^{[X_1]}} \xrightarrow{\lambda g.\nu_k \circ g} A^{[X_2\times\cdots\times X_{k+1}]^{[X_1]}} \xrightarrow{\nu_2} A^{[X_1\times(X_2\times\cdots\times X_{k+1})]}$$
$$\xrightarrow{\lambda f. f\circ\lambda\langle x_1,\ldots,x_{k+1}\rangle.\langle x_1,\langle x_2,\ldots,x_{k+1}\rangle\rangle} A^{[X_1\times\cdots\times X_{k+1}]}$$

of which each step is a one-to-one correspondence (the instance of ν_2 in the middle step is with $X \equiv X_1,\ Y \equiv X_2 \times \cdots \times X_{k+1}$) since following any $g \in A^{[X_{k+1}]\cdots[X_1]}$ from left to right we find

$$g \mapsto \lambda x_1 \lambda\langle x_2, \ldots, x_{k+1}\rangle . g\, x_1 \cdots x_{k+1} \mapsto \lambda\langle x_1, \langle x_2, \ldots, x_{k+1}\rangle\rangle . g\, x_1 \cdots x_{k+1}$$
$$\mapsto \lambda\langle x_1, \ldots, x_{k+1}\rangle . g\, x_1 \cdots x_{k+1} \equiv \nu_{k+1} g. \quad \square$$

A generalized trie has to represent a function whose values are functions. That is, an ordinary trie and its sub-tries all represent elements of $A^{[H]}$, whereas a generalized trie and its parts have to represent elements of $A^{[T]}$, $A^{[T]^{[T]}} \cong A^{[T\times T]}$, etc. But we can think of $A^{[T]^{[T]}}$, for example, as two iterations of "$-^{[T]}$" at the set A; so if we can abstract on the type A, and regard the generalized trie idea as a scheme for representing functions from T to any type, then we should be able to iterate this scheme twice at A to get a representation of $A^{[T]^{[T]}}$. This motivates the following definition of the set of T-indexed, A-valued (generalized) tries, where now A is an explicit parameter ranging over pointed sets:

$$R_0(A) = \{\bullet\}$$
$$R_{n+1}(A) = \prod_{i=1}^{m}{}_{\bullet} R_n^{(k_i)}(A),$$

here $R_n^{(k_i)}(A)$ denotes $R_n(\ldots(R_n(A))\ldots)$ with k_i iterations of R_n, and

$$R(A) = \bigcup_n R_n(A),$$

with $*_{R(A)} = \bullet$ independent of A. When $k_i = 1$ for $i < m$ and $k_m = 0$, $R(A)$ reproduces the string-indexed tries as previously defined (that is, $R(A) = R$).

As an example term algebra let us take binary trees. We simulate a one-bit label on each node by providing two binary constructors; that is, we take $m = 3$, $k_1 = 0$ (to construct the empty binary tree), $k_2 = k_3 = 2$; we may express this more comprehensibly as the recursive set definition

$$T_{\mathrm{B}} = \{\langle\,\rangle\} + T_{\mathrm{B}}{}^2 + T_{\mathrm{B}}{}^2 .$$

Then the T_{B}-indexed, A-valued tries have the corresponding recursive definition

$$R_{\mathrm{B}}(A) = A \times R_{\mathrm{B}}(R_{\mathrm{B}}(A)) \times R_{\mathrm{B}}(R_{\mathrm{B}}(A)) .$$

To actually make an example of a trie, let $\mathbf{Z}_* \stackrel{\text{def}}{=} \mathbf{Z} \cup \{*\}$, that is the integers extended with a base point; then the function from T_{B} to $\mathbf{Z}_*$ that maps the one-node binary tree $\langle 2, \langle\langle 1, \langle\rangle\rangle, \langle 1, \langle\rangle\rangle\rangle\rangle$ to 7 and the two-node binary tree $\langle 2, \langle\langle 1, \langle\rangle\rangle, \langle 3, \langle\langle 1, \langle\rangle\rangle, \langle 1, \langle\rangle\rangle\rangle\rangle\rangle\rangle$ to 8, everything else to $*$, is represented by the trie

$$\langle *, \langle\langle 7, \bullet, \langle\langle 8, \bullet, \bullet\rangle, \bullet, \bullet\rangle\rangle, \bullet, \bullet\rangle, \bullet\rangle .$$

Returning to the general treatment, it seems intuitively reasonable to suppose that the following equations define a family of look-up functions, also parameterized by the pointed set of possible values, and similar to what we had in the string-indexed case but more recursion-intensive:

$$\begin{aligned}
&\mathrm{ap}_A \bullet t = *_A \\
&\mathrm{ap}_A \langle r_1, \ldots, r_m\rangle\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle \\
&\qquad\qquad = \mathrm{ap}_A(\mathrm{ap}_{R(A)}(\cdots(\mathrm{ap}_{R^{(k_i-1)}(A)}\, r_i\, t_1)\cdots)\, t_{k_i-1})\, t_{k_i}.
\end{aligned}$$

Note that values of i for which $k_i = 0$ correspond to nullary operators, that is constants, of the term algebra; for such i we have

$$\mathrm{ap}_A \langle r_1, \ldots, r_m\rangle\langle i, \langle\rangle\rangle = r_i.$$

It is by encountering a nullary subterm that ap is able to take a step towards escaping from its apparently ever-more-deeply-nesting recursion.

It may be easier to make sense of ap specialized to our binary-tree-indexed example, which comes out as

$$\begin{aligned}
&\text{B-ap}_A \bullet b = *_A \\
&\text{B-ap}_A\langle r_1, r_2, r_3\rangle\langle 1, \langle\rangle\rangle = r_1 \\
&\text{B-ap}_A\langle r_1, r_2, r_3\rangle\langle 2, \langle b_1, b_2\rangle\rangle = \text{B-ap}_A(\text{B-ap}_{R_{\mathrm{B}}(A)}\, r_2\, b_1)\, b_2 \\
&\text{B-ap}_A\langle r_1, r_2, r_3\rangle\langle 3, \langle b_1, b_2\rangle\rangle = \text{B-ap}_A(\text{B-ap}_{R_{\mathrm{B}}(A)}\, r_3\, b_1)\, b_2\,.
\end{aligned}$$

The reader may care to verify that $\text{B-ap}_{\mathbf{Z}_*}$ actually will return 7 and 8 from the example trie for the appropriate two binary trees as keys, and $*$ for other keys.

We will prove, by the end of Section 3, that this ap actually is well defined. (In reality, of course, one wants to implement ap as a single subroutine, not an infinite family. Apparently, all ap really needs to know about the type at which it is supposedly working is the relevant base point which it might have to return. Thus a practical program might be

$$\begin{aligned}
&\mathrm{ap}'(bp, \bullet)\, t = bp \\
&\mathrm{ap}'(bp, \langle r_1, \ldots, r_m\rangle)\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle = \mathrm{ap}'(bp, \mathrm{ap}'(\bullet, \cdots(\mathrm{ap}'(\bullet, r_i)\, t_1)\cdots)\, t_{k_i-1})\, t_{k_i}.
\end{aligned}$$

In situations where it can be arranged that $\bullet$ and $*_A$ are identical, even the parameter bp would be unnecessary. We shall, however, not pursue this line further, preferring to keep the value type as a parameter.)

Stated in terms of sets, what we hope to establish, in order to show that the generalized tries really do represent finite functions, is a one-to-one correspondence $R(A) \cong A^{[T]}$ for every pointed set A. In outline, the proof goes as follows: For any set Z, write $\mathcal{F}_Z$ for the set-to-set mapping $A \mapsto A^{[Z]}$; then Proposition 1.1 and Proposition 1.3 will yield

$$\prod_{i=1}^{m}{}_{\bullet}\, \mathcal{F}_Z^{(k_i)}(A) \cong \mathcal{F}_{(\sum_{i=1}^{m} Z^{k_i})}(A) \qquad \text{for all } A.$$

Then we may hope to prove by induction that for all $n \geq 0$,

$$R_n(A) \cong \mathcal{F}_{T_n}(A) \equiv A^{[T_n]}.$$

For $n = 0$, $\{\bullet\} \cong A^{[\emptyset]}$; as an inductive step, calculate

$$R_{n+1}(A) \equiv \prod_{i=1}^{m}{}_{\bullet} R_n^{(k_i)}(A) \cong \prod_{i=1}^{m}{}_{\bullet} \mathcal{F}_{T_n}^{(k_i)}(A) \cong \mathcal{F}_{(\sum_{i=1}^{m} T_n^{k_i})}(A) \equiv \mathcal{F}_{T_{n+1}}(A) \equiv A^{[T_{n+1}]}.$$

One would then like to conclude that, in the limit,

$$R(A) \equiv \bigcup_n R_n(A) \cong A^{[\bigcup_n T_n]} \equiv A^{[T]}.$$

Making this calculation rigorous, and showing that the family of one-to-one correspondences $R(A) \cong A^{[T]}$ it yields is in a suitable sense the least fixed point of the recursion equations for ap, will be the purpose of the following two sections, with assistance from the appendix, where we have segregated such necessary definitions and theorems as belong entirely to category theory.

Not surprisingly, we are able to view the $\mathcal{F}$ introduced above as a (Curried) two-argument functor, having its second argument and its result in the category $\mathbf{Set}_*$ of pointed sets and base-point-preserving functions, but taking its first argument (the Z in $A^{[Z]}$) from the partial order of sets and set inclusions. By this choice for its domain, we are able to define $\mathcal{F}$ so as to be covariant in both arguments, and are spared the difficulties which led Smythe and Plotkin in [11], needing a covariant arrow bifunctor within a category of domains, to introduce a subcategory with "embeddings" as morphisms. Nevertheless, $\mathcal{F}$ bears some resemblance to an exponential functor, and the isomorphism of Proposition 1.3 is much the same as that which gives $\mathbf{Set}_*$ with smash product as a tensor product—or equivalently the category of sets and partial functions with Cartesian product as tensor product—its monoidal closed structure (see, for example, Poigné [8]).

The function R which constructs for any A the T-indexed, A-valued tries is in fact an endofunctor of $\mathbf{Set}_*$. Moreover, ap is a natural isomorphism from R to the functor $-^{[T]}$, as we shall show. The formula given above for the construction of each R_{n+1} from R_n amounts to the definition of an endofunctor $\mathcal{R}$ of $\mathbf{Set}_*^{\mathbf{Set}_*}$, of which R is an "initial fixed point" as defined by Smyth and Plotkin [11].

It is because the definition of ap_A appeals to $\mathrm{ap}_{R(A)}$, etc.—in other words, because ap must be polymorphic to work at all—that we are compelled to look for a whole functor, rather than a single data type, to be an initial fixed point.

It was originally our intention to carry out the rigorous construction of R and the natural isomorphism ap as a straightforward application of the Smyth and Plotkin method, which generalizes the familiar "least fixed point of a continuous function" construction from domain theory to a categorical setting. However, it has turned out that, despite the presence of categories and functors, the trie construction retains a very domain-theoretical flavor, because the sets $R(A)$ are unions of inclusion towers, as is also the set T of terms. It therefore has seemed to us worthwhile to make a preparatory digression, introducing in Section 2 a kind of domain-category hybrid, the "category-cpo". This will allow our desired natural isomorphism of functors to be constructed (in Section 3) as the least upper bound of an ascending chain rather than, as would be done by a more general category-theoretic treatment, as the colimit of a general ω-sequence of objects and morphisms.

There seems to be a growing recognition that category theory is relevant not only to semantics, but to the more mundane algorithms-and-data-structures side of computer science. Spivey [12], for example, uncovers natural transformations and adjunctions in familiar list-processing functions. It may not, however, be generally appreciated that the construction of even a first-order data type can, as here, call for categorical methods. The present paper uses rather a lot of mathematics to arrive at a modest algorithmic result, but we hope that some of the tools developed here will be reusable in other applications.

Section 2. The notion "Category-cpo".

It is a commonplace observation (see for example [7, p. 11]) that a partial order may be regarded as a category in which each hom-set contains at most one morphism, and which moreover is *skeletal:* isomorphic objects are identical (this, together with the uniqueness of morphisms, entails that the only isomorphisms are the identities). For a category which in this way "is" a partial order, $\sqsubseteq$, on its objects, we will, when a and b are objects such that $a \sqsubseteq b$, write $(a \sqsubseteq b)$ as a notation for the (unique) morphism.

Definition 2.1. *A category-partial order (category-po for short) is a pair $\langle K, \sqsubseteq \rangle$ where K is a category and $\sqsubseteq$ is a subcategory of K which is a partial order on all the objects of K, and such that the identities are the only morphisms of $\sqsubseteq$ that are isomorphisms in K.*

The morphisms of the subcategory $\sqsubseteq$ will be called the "inequalities" of K. Note that this usage of "inequalities" includes also "equalities", i.e., identity morphisms. The last condition in the definition is equivalent to requiring that the insertion functor $i : \sqsubseteq \longrightarrow K$ reflect isomorphisms [7, p. 150], since no non-identity is an isomorphism in $\sqsubseteq$. We will never have occasion to consider more than a single partial-order subcategory of any one category K; hence, by abuse of notation, we will generally write just K and not $\langle K, \sqsubseteq \rangle$ as our name for the category-po.

In any category-po $\langle K, \sqsubseteq \rangle$ a partial order is induced on all the morphisms of K, as follows: if $f : a \longrightarrow b$ and $g : c \longrightarrow d$, we write $f \sqsubseteq g$ (using the same partial order symbol as between objects) just in case $a \sqsubseteq c$, $b \sqsubseteq d$, and we have the commutative square

(1)
$$\begin{tikzcd} a \arrow[r, phantom, "\sqsubseteq"] \arrow[d, "f"'] & c \arrow[d, "g"] \\ b \arrow[r, phantom, "\sqsubseteq"] & d. \end{tikzcd}$$

Verification that $\sqsubseteq$ between morphisms is a partial order is immediate. This partial order is identical with the comma category $i{\downarrow}i$. Since i reflects isomorphisms, so does the insertion from $i{\downarrow}i$ to $I_K{\downarrow}I_K$. Consequently, $I_K{\downarrow}I_K$ is also a category-po.

It is also immediate, for any objects a and b, that $a \sqsubseteq b$ if and only if $1_a \sqsubseteq 1_b$. Composition in a category-po is monotone where defined: if $a \xrightarrow{f} b \xrightarrow{g} c$ and

$a' \xrightarrow{f'} b' \xrightarrow{g'} c'$ are such that $f \sqsubseteq f'$ and $g \sqsubseteq g'$, then $g \circ f \sqsubseteq g' \circ f'$, as one sees by pasting the squares together.

A noteworthy elementary fact is the following:

Fact 2.2. *In a category-po, if $f, g : a \longrightarrow b$ with $f \sqsubseteq g$, then $f = g$.*

This is because the commuting square by virtue of which it holds that $f \sqsubseteq g$ must have 1_a and 1_b for sides. (This observation makes it clear that category-pos are very unlike typical categories of domains and continuous functions: in a category-po, *all* the non-trivial instances of "approximation" of one morphism by another must be between morphisms from different hom-sets. A motivation for the development of category-pos may be taken from the familiar observation that if f and g are (graphs of) two total functions with the same domain, then $f \subseteq g$ implies $f = g$.)

We now introduce the principal notion with which we intend to work.

Definition 2.3. *A category-po $\langle K, \sqsubseteq \rangle$ is a category-complete partial order (for short, a category-cpo) if $\sqsubseteq$ is ω-complete (that is, every ascending ω-chain of objects has a l.u.b.) and each such l.u.b. is an ω-colimit in K.*

The requirement that each ω-l.u.b. be a colimit is identical to requiring that the insertion functor $i : \sqsubseteq \longrightarrow K$ preserve ω-l.u.b.s as colimits. (We should remark that a category-cpo is a special case of a "double category" as defined by MacLane [7, p. 44], since the instances of $\sqsubseteq$ between morphisms are certain commutative squares in K; however, we do not know how to apply this observation.)

Notational remark: We use the tuple brackets $\langle - \rangle_n$, with a binding occurrence of n indicating as with $\bigsqcup$ and the like that n runs from zero to infinity, as a notation for infinite sequences and especially ascending chains. For example, as a synonym for "sequence $k_0 \sqsubseteq k_1 \sqsubseteq k_2 \sqsubseteq \ldots$ of objects" we may write "ω-chain of objects $\langle k_n \rangle_n$".

The next lemma shows that the insertion functor $i : \sqsubseteq \longrightarrow K$ reflects [7, p. 150] colimits of ω-chains.

Lemma 2.4. *Given a category-cpo K, let $\bar{k}$ be an upper bound of an ω-chain $k_0 \sqsubseteq k_1 \sqsubseteq \cdots$ of objects of K. If $\langle (k_n \sqsubseteq \bar{k}) \rangle_n$ is a colimit cone, then $\bar{k} = \bigsqcup_n k_n$.*

Proof. Since $(\bigsqcup_n k_n \sqsubseteq \bar{k})$ mediates from colimit cone $\langle (k_n \sqsubseteq \bigsqcup_n k_n) \rangle_n$ to cone $\langle (k_n \sqsubseteq \bar{k}) \rangle_n$, it must be the unique mediating morphism (u.m.m. for short). Since $\langle (k_n \sqsubseteq \bar{k}) \rangle_n$ is also a colimit cone, $(\bigsqcup_n k_n \sqsubseteq \bar{k})$ is an isomorphism and so must be an identity. □

The product category of a family of category-cpos is a category-cpo using the componentwise ordering and l.u.b. The following proposition shows that the morphisms in a category-cpo are ω-complete.

Proposition 2.5. *Let K be any category-cpo. The morphisms of K are ω-complete using the ordering defined by Diagram 1. Specifically, for any ω-chain $\langle f_n : a_n \longrightarrow b_n \rangle_n$, there is a l.u.b. $\bigsqcup_n f_n : \bigsqcup_n a_n \longrightarrow \bigsqcup_n b_n$.*

Proof. Consider $\langle f_n \rangle_n$ as a sequence $\langle \langle a_n, b_n, f_n \rangle \rangle_n$ in $i{\downarrow}i$. Theorem A.4 gives that $P : i{\downarrow}i \longrightarrow K \times K$ creates a colimit object $\langle \bigsqcup_n a_n, \bigsqcup_n b_n, f \rangle$. The morphism f is an upper bound of $\langle f_n \rangle_n$, because the created colimit cone is $\langle\langle(a_n \sqsubseteq \bigsqcup_m a_m), (b_n \sqsubseteq \bigsqcup_m b_m)\rangle\rangle_n$. To see that f is the l.u.b., let $g : c \longrightarrow d$ be any upper bound of $\langle f_n \rangle_n$. Since $\langle c, d, g\rangle$ is a vertex of $\langle\langle a_n, b_n, f_n\rangle\rangle_n$, Fact A.1 applied to the functor P gives that $\langle(\bigsqcup_n a_n \sqsubseteq c), (\bigsqcup_n b_n \sqsubseteq d)\rangle$ is the u.m.m. from $\langle \bigsqcup_n a_n, \bigsqcup_n b_n, f\rangle$ to $\langle c, d, g\rangle$. That is, $f \sqsubseteq g$. □

Fact 2.2 gives the following useful fact.

Fact 2.6. *If $g : \bigsqcup_n a_n \longrightarrow \bigsqcup_n b_n$ is an upper bound of an ω-chain of morphisms $\langle f_n : a_n \longrightarrow b_n\rangle_n$, then $g = \bigsqcup_n f_n$.*

It follows from Fact 2.6 that $1_{\bigsqcup_n a_n} = \bigsqcup_n 1_{a_n}$ for $a_0 \sqsubseteq a_1 \sqsubseteq a_2 \sqsubseteq \cdots$. It also follows that composition of morphisms is ω-continuous: by monotonicity, $\bigsqcup_n g_n \circ \bigsqcup_n f_n \sqsupseteq g_n \circ f_n$ for each n; therefore, by Fact 2.6, $\bigsqcup_n g_n \circ \bigsqcup_n f_n = \bigsqcup_n (g_n \circ f_n)$.

For any category-cpos L and K, we say that a functor $F : L \longrightarrow K$ is *continuous* if and only if F preserves inequalities [$F(a \sqsubseteq \bar{a}) = (Fa \sqsubseteq F\bar{a})$] and is ω-continuous on objects [$F(\bigsqcup_n a_n) = \bigsqcup_n F(a_n)$]. ("Continuous" seems the only reasonable word to use in our context. Note, however, that this is not ω-continuity of functors as ordinarily defined, that is preservation of all ω-colimits.)

Any continuous functor F is ω-continuous on morphisms: First, F is monotone on morphisms: an instance of $\sqsubseteq$ between morphisms, say

$$\begin{array}{ccc} a & \sqsubseteq & \bar{a} \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \bar{f}} \\ b & \sqsubseteq & \bar{b}, \end{array}$$

is sent by F to

$$\begin{array}{ccc} Fa & \xrightarrow{F(\sqsubseteq)=\sqsubseteq} & F\bar{a} \\ {\scriptstyle Ff}\downarrow & & \downarrow{\scriptstyle F\bar{f}} \\ Fb & \xrightarrow{F(\sqsubseteq)=\sqsubseteq} & F\bar{b}. \end{array}$$

Then, for any ω-chain of morphisms $\langle f_n : a_n \longrightarrow b_n\rangle_n$, monotonicity gives that $F(\bigsqcup_n f_n) \sqsupseteq F(f_n)$ for each n. Since $F(\bigsqcup_n f_n) : F(\bigsqcup_n a_n) = \bigsqcup_n F(a_n) \longrightarrow F(\bigsqcup_n b_n) = \bigsqcup_n F(b_n)$, Fact 2.6 gives $F(\bigsqcup_n f_n) = \bigsqcup_n F(f_n)$.

For any continuous functors $F, F' : L \longrightarrow K$, let $\tau : F \stackrel{\cdot}{\longrightarrow} F'$ be any natural transformation. For any $a \sqsubseteq \bar{a}$ in L, we have the commutative diagram

$$\begin{array}{ccc} Fa & \xrightarrow{\tau_a} & F'a \\ {\scriptstyle F(\sqsubseteq)=\sqsubseteq}\downarrow & & \downarrow{\scriptstyle F'(\sqsubseteq)=\sqsubseteq} \\ F\bar{a} & \xrightarrow{\tau_{\bar{a}}} & F'\bar{a}. \end{array}$$

That is, $a \sqsubseteq \bar{a}$ implies $\tau_a \sqsubseteq \tau_{\bar{a}}$. For an ω-chain $\langle a_n\rangle_n$ in L, it follows that $\tau_{\bigsqcup_n a_n} \sqsupseteq \tau_{a_n}$ for each n. Since the domain of $\tau_{\bigsqcup_n a_n}$ is $F(\bigsqcup_n a_n) = \bigsqcup_n F(a_n)$ and the codomain is $F'(\bigsqcup_n a_n) = \bigsqcup_n F'(a_n)$, Fact 2.6 gives $\tau_{\bigsqcup_n a_n} = \bigsqcup_n \tau_{a_n}$. That is, any natural transformation between continuous functors is a ω-continuous map from objects to morphisms.

Clearly, functor composition preserves continuity.

Let L be any (index) category and K be any category-cpo. Define a partial order $\dot{\sqsubseteq}$ between functors from L to K by

$$F \mathrel{\dot{\sqsubseteq}} \bar{F} \quad \Leftrightarrow \quad Fl \sqsubseteq \bar{F}l \text{ for every object } l \text{ and } Ff \sqsubseteq \bar{F}f \text{ for every morphism } f.$$

This is the same as to say that the assignment $l \mapsto (Fl \sqsubseteq \bar{F}l)$ is a natural transformation from F to $\bar{F}$. If $(F \mathrel{\dot{\sqsubseteq}} \bar{F})$ is an isomorphism, then each component $(Fl \sqsubseteq \bar{F}l)$ must be an identity, making $(F \mathrel{\dot{\sqsubseteq}} \bar{F})$ an identity transformation. Thus the transformations $\dot{\sqsubseteq}$ as inequalities make K^L into a category-po. The next proposition shows that K^L is a category-cpo.

Proposition 2.7. *Let L be any category and let K be any category-cpo. The category K^L is a category-cpo using the $\dot{\sqsubseteq}$ ordering. Specifically, for any ω-chain $\langle F_n : L \longrightarrow K\rangle_n$, the l.u.b. is $(\bigsqcup_n F_n)(l) = \bigsqcup_n F_n(l)$ for each object l and $(\bigsqcup_n F_n)(h) = \bigsqcup_n F_n(h)$ for each morphism h. Further, if L is a category-cpo and each F_n is continuous, then $\bigsqcup_n F_n$ is continuous.*

Proof. By Fact A.3, the functor $i^* : K^L \longrightarrow K^{|L|}$ creates a colimit object F with colimit cone $\langle (F_n \mathrel{\dot{\sqsubseteq}} F)\rangle_n$ where $F(l) = \bigsqcup_n F_n(l)$ for each object l. For any morphism h, since $F(h) \sqsupseteq F_n(h)$ for each n, Fact 2.6 gives that $F(h) = \bigsqcup_n F_n(h)$. Let G be any upper bound of $\langle F_n\rangle_n$. Since $\dot{\sqsubseteq} : i^*(F) \longrightarrow i^*(G)$ is the u.m.m. in $K^{|L|}$, Fact A.1 gives that it is also the u.m.m. in K^L. That is, $F \mathrel{\dot{\sqsubseteq}} G$ and F is the l.u.b. $\bigsqcup_n F_n$.

Now let L be a category-cpo as well and suppose that each F_n in the ω-chain $\langle F_n : L \longrightarrow K\rangle_n$ is continuous. It needs to be shown that $\bigsqcup_n F_n$ is continuous. To show that $\bigsqcup_n F_n$ preserves inequalities, let $l \sqsubseteq \bar{l}$, and calculate

$$\Big(\bigsqcup_n F_n\Big)(l \sqsubseteq \bar{l}) = \bigsqcup_n F_n(l \sqsubseteq \bar{l}) = \bigsqcup_n (F_n l \sqsubseteq F_n \bar{l}).$$

Since $\big((\bigsqcup_n F_n)(l) \sqsubseteq (\bigsqcup_n F_n)(\bar{l})\big)$ is an upper bound of the sequence of inequalities $(F_0 l \sqsubseteq F_0 \bar{l}) \sqsubseteq (F_1 l \sqsubseteq F_1 \bar{l}) \sqsubseteq \cdots$, Fact 2.6 gives us that also

$$\Big(\Big(\bigsqcup_n F_n\Big)(l) \sqsubseteq \Big(\bigsqcup_n F_n\Big)(\bar{l})\Big) = \bigsqcup_n (F_n l \sqsubseteq F_n \bar{l}).$$

To show ω-continuity on objects, let $l_0 \sqsubseteq l_1 \sqsubseteq \cdots$ be an ω-chain in L; then

$$\Big(\bigsqcup_n F_n\Big)\Big(\bigsqcup_m l_m\Big) \equiv \bigsqcup_n F_n\Big(\bigsqcup_m l_m\Big) = \bigsqcup_n \bigsqcup_m F_n(l_m) = \bigsqcup_m \bigsqcup_n F_n(l_m) = \bigsqcup_m \Big(\bigsqcup_n F_n\Big)(l_m).$$

□

If K and L are both category-cpos, we henceforth denote by K^L not the category of all functors from L to K but the full subcategory thereof whose objects are the continuous functors. Exponential objects given by this definition of K^L, together with the products noted above, may be shown to make the category of small category-cpos and continuous functors Cartesian closed.

Let K, L, M be any category-cpos. The composition functor $\circ : M^L \times L^K \longrightarrow M^K$ [7, Exercise II.6.3, p. 45] is continuous. To see this, again treating preservation of inequalities first, calculate, for any $k \in K$ (see [7, p. 43 eqn. 2] for the horizontal composition of natural transformations):

$$\begin{aligned}
[(H \mathrel{\dot\sqsubseteq} \bar{H}) \circ (G \mathrel{\dot\sqsubseteq} \bar{G})]k &= \bar{H}(Gk \sqsubseteq \bar{G}k) \circ (H(Gk) \sqsubseteq \bar{H}(Gk)) \\
&= (\bar{H}(Gk) \sqsubseteq \bar{H}(\bar{G}k)) \circ (H(Gk) \sqsubseteq \bar{H}(Gk)) \\
&= (H(Gk) \sqsubseteq \bar{H}(\bar{G}k)) \\
&= (H \circ G \mathrel{\dot\sqsubseteq} \bar{H} \circ \bar{G})k\,.
\end{aligned}$$

Then for ω-continuity on objects (functors):

$$\Big(\bigsqcup_n H_n \circ \bigsqcup_n G_n\Big)(k) = \bigsqcup_n H_n\Big(\bigsqcup_m G_m(k)\Big) = \bigsqcup_m \bigsqcup_n H_n(G_m(k)) = \bigsqcup_n (H_n \circ G_n(k))$$

for every object k, and

$$\Big(\bigsqcup_n H_n \circ \bigsqcup_n G_n\Big)(f) = \bigsqcup_n H_n\Big(\bigsqcup_m G_m(f)\Big) = \bigsqcup_m \bigsqcup_n H_n(G_m(f)) = \bigsqcup_n (H_n \circ G_n(f))$$

for every morphism f.

Of particular use will be the n-fold composition functor $\circ_n : \left(K^K\right)^n \longrightarrow K^K$ which by induction is continuous. (Zero-fold composition picks out the identity functor on K.) If we write Δ_n for the diagonal functor, defined for both objects and morphisms by $\Delta_n(x) = \langle x, \ldots, x\rangle$, with the result an n-tuple—this notation leaves the domain and codomain of Δ_n, in every particular use of it, to be inferred from context—then it follows that n-fold iteration,

$$\_^{(n)} \stackrel{\text{def}}{=} \circ_n \circ \Delta_n,$$

is, for each $n \geq 0$, a continuous endofunctor of K^K.

It has been noted that $I_K{\downarrow}I_K$ is a category-po when K is, and the reader may have surmised that the same holds with "category-cpo" replacing "category-po". Here is a proposition that gives a more general condition under which a comma category is a category-cpo. First, for any category-cpos L and M, any category K, and any functors $T : L \longrightarrow K$ and $S : M \longrightarrow K$, the comma category $T{\downarrow}S$ is a category-po, where we take the inequalities $(\langle l, m, f\rangle \sqsubseteq \langle \bar{l}, \bar{m}, \bar{f}\rangle)$ to be the morphisms of the form $\langle (l \sqsubseteq \bar{l}), (m \sqsubseteq \bar{m})\rangle$.

Proposition 2.8. *Let L and M be any category-cpos and let $T : L \longrightarrow K$ and $S : M \longrightarrow K$ any functors where T preserves l.u.b.s as colimits. Then the comma category $T{\downarrow}S$ is a category-cpo. Specifically, for any ω-chain $\left\langle\langle l_n, m_n, f_n\rangle\right\rangle_n$, and letting $l = \bigsqcup_n l_n$ and $m = \bigsqcup_n m_n$ for conciseness, the morphism f in the l.u.b. $\langle l, m, f\rangle$ is the u.m.m. in K from $\left\langle T(l_n \sqsubseteq l)\right\rangle_n$ to $\left\langle S(m_n \sqsubseteq m) \circ f_n\right\rangle_n$. Additionally,* *if S preserves l.u.b.s as colimits and f_n is an isomorphism for each n, then f is an isomorphism.*

Proof. Theorem A.4 gives that the forgetful functor $P : T{\downarrow}S \longrightarrow L \times M$ creates a colimit object $\langle l, m, f\rangle$ with colimit cone $\big\langle\langle(l_n \sqsubseteq l), (m_n \sqsubseteq m)\rangle\big\rangle_n$. Theorem A.4 also gives that f is the u.m.m. from $\big\langle T(l_n \sqsubseteq l)\big\rangle_n$ to $\big\langle S(m_n \sqsubseteq m) \circ f_n\big\rangle_n$ and that f inherits the isomorphism property when S preserves l.u.b.s as colimits.

To see that $\langle l, m, f\rangle$ is the l.u.b., let $\langle c, d, g\rangle$ be any upper bound of $\big\langle\langle l_n, m_n, f_n\rangle\big\rangle_n$. Since $\langle(l \sqsubseteq c), (m \sqsubseteq d)\rangle$ is the u.m.m. in $L \times M$, Fact A.1 gives that it is also the u.m.m. in $T{\downarrow}S$. That is, $\langle l, m, f\rangle \sqsubseteq \langle c, d, g\rangle$. □

Here is an analog to Fact 2.2 for comma category-pos.

Fact 2.9. *In any comma category-po $T{\downarrow}S$, if $\langle l, m, f\rangle \sqsubseteq \langle l, m, g\rangle$, then $f = g$.*

This fact may be seen by realizing that the morphism $(\langle l, m, f\rangle \sqsubseteq \langle l, m, g\rangle)$ is the identity $\langle 1_l, 1_m\rangle$. From Fact 2.9 follows also an analog for comma category-cpos of Fact 2.6: If an object of the form $\langle\bigsqcup_n l_n, \bigsqcup_n m_n, g\rangle$ is an upper bound of the chain $\big\langle\langle l_n, m_n, f_n\rangle\big\rangle_n$ in $T{\downarrow}S$, then it is the least upper bound.

The next proposition uses Lemma 2.4 to give a condition under which a functor whose codomain is a comma category-cpo is continuous.

Proposition 2.10. *Given a comma category-cpo $T{\downarrow}S$ where $T : L \longrightarrow K$ and $S : M \longrightarrow K$, let Q be any category-cpo and let $F : Q \longrightarrow T{\downarrow}S$ be any functor. If $P \circ F$ is continuous, where $P : T{\downarrow}S \longrightarrow L \times M$ is the forgetful functor, and T preserves l.u.b.s as colimits, then F is continuous.*

Proof. F preserves inequalities, because $P \circ F$ does and P does not modify morphisms. Let $q_0 \sqsubseteq q_1 \sqsubseteq \cdots$ be any ω-chain in Q. Since $P \circ F$ is continuous, $P(F(\bigsqcup_n q_n))$ is the vertex of a colimit cone $\big\langle(P(F(q_n)) \sqsubseteq P(F(\bigsqcup_n q_n)))\big\rangle_n$. Since, by Theorem A.4, P creates colimits, Corollary A.4.1 gives that $\big\langle(F(q_n) \sqsubseteq F(\bigsqcup_n q_n))\big\rangle_n$ is a colimit cone. Lemma 2.4 then gives that $F(\bigsqcup_n q_n) = \bigsqcup_n F(q_n)$. □

When K in Proposition 2.8 is a category-cpo, we may regenerate the partial order on morphisms by taking $T = S = i : \sqsubseteq \longrightarrow K$, the insertion functor. We chose to introduce $\sqsubseteq$ on morphisms beforehand in order to be able to partially order functors.

Section 3. One-sorted Tries.

We will need four particular category-cpos for our application to tries, three which we introduce now, and a comma category-cpo to be named later. The first is the category-cpo (trivially one, because it is a cpo) whose objects are sets and whose morphisms are only the inclusions between sets; we denote it by $\mathbf{Set}_{\subseteq}$. The second is $\langle\mathbf{Set}_*, \subseteq\rangle$, the category-cpo of pointed sets, with an inequality taken to be any set inclusion $A \subseteq \bar{A}$ (provided this actually is a morphism of $\mathbf{Set}_*$, that is, provided A and $\bar{A}$ have the same base point). We will invariably call this category-cpo simply $\mathbf{Set}_*$. The third is $\mathbf{Set}_*^{\mathbf{Set}_*}$, the category-cpo of continuous endofunctors of $\mathbf{Set}_*$ with their natural transformations; it is here that we hope to find the trie functor R. For now we identify one object of $\mathbf{Set}_*^{\mathbf{Set}_*}$: we denote by $\perp_{\bullet}$ the constant functor that maps every object of $\mathbf{Set}_*$ to $\{\bullet\}$ and every morphism to $1_{\{\bullet\}}$.

We define the mapping of sets

$$\mathcal{T}(Z) = \sum_{i=1}^{m} Z^{k_i},$$

so that our word algebra T is given by $T = \bigcup_n \mathcal{T}^{(n)}(\emptyset)$. The monotonicity and continuity with respect to $\subseteq$ of disjoint union (finitary or not) and finitary Cartesian product of sets are elementary facts—but for a proof of the latter, remove the spots in the proof of Proposition 3.3 below—hence these constructions are continuous endofunctors of $\mathbf{Set}_{\subseteq}$. It follows, by composition of functors, that $\mathcal{T} : \mathbf{Set}_{\subseteq} \longrightarrow \mathbf{Set}_{\subseteq}$ is a continuous functor.

We have to show that $\mathcal{F}$, defined in Section 1 as the mapping $Z \mapsto -^{[Z]}$, is a functor from $\mathbf{Set}_{\subseteq}$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$ which preserves l.u.b.s as colimits, that there is a continuous functor $\mathcal{R} : \mathbf{Set}_*^{\mathbf{Set}_*} \to \mathbf{Set}_*^{\mathbf{Set}_*}$ such that $R_n = \mathcal{R}^{(n)}(\perp_{\centerdot})$ and consequently $R = \dot{\bigcup}_n \mathcal{R}^{(n)}(\perp_{\centerdot})$, that for every $n \geq 0$ there is a natural isomorphism $\gamma_n : \mathcal{R}^{(n)}(\perp_{\centerdot}) \cong \mathcal{F}(\mathcal{T}^{(n)}(\emptyset))$, and finally that the γ_n are the morphism parts of an ascending ω-chain of objects in a suitable comma category-cpo. We may then conclude that the morphism part of the l.u.b. is a natural isomorphism

$$\gamma : R = \dot{\bigcup_n} \mathcal{R}^{(n)}(\perp_{\centerdot}) \cong \mathcal{F}(T),$$

and verify that γ satisfies the equations given for ap in Section 1.

The "finite functions" functor $\mathcal{F}$

For Z a fixed set, we may define what we will show is a continuous functor from $\mathbf{Set}_*$ to $\mathbf{Set}_*$ by the formulas, where A and B are any objects of $\mathbf{Set}_*$ and $h : A \longrightarrow B$ is any morphism, $A \mapsto A^{[Z]}$ (taking the base point of $A^{[Z]}$ to be $\lambda z.*_A$) and $h \mapsto (\lambda f.h \circ f)$, which we correspondingly denote by $h^{[Z]}$. We may see that this is a functor by noting that its action on morphisms is that of a covariant hom-functor, or we may verify in detail

$$\begin{aligned} {1_A}^{[Z]} &= \lambda f.1_A \circ f = \lambda f.f = 1_{A^{[Z]}} \\ (h \circ g)^{[Z]} &= \lambda f.h \circ g \circ f = (\lambda f.h \circ f) \circ (\lambda f.g \circ f) = h^{[Z]} \circ g^{[Z]}. \end{aligned}$$

For continuity, we verify that $-^{[Z]}$ is ω-continuous on objects, i.e. that

$$\Big(\bigcup_n A_n\Big)^{[Z]} = \bigcup_n {A_n}^{[Z]},$$

because first, all terms and the l.u.b. of any ω-chain $A_0 \subseteq A_1 \subseteq \ldots \bigcup_n A_n$ must share a common base point $*$ in order for the inclusions to be base-point-preserving; second, if $A \subseteq A'$, then an almost-everywhere-$*$ function from Z to A is also such a function from Z to A', making $-^{[Z]}$ monotone on objects, whence $\bigcup_n {A_n}^{[Z]} \subseteq \left(\bigcup_n A_n\right)^{[Z]}$, and third, if $f \in \left(\bigcup_n A_n\right)^{[Z]}$, then the finitely many non-$*$ values of f

must already lie in some A_n, whence $\left(\bigcup_n A_n\right)^{[Z]} \subseteq \bigcup_n A_n{}^{[Z]}$. To show that $-^{[Z]}$ preserves inclusions, calculate

$$(A \subseteq A')^{[Z]} = \lambda f.(A \subseteq A') \circ f = \lambda f \in A^{[Z]}.f = (A^{[Z]} \subseteq A'^{[Z]}).$$

This shows that the functor $-^{[Z]}$ is an object of $\mathbf{Set}_*^{\mathbf{Set}_*}$.

We next show that the mapping $Z \mapsto -^{[Z]}$ is the object part of a functor $\mathcal{F} : \mathbf{Set}_{\subseteq} \longrightarrow \mathbf{Set}_*{}^{\mathbf{Set}_*}$, and that $\mathcal{F}$ preserves l.u.b.s as colimits. Note that $\mathcal{F}$ preserves a l.u.b as a colimit, but not as a l.u.b. Also, restricting its domain to $\mathbf{Set}_{\subseteq}$ allows $\mathcal{F}$ to be covariant instead of contravariant.

We first introduce some compact notation. Let $\underline{Z} \subseteq Z \subseteq \bar{Z}$ be sets and let $f \in A^{[Z]}$ be any function. We denote by $f\downharpoonleft\underline{Z}$ the function $\lambda z \in \underline{Z}.f(z)$ in $A^{[\underline{Z}]}$, that is the restriction of f to $\underline{Z}$, and by $f\upharpoonleft\bar{Z}$ the function ($\lambda z \in \bar{Z}$. if $z \in Z$ then $f(z)$ else $*_A$) in $A^{[\bar{Z}]}$. The following facts are immediate:

$$f\upharpoonleft\bar{Z}\downharpoonleft\underline{Z} = f\downharpoonleft\underline{Z}; \tag{1}$$

$$\text{If } \{z \in Z \mid f(z) \neq *_A\} \subseteq \underline{Z}, \text{ then } f\downharpoonleft\underline{Z}\upharpoonleft Z = f. \tag{2}$$

For any $h : A \longrightarrow A'$ in $\mathbf{Set}_*$,

$$(h \circ f)\downharpoonleft\underline{Z} = h \circ (f\downharpoonleft\underline{Z}), \qquad \text{and} \tag{3a}$$

$$(h \circ f)\upharpoonleft\bar{Z} = h \circ (f\upharpoonleft\bar{Z}) \qquad [\text{because } h : *_A \mapsto *_{A'}]. \tag{3b}$$

Let $Z_0 \subseteq Z_1 \subseteq Z_2 \subseteq \ldots$ be any sequence in $\mathbf{Set}_{\subseteq}$. For any $f \in A^{[\bigcup_n Z_n]}$, let n_f denote the minimum index for which $f\downharpoonleft Z_{n_f}\upharpoonleft\bigcup_n Z_n = f$, that is, the minimum index such that $f(z) = *_A$ for all $z \in (\bigcup_n Z_n) - Z_{n_f}$. Then we may note the following facts:

$$\text{For any } \bar{n} \geq n_f, \qquad f\downharpoonleft Z_{n_f}\upharpoonleft Z_{\bar{n}} = f\downharpoonleft Z_{\bar{n}}, \tag{4}$$

and

$$\text{For any } h : A \longrightarrow A', \qquad n_{h \circ f} \leq n_f. \tag{5}$$

We may now define the functor $\mathcal{F} : \mathbf{Set}_{\subseteq} \longrightarrow \mathbf{Set}_*^{\mathbf{Set}_*}$ by

$$\mathcal{F}(Z) = -^{[Z]},$$
$$\mathcal{F}(Z \subseteq \bar{Z}) = A \mapsto \lambda f \in A^{[Z]}.f\upharpoonleft\bar{Z}.$$

To show that $\mathcal{F}(Z\subseteq\bar{Z})$ is a natural transformation is to show that the following diagram commutes for any $h : A \longrightarrow A'$:

$$\begin{tikzcd}
A^{[Z]} \arrow[r, "\lambda f.f\upharpoonleft\bar{Z}"] \arrow[d, "\lambda f.h\circ f"'] & A^{[\bar{Z}]} \arrow[d, "\lambda f.h\circ f"] \\
A'^{[Z]} \arrow[r, "\lambda f.f\upharpoonleft\bar{Z}"'] & A'^{[\bar{Z}]}.
\end{tikzcd}$$

But this is just Fact 3b: $(h \circ f){\upharpoonright}\bar{Z} = h \circ (f{\upharpoonright}\bar{Z})$, for each $f \in A^{[Z]}$.

We must check that $\mathcal{F}$ is a functor. It is immediate that $\mathcal{F}(Z{=}Z) = 1_{\_[Z]}$, the identity natural transformation. For a composite inclusion $Z \subseteq \bar{Z} \subseteq \bar{\bar{Z}}$ of sets, we have $\mathcal{F}(Z{\subseteq}\bar{\bar{Z}}) = \mathcal{F}(\bar{Z}{\subseteq}\bar{\bar{Z}}) \cdot \mathcal{F}(Z{\subseteq}\bar{Z})$ because $f{\upharpoonright}\bar{Z}{\upharpoonright}\bar{\bar{Z}} = f{\upharpoonright}\bar{\bar{Z}}$ for any $f \in A^{[Z]}$.

It remains to prove that $\mathcal{F}$ preserves any l.u.b. as a colimit. Let $Z_0 \subseteq Z_1 \subseteq \cdots$ be any ω-chain in $\mathbf{Set}_{\subseteq}$ with $Z \stackrel{\text{def}}{=} \bigcup_n Z_n$. Fact A.3 gives that $i^* : \mathbf{Set}_*^{\mathbf{Set}_*} \longrightarrow \mathbf{Set}_*^{|\mathbf{Set}_*|}$ creates colimits, and so by Lemma A.2 it will be sufficient to prove that $i^* \circ \mathcal{F}$ preserves Z as a colimit. That is, for each $A \in \mathbf{Set}_*$, we shall prove that $A^{[Z]}$ with cone $\langle \lambda f \in A^{[Z_n]}.f{\upharpoonright}Z \rangle_n$ is a colimiting cone on the base $A^{[Z_0]} \xrightarrow{\lambda f.f{\upharpoonright}Z_1} A^{[Z_1]} \xrightarrow{\lambda f.f{\upharpoonright}Z_2} \cdots$. To this end, let $\sigma = \langle \sigma_n : A^{[Z_n]} \longrightarrow X \rangle_n$ be any cone on the same base with vertex X. Let $\mu(\sigma)$ denote the morphism $\lambda f.\sigma_{n_f}(f{\downharpoonright}Z_{n_f}) : A^{[Z]} \longrightarrow X$. We claim that $\mu(\sigma)$ is the unique mediating morphism to the cone σ, showing that $A^{[Z]}$ and its cone are a colimit.

First we must show that $\mu(\sigma)$ mediates: For any n and any $f \in A^{[Z_n]}$, let n' denote $n_{f{\upharpoonright}Z}$. By its definition, $n' \leq n$. Since σ is a cone, the following equation holds for $f{\downharpoonright}Z_{n'}$:

$$\sigma_n(f{\downharpoonright}Z_{n'}{\upharpoonright}Z_n) = \sigma_{n'}(f{\downharpoonright}Z_{n'}). \tag{6}$$

By Fact 2, $f = f{\downharpoonright}Z_{n'}{\upharpoonright}Z_n$. By Fact 1, $f{\upharpoonright}Z{\downharpoonright}Z_{n'} = f{\downharpoonright}Z_{n'}$. Consequently, Equation 6 is the required mediating equation as follows:

$$\sigma_n(f) = \sigma_{n'}(f{\upharpoonright}Z{\downharpoonright}Z_{n'}). \tag{7}$$

Now to show uniqueness of $\mu(\sigma)$: Let $g : A^{[Z]} \to X$ be any mediating morphism. For any $f \in A^{[Z]}$, since g mediates and since $f = f{\downharpoonright}Z_{n_f}{\upharpoonright}Z$, the following equations hold:

$$g(f) = g(f{\downharpoonright}Z_{n_f}{\upharpoonright}Z) = (g \circ (\lambda f.f{\upharpoonright}Z))(f{\downharpoonright}Z_{n_f}) = \sigma_{n_f}(f{\downharpoonright}Z_{n_f}) = \mu(\sigma)(f), \tag{8}$$

that is, $g = \mu(\sigma)$.

It will be useful to spell out the effect of composite functors $\mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_k$ on morphisms of $\mathbf{Set}_*$ and of composite natural transformations $\mathcal{F}(Z_1{\subseteq}\bar{Z}_1) \circ \cdots \circ \mathcal{F}(Z_k{\subseteq}\bar{Z}_k)$ on objects. Restating the definition of $\mathcal{F}$ somewhat redundantly, we may write, for any one set $Z \in \mathbf{Set}_{\subseteq}$, for $h : A \longrightarrow B$ in $\mathbf{Set}_*$, for $f \in A^{[Z]}$, and for $z \in Z$,

$$\mathcal{F}Zhfz = h(fz).$$

This is the case $k = 1$ of the following generalization to composites:

Proposition 3.1. *For $Z_1, \ldots, Z_k \in \mathbf{Set}_{\subseteq}$, for $h : A \longrightarrow B$ in $\mathbf{Set}_*$, for $f \in A^{[Z_k]\cdots[Z_1]}$, and for $z_1 \in Z_1, \ldots z_k \in Z_k$,*

$$[\mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_k]hfz_1 \cdots z_k = h(fz_1 \cdots z_k). \tag{9}$$

Proof. By induction on k. For $k = 0$ we have $f \in A$, the composite functor is the identity $I_{\mathbf{Set}_*}$, and (9) is simply

$$I_{\mathbf{Set}_*}hf = hf.$$

For $k > 1$ we have

$$\begin{aligned}[\mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_k]hfz_1 \cdots z_k &= [\mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_{k-1}](\mathcal{F}Z_k h)fz_1 \cdots z_{k-1}z_k \\ &= \mathcal{F}Z_k h(fz_1 \cdots z_{k-1})z_k \\ &= h(fz_1 \cdots z_k). \quad \square\end{aligned}$$

Formula 9 should be familiar from the theory of combinators: it gives the effect of k-fold iteration of the composition combinator $\mathbf{B}$.

We may similarly write out the definition of $\mathcal{F}(Z{\subseteq}\bar{Z})$, for $Z \subseteq \bar{Z} \in \mathbf{Set}_{\subseteq}$, for $A \in \mathbf{Set}_*$, for $f \in A^{[Z]}$, and for $z \in \bar{Z}$:

$$\mathcal{F}(Z{\subseteq}\bar{Z})_A fz = (f{\upharpoonleft}\bar{Z})z.$$

Again, this is the case $k = 1$ of a generalization to composites:

Proposition 3.2. *For $Z_i \subseteq \bar{Z}_i$, $i = 1, \ldots, k$, for $A \in \mathbf{Set}_*$, for $f \in A^{[Z_k]\cdots[Z_1]}$, and for $z_1 \in Z_1, \ldots, z_k \in Z_k$,*

$$[\mathcal{F}(Z_1{\subseteq}\bar{Z}_1) \circ \cdots \circ \mathcal{F}(Z_k{\subseteq}\bar{Z}_k)]_A fz_1 \cdots z_k = (\cdots(f{\upharpoonleft}\bar{Z}_1)z_1 \cdots {\upharpoonleft}\bar{Z}_k)z_k.$$

Proof. For $k = 0$, whence $f \in A$, the composite natural transformation is the (horizontal) identity, $1_{\mathbf{Set}_*} : I_{\mathbf{Set}_*} \stackrel{\cdot}{\longrightarrow} I_{\mathbf{Set}_*}$, and the asserted equation is merely

$$(1_{\mathbf{Set}_*})_A f = 1_A f = f.$$

For $k > 1$, write as a shorthand S for $\mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_{k-1}$, $\bar{S}$ for $\mathcal{F}\bar{Z}_1 \circ \cdots \circ \mathcal{F}\bar{Z}_{k-1}$, and σ for $\mathcal{F}(Z_1{\subseteq}\bar{Z}_1) \circ \cdots \circ \mathcal{F}(Z_{k-1}{\subseteq}\bar{Z}_{k-1}) : S \stackrel{\cdot}{\longrightarrow} \bar{S}$. Then

$$\begin{aligned}[\sigma \circ \mathcal{F}(Z_k{\subseteq}\bar{Z}_k)]_A fz_1 \cdots z_k &= [\bar{S}(\mathcal{F}(Z_k{\subseteq}\bar{Z}_k)_A) \circ \sigma_{\mathcal{F}(Z_k)A}]fz_1 \cdots z_k \\ &= \bar{S}(\mathcal{F}(Z_k{\subseteq}\bar{Z}_k)_A)(\sigma_{\mathcal{F}(Z_k)A}f)z_1 \cdots z_k \\ &= \mathcal{F}(Z_k{\subseteq}\bar{Z}_k)_A(\sigma_{\mathcal{F}(Z_k)A}fz_1 \cdots z_{k-1})z_k \\ &= ((\sigma_{\mathcal{F}(Z_k)A}fz_1 \cdots z_{k-1}){\upharpoonleft}\bar{Z}_k)z_k \\ &= ((\cdots(f{\upharpoonleft}\bar{Z}_1)z_1 \cdots {\upharpoonleft}\bar{Z}_{k-1})z_{k-1}{\upharpoonleft}\bar{Z}_k)z_k\end{aligned}$$

respectively by definition of horizontal composition [7, p. 43 eqn. (2)], by definition of function composition, by Proposition 3.1, by definition of $\mathcal{F}(Z_k{\subseteq}\bar{Z}_k)_A$, and by induction. $\square$

Products and the functor $\mathcal{R}$

We recall from Section 1 the definition of the modified Cartesian product, $\prod\limits_{\bullet}^{m}$, of m sets with base points, for any $m \geq 0$:

$$\prod_{\bullet\, i=1}^{m} A_i = \prod_{i=1}^{m} A_i - \{\langle *_{A_i} \rangle_{i=1}^{m}\} \cup \{\bullet\}.$$

We take it to be so clear as not to need a written-out proof that $\prod^m$ and $\prod^m_\bullet$ are the object parts of two m-ary product functors on $\mathbf{Set}_*$, naturally isomorphic via the evident

$$\begin{aligned} -^\bullet &: \prod_{i=1}^m A_i \longrightarrow \prod_{\bullet\, i=1}^m A_i \\ &: \langle *_{A_i}\rangle_{i=1}^m \longmapsto \bullet \\ &: \langle a_i\rangle_{i=1}^m \longmapsto \langle a_i\rangle_{i=1}^m \qquad \text{otherwise,} \end{aligned}$$

and that the action of $\prod^m_\bullet$ on morphisms of $\mathbf{Set}_*^m$ is given by

$$\Big(\prod_{\bullet\, i=1}^m h_i\Big)\langle a_1,\ldots,a_m\rangle^\bullet = \langle h_1(a_1),\ldots,h_m(a_m)\rangle^\bullet .$$

Note that $\prod_{\bullet\, i=1}^m h_i$ is a base-point-preserving map, as is $\prod_{i=1}^m h_i$ (same definition without the spots); also that we have defined exactly one value of $\prod_{\bullet\, i=1}^m h_i$ for every element of $\prod_{\bullet\, i=1}^m A_i$, because $-^\bullet$ is a one-to-one correspondence. We will similarly take the liberty of writing λ-abstractions, when convenient, in the form $\lambda\langle a_1,\ldots,a_m\rangle^\bullet.\ \ldots$.

We give the proof of continuity of $\prod^m_\bullet$ as a reminder that the restriction to finitary products is essential:

Proposition 3.3. $\prod^m_\bullet : \mathbf{Set}_*^m \longrightarrow \mathbf{Set}_*$ *is a continuous functor.*

Proof. We prove first ω-continuity on objects. For $i = 1,\ldots,m$, let $A_{i0} \subseteq A_{i1} \subseteq \cdots$ be an ascending ω-chain in $\mathbf{Set}_*$ (necessarily sharing a common base point). Then

$$\begin{aligned} \prod_{\bullet\, i=1}^m \bigcup_n A_{in} &= \{\langle a_1,\ldots,a_m\rangle^\bullet \mid a_i \in \bigcup_n A_{in},\ i=1,\ldots,m\} \\ &= \{\langle a_1,\ldots,a_m\rangle^\bullet \mid \exists n_1\cdots\exists n_m\,.\, a_1 \in A_{1n_1} \wedge \cdots \wedge a_m \in A_{mk_m}\} \\ &= \{\langle a_1,\ldots,a_m\rangle^\bullet \mid \exists n\,.\, a_i \in A_{in},\ i=1,\ldots,m\} \\ &= \bigcup_n \prod_{\bullet\, i=1}^m A_{in}. \end{aligned}$$

Then, to show preservation of inequalities, suppose that $A_i \subseteq \bar{A}_i$, $i = 1,\ldots,m$, and recall that as a function, an inclusion sends any element of A_i to itself as an element of $\bar{A}_i$; hence we may calculate

$$\prod_{\bullet\, i=1}^m (A_i \subseteq \bar{A}_i) = \lambda\langle a_1,\ldots,a_m\rangle^\bullet.\langle a_1,\ldots,a_m\rangle^\bullet = \Big(\prod_{\bullet\, i=1}^m A_i \subseteq \prod_{\bullet\, i=1}^m \bar{A}_i\Big). \quad \square$$

The finite products in $\mathbf{Set}_*$ given by $\prod^m_\bullet$ yield—see, for example, [7, III.5 ex. 5]—a "pointwise" product of any m functors $G_1, \ldots, G_m \in \mathbf{Set}_*^{\mathbf{Set}_*}$ which we denote by $\dot{\prod\limits_{i=1}^{m}}_{\bullet} G_i$:

$$\Big(\dot{\prod_{i=1}^{m}}_{\bullet} G_i\Big)A = \prod_{i=1}^{m}{}_{\bullet}(G_iA)\,,$$
$$\Big(\dot{\prod_{i=1}^{m}}_{\bullet} G_i\Big)h = \prod_{i=1}^{m}{}_{\bullet}(G_ih)\,.$$

Moreover, it is immediate that $\dot{\prod\limits_{i=1}^{m}}_{\bullet} G_i$ is continuous because $\prod\limits^{m}_{\bullet}$ is and the G_i are; thus $\mathbf{Set}_*^{\mathbf{Set}_*}$ has finite products.

Now it follows, by [7, III.5 Proposition 1], that $\dot{\prod\limits^{m}}_{\bullet}$ is (the object function of) an m-ary product functor in $\mathbf{Set}_*^{\mathbf{Set}_*}$; the arrow function also works "pointwise": if $\tau_i : G_i \stackrel{\cdot}{\longrightarrow} H_i$, $i = 1, \ldots, m$, then

$$\dot{\prod_{i=1}^{m}}_{\bullet} \tau_i A = \prod_{i=1}^{m}{}_{\bullet}(\tau_i A) \qquad \text{for each } A \in \mathbf{Set}_*.$$

Finally as to products, we have

Proposition 3.4. $\dot{\prod\limits^{m}}_{\bullet} : \big(\mathbf{Set}_*^{\mathbf{Set}_*}\big)^m \longrightarrow \mathbf{Set}_*^{\mathbf{Set}_*}$ *is continuous.*

Proof. Continuity on objects: For $i = 1, \ldots, m$, let $G_{i0} \mathrel{\dot{\subseteq}} G_{i1} \mathrel{\dot{\subseteq}} \cdots$ be an ω-chain in $\mathbf{Set}_*^{\mathbf{Set}_*}$. Then for each object A of $\mathbf{Set}_*$,

$$\begin{aligned}\Big(\dot{\prod_{i=1}^{m}}_{\bullet} \dot{\bigcup_{n}} G_{in}\Big)A &= \prod_{i=1}^{m}{}_{\bullet}\Big(\dot{\bigcup_{n}} G_{in}\Big)A = \prod_{i=1}^{m}{}_{\bullet}\Big(\bigcup_{n} G_{in}A\Big)\\ &= \bigcup_{n}\prod_{i=1}^{m}{}_{\bullet} G_{in}A = \bigcup_{n}\Big(\dot{\prod_{i=1}^{m}}_{\bullet} G_{in}A\Big) = \Big(\dot{\bigcup_{n}}\dot{\prod_{i=1}^{m}}_{\bullet} G_{in}\Big)A,\end{aligned}$$

and for each morphism $h : A \longrightarrow B$ of $\mathbf{Set}_*$, we may make an identical calculation with h replacing A throughout. (For the central equality, recall that $\prod\limits^{m}_{\bullet}$, as a continuous functor, preserves l.u.b.s of ω-chains of morphisms as well as of objects.)

Preservation of inequalities: If $G_i \mathrel{\dot{\subseteq}} \bar{G}_i$, $i = 1, \ldots, m$, then for each object $A \in \mathbf{Set}_*$,

$$\begin{aligned}\Big(\dot{\prod_{i=1}^{m}}_{\bullet}(G_i \mathrel{\dot{\subseteq}} \bar{G}_i)\Big)A &= \prod_{i=1}^{m}{}_{\bullet}(G_i \mathrel{\dot{\subseteq}} \bar{G}_i)A = \prod_{i=1}^{m}{}_{\bullet}(G_iA \subseteq \bar{G}_iA)\\ &= \Big(\prod_{i=1}^{m}{}_{\bullet} G_iA \subseteq \prod_{i=1}^{m}{}_{\bullet} \bar{G}_iA\Big) = \Big(\Big(\dot{\prod_{i=1}^{m}}_{\bullet} G_i\Big)A \subseteq \Big(\dot{\prod_{i=1}^{m}}_{\bullet} \bar{G}_i\Big)A\Big) = \Big(\dot{\prod_{i=1}^{m}}_{\bullet} G_i \mathrel{\dot{\subseteq}} \dot{\prod_{i=1}^{m}}_{\bullet} \bar{G}_i\Big)A. \quad \square\end{aligned}$$

We are now in a position to observe that the functor $\mathcal{R} : \mathbf{Set}_*^{\mathbf{Set}_*} \longrightarrow \mathbf{Set}_*^{\mathbf{Set}_*}$, although most perspicuously defined pointwise:

$$(\mathcal{R}G)A = \prod_{i=1}^{m}{}_{\bullet}\, G^{(k_i)}A, \qquad (\mathcal{R}\tau)A = \prod_{i=1}^{m}{}_{\bullet}\, \tau^{(k_i)}A,$$
$$(\mathcal{R}G)h = \prod_{i=1}^{m}{}_{\bullet}\, G^{(k_i)}h;$$

may in fact be built by composition from the continuous functors $\dot{\prod}^{m}_{\bullet}$ and k_i-fold iteration:

$$\mathcal{R}G = \dot{\prod_{i=1}^{m}}{}_{\bullet}\, G^{(k_i)} \qquad \mathcal{R}\tau = \dot{\prod_{i=1}^{m}}{}_{\bullet}\, \tau^{(k_i)}$$

and is therefore itself continuous.

Laws of exponents as natural isomorphisms

We turn next to the reinforcement of Propositions 1.1 and 1.3 with categorical sinews.

Proposition 3.5. *The one-to-one correspondence* $\mu^\bullet_m : \prod_{i=1}^{m}{}_{\bullet}\, A^{[X_i]} \cong A^{[\sum_{i=1}^{m} X_i]}$ *of Definition 1.2 is natural in* A *(regarded as an object of* $\mathbf{Set}_*$*) and in* $X_1, \ldots, X_m$ *(regarded as objects of* $\mathbf{Set}_\subseteq$*).*

Proof. We show naturality in A first; that is, for fixed sets $X_1, \ldots, X_m$ and any morphism $h : A \to B$ in $\mathbf{Set}_*$, that the diagram

$$\begin{array}{ccc}
\prod_{i=1}^{m}{}_{\bullet}\, A^{[X_i]} & \xrightarrow{\mu^\bullet_m} & A^{[\sum_{i=1}^{m} X_i]} \\
{\scriptstyle \prod_{i=1}^{m}{}_{\bullet}\, h^{[X_i]}}\downarrow & & \downarrow{\scriptstyle h^{[\sum_{i=1}^{m} X_i]}} \\
\prod_{i=1}^{m}{}_{\bullet}\, B^{[X_i]} & \xrightarrow{\mu^\bullet_m} & B^{[\sum_{i=1}^{m} X_i]}
\end{array}$$

commutes. (For compactness, we shall omit writing object arguments such as A and $\langle X_1, \ldots, X_m\rangle$ for $\mu^\bullet_m$).

We may write out the definition of $\mu^\bullet_m : \prod_{i=1}^{m}{}_{\bullet}\, A^{[X_i]} \longrightarrow A^{[\sum_{i=1}^{m} X_i]}$ explicitly:

$$\mu^\bullet_m \langle f_1, \ldots, f_m\rangle^\bullet = \lambda\langle i, x\rangle . f_i x\,.$$

Then, for any $g_1 \in A^{[X_1]}, \ldots, g_m \in A^{[X_m]}$,

$$\begin{aligned}
\big[\mu^\bullet_m \circ \prod_{i=1}^{m}{}_{\bullet}\, h^{[X_i]}\big]\langle g_1, \ldots, g_m\rangle^\bullet &= \mu^\bullet_m\langle h\circ g_1, \ldots, h\circ g_m\rangle^\bullet \\
&= \lambda\langle i, x\rangle . [h \circ g_i](x),
\end{aligned}$$

and

$$\begin{aligned}
\big[h^{[\sum_{i=1}^{m} X_i]} \circ \mu^\bullet_m\big]\langle g_1, \ldots, g_m\rangle^\bullet &= h^{[\sum_{i=1}^{m} X_i]}(\lambda\langle i, x\rangle . g_i x) \\
&= h \circ (\lambda\langle i, x\rangle . g_i x),
\end{aligned}$$

which is the same function.

This has shown naturality in A, that is, for fixed $X_1, \ldots, X_m$, that

$$\mu^\bullet_m : \prod_{i=1}^{m}{}_\bullet\, \mathcal{F}(X_i) \stackrel{\cdot}{\longrightarrow} \mathcal{F}(\sum_{i=1}^{m} X_i).$$

Now for naturality in $X_1, \ldots, X_m$ we need, supposing $X_i \subseteq \bar{X}_i$ for $i = 1, \ldots, m$, commutativity of

$$\begin{array}{ccc} \prod_{i=1}^{m}{}_\bullet\, A^{[X_i]} & \xrightarrow{\mu^\bullet_m} & A^{[\sum_{i=1}^m X_i]} \\ {\scriptstyle \prod_{i=1}^{m}{}_\bullet\, \lambda f.f\upharpoonright\bar{X}_i}\downarrow & & \downarrow{\scriptstyle \lambda f.f\upharpoonright \sum_{i=1}^{m} \bar{X}_i} \\ \prod_{i=1}^{m}{}_\bullet\, A^{[\bar{X}_i]} & \xrightarrow{\mu^\bullet_m} & A^{[\sum_{i=1}^m \bar{X}_i]}\,. \end{array}$$

For $g_i \in A^{[X_i]}$, $i = 1, \ldots, m$, we find

$$\begin{aligned} \big[\mu^\bullet_m \circ \prod_{i=1}^{m}{}_\bullet\, \lambda f.f\upharpoonright\bar{X}_i\big]\langle g_1, \ldots, g_m\rangle^\bullet &= \mu^\bullet_m\langle g_1\upharpoonright\bar{X}_1, \ldots, g_m\upharpoonright\bar{X}_m\rangle^\bullet = \lambda\langle i, x\rangle.(g_i\upharpoonright\bar{X}_i)x \\ &= (\lambda\langle i, x\rangle.g_i x)\upharpoonright \textstyle\sum_{i=1}^m \bar{X}_i = \big[(\lambda f.f\upharpoonright \sum_{i=1}^m \bar{X}_i) \circ \mu^\bullet_m\big]\langle g_1, \ldots, g_m\rangle^\bullet. \quad \square \end{aligned}$$

Proposition 3.6. *The one-to-one correspondence $\nu_k : A^{[Z_k]\cdots[Z_1]} \cong A^{[\prod_{i=1}^k Z_i]}$ of Proposition 1.3 is natural in A and in $Z_1, \ldots, Z_k$ for any $k \geq 0$; that is, ν_k is a natural isomorphism of functors from $\mathbf{Set}^k_\subseteq$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$.*

Proof. Recall that for $g \in A^{[Z_k]\cdots[Z_1]}$ we have

$$\nu_k(g) = \lambda\langle z_1, \ldots, z_k\rangle.g z_1 \cdots z_k\,.$$

We first show naturality in A, that is, that for fixed $Z_1, \ldots, Z_k \in \mathbf{Set}_\subseteq$, and $h : A \longrightarrow B$ in $\mathbf{Set}_*$, the diagram

$$\begin{array}{ccc} A^{[Z_k]\cdots[Z_1]} & \xrightarrow{\nu_k} & A^{[Z_1\times\cdots\times Z_k]} \\ {\scriptstyle \lambda f\lambda z_1\ldots\lambda z_k.h(f\,z_1\cdots z_k)}\downarrow & & \downarrow{\scriptstyle \lambda f\lambda z.h(f\,z)} \\ B^{[Z_k]\cdots[Z_1]} & \xrightarrow{\nu_k} & B^{[Z_1\times\cdots\times Z_k]} \end{array}$$

commutes. (As with $\mu^\bullet_m$ in Proposition 3.5, we have not written object parameters for ν_k. The label on the left side arrow comes from Proposition 3.1.) For any $f \in A^{[Z_k]\cdots[Z_1]}$, this works out to the true equation

$$\lambda\langle z_1, \ldots, z_k\rangle.h(f\,z_1\cdots z_k) = h \circ (\lambda\langle z_1, \ldots, z_k\rangle.f\,z_1\cdots z_k).$$

Thus $\nu_k : \mathcal{F}Z_1 \circ \cdots \circ \mathcal{F}Z_k \stackrel{\cdot}{\longrightarrow} \mathcal{F}(Z_1 \times \cdots \times Z_k)$ is natural.

Now for naturality in $Z_1, \ldots, Z_k$, suppose $Z_i \subseteq \bar{Z}_i$ for $i = 1, \ldots, k$, and for any $A \in \mathbf{Set}_*$ consider the diagram (the label on the left side arrow comes from Proposition 3.2):

$$\begin{array}{ccc} A^{[Z_k]\cdots[Z_1]} & \xrightarrow{\nu_k} & A^{[Z_1\times\cdots\times Z_k]} \\ {\scriptstyle \lambda f \lambda z_1 \ldots \lambda z_k.(\cdots(f\upharpoonright\bar{Z}_1)z_1\cdots\upharpoonright\bar{Z}_k)z_k}\Big\downarrow & & \Big\downarrow{\scriptstyle \lambda g. g\upharpoonright(\bar{Z}_1\times\cdots\times\bar{Z}_k)} \\ A^{[\bar{Z}_k]\cdots[\bar{Z}_1]} & \xrightarrow{\nu_k} & A^{[\bar{Z}_1\times\cdots\times\bar{Z}_k]}\,. \end{array}$$

To show commutativity of this diagram is to show, for any $f \in A^{[Z_k]\cdots[Z_1]}$, that

$$\lambda\langle z_1, \ldots, z_k\rangle.(\cdots(f\upharpoonright\bar{Z}_1)z_1\cdots\upharpoonright\bar{Z}_k)z_k = (\lambda\langle z_1, \ldots, z_k\rangle.f\, z_1\cdots z_k)\upharpoonright(\bar{Z}_1\times\cdots\times\bar{Z}_k)\,.$$

For any k-tuple $\langle z_1, \ldots, z_k\rangle \in Z_1 \times \cdots \times Z_k$, both sides of this equation yield $f\, z_1 \cdots z_k$. On the other hand, if $\langle z_1, \ldots, z_k\rangle \in \bar{Z}_1\times\cdots\times\bar{Z}_k - Z_1\times\cdots\times Z_k$, then the right-hand side of the equation yields $*_A$. Let j be the smallest index such that $z_j \in \bar{Z}_j - Z_j$; then the left-hand side also is

$$\begin{aligned}&(\cdots((*_{A^{[Z_k]\cdots[Z_{j+1}]}})\upharpoonright\bar{Z}_{j+1})z_{j+1}\cdots\upharpoonright\bar{Z}_k)z_k \\ &\qquad = (\cdots((\lambda z_{j+1}\ldots\lambda z_k.*_A)\upharpoonright\bar{Z}_{j+1})z_{j+1}\cdots\upharpoonright\bar{Z}_k)z_k \\ &\qquad = *_A. \quad \square\end{aligned}$$

Combining the results of Proposition 3.5 and Proposition 3.6, we obtain:

Proposition 3.7. *The mapping $\psi_{k_1\cdots k_m}$, given for $g_i \in A^{[Z_{ik_i}]\cdots[Z_{i1}]}$ $(i = 1, \ldots, m)$ by*

$$\psi_{k_1\cdots k_m}\langle g_1, \ldots, g_m\rangle^\bullet = \lambda\langle i, \langle z_1, \ldots, z_{k_i}\rangle\rangle.g_i\, z_1\cdots z_{k_i}\,,$$

is an isomorphism

$$\psi_{k_1\cdots k_m} : \prod_{i=1}^{m}{}_{\bullet}\, A^{[Z_{ik_i}]\cdots[Z_{i1}]} \cong A^{[\sum_{i=1}^m \prod_{j=1}^{k_i} Z_{ij}]}$$

natural in the $k_1 + \cdots + k_m$ sets Z_{ij} and in A.

Proof. When natural isomorphism $\mu_m^\bullet$ (Proposition 3.5) is written without parameters, we have the following explicit expression giving functors from $\mathbf{Set}_\subseteq^m$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$ as its domain and codomain:

$$\mu_m^\bullet : \dot{\prod_{\bullet}^{m}} \circ \mathcal{F}^m \cong \mathcal{F} \circ \sum^{m}\,.$$

Here $\mathcal{F}^m$ denotes $\mathcal{F} \times \cdots \times \mathcal{F}$ with m factors, and $\sum^m$ denotes the m-ary coproduct functor in $\mathbf{Set}_\subseteq$. Consequently, writing $\prod^{k_i}$ for k_i-ary Cartesian product in $\mathbf{Set}_\subseteq$,

$$\mu_m^\bullet \circ (\prod^{k_1} \times \cdots \times \prod^{k_m}) : \dot{\prod_{\bullet}^{m}} \circ \mathcal{F}^m \circ (\prod^{k_1} \times \cdots \times \prod^{k_m}) \cong \mathcal{F} \circ \sum^{m} \circ (\prod^{k_1} \times \cdots \times \prod^{k_m})$$

is a natural isomorphism of functors from $\mathbf{Set}_{\subseteq}^{k_1} \times \cdots \times \mathbf{Set}_{\subseteq}^{k_m}$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$.

Similarly Proposition 3.6 gives, for $i = 1, \ldots, m$,

$$\nu_{k_i} : (\circ_{k_i}) \circ \mathcal{F}^{k_i} \cong \mathcal{F} \circ \prod^{k_i},$$

displaying two functors from $\mathbf{Set}_{\subseteq}^{k_i}$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$ as explicit domain and codomain for ν_{k_i}, and so we have by composition and m-ary product the natural isomorphism

$$\dot{\prod_{\bullet}^{m}} \circ (\nu_{k_1} \times \cdots \times \nu_{k_m}) : \dot{\prod_{\bullet}^{m}} \circ \left(\circ_{k_1} \circ \mathcal{F}^{k_1} \times \cdots \times \circ_{k_m} \circ \mathcal{F}^{k_m}\right) \cong \dot{\prod_{\bullet}^{m}} \circ \left(\mathcal{F} \circ \prod^{k_1} \times \cdots \times \mathcal{F} \circ \prod^{k_m}\right)$$

again of functors from $\mathbf{Set}_{\subseteq}^{k_1} \times \cdots \times \mathbf{Set}_{\subseteq}^{k_m}$ to $\mathbf{Set}_*^{\mathbf{Set}_*}$. Here the natural isomorphism $\nu_{k_1} \times \cdots \times \nu_{k_m}$ is defined to be the componentwise mapping

$$\langle Y_1, \ldots, Y_m \rangle \mapsto \langle \nu_{k_1} Y_1, \ldots, \nu_{k_m} Y_m \rangle$$

where each Y_i is a k_i-tuple of sets.

But

$$\mathcal{F}^m \circ \left(\prod^{k_1} \times \cdots \times \prod^{k_m}\right) = \left(\mathcal{F} \circ \prod^{k_1} \times \cdots \times \mathcal{F} \circ \prod^{k_m}\right),$$

so the above two natural isomorphisms are composable, yielding

$$\begin{aligned} (10) \quad \psi_{k_1 \cdots k_m} &\stackrel{\text{def}}{=} \left(\mu_m^{\bullet} \circ \left(\prod^{k_1} \times \cdots \times \prod^{k_m}\right)\right) \cdot \left(\dot{\prod_{\bullet}^{m}} \circ (\nu_{k_1} \times \cdots \times \nu_{k_m})\right) \\ &: \dot{\prod_{\bullet}^{m}} \circ \left(\circ_{k_1} \circ \mathcal{F}^{k_1} \times \cdots \times \circ_{k_m} \circ \mathcal{F}^{k_m}\right) \cong \mathcal{F} \circ \sum^{m} \circ \left(\prod^{k_1} \times \cdots \times \prod^{k_m}\right). \end{aligned}$$

Reverting to consideration of arbitrary objects Z_{ij}, $j = 1, \ldots, k_i$, $i = 1, \ldots, m$ of $\mathbf{Set}_{\subseteq}$ and A of $\mathbf{Set}_*$, we may work out the effect of $\psi_{k_1 \cdots k_m}\left(\langle\langle Z_{ij}\rangle_{j=1}^{k_i}\rangle_{i=1}^{m}\right)_A$ on any element $\langle g_1, \ldots, g_m\rangle^{\bullet}$ of $\dot{\prod}_{\bullet\, i=1}^{m} A^{[Z_{ik_i}]\cdots[Z_{i1}]}$ (as usual we suppress the object arguments):

$$\begin{aligned} \psi_{k_1 \cdots k_m} \langle g_1, \ldots, g_m \rangle^{\bullet} &= \mu_m^{\bullet} \langle \nu_{k_1}\, g_1, \ldots, \nu_{k_m}\, g_m \rangle^{\bullet} \\ &= \mu_m^{\bullet} \langle \lambda\langle z_1, \ldots, z_{k_1}\rangle . g_1\, z_1 \cdots z_{k_1}, \ \ldots, \ \lambda\langle z_1, \ldots, z_{k_m}\rangle . g_m\, z_1 \cdots z_{k_m} \rangle^{\bullet} \\ &= \lambda\langle i, \langle z_1, \ldots, z_{k_i}\rangle\rangle . g_i\, z_1 \cdots z_{k_i}. \quad \square \end{aligned}$$

Construction of γ as a least fixed point

In developing $\psi_{k_1 \cdots k_m}$, we deliberately made provision for $k_1 + \cdots + k_m$ separate sets Z_{ij}; this was with an eye to facilitating the treatment of many-sorted term algebras in Section 4. Our need at the moment, however, is to come down to one set Z. We note the following fact about diagonal functors. For any functor F and any $k \geq 0$,

$$(11) \qquad F^k \circ \Delta_k = \Delta_k \circ F.$$

Now observe that we may express our term-algebra functor $\mathcal{T} : \mathbf{Set}_{\subseteq} \longrightarrow \mathbf{Set}_{\subseteq}$ as

$$\mathcal{T} = \sum^{m} \circ \left(\prod^{k_1} \times \cdots \times \prod^{k_m}\right) \circ (\Delta_{k_1} \times \cdots \times \Delta_{k_m}) \circ \Delta_m$$

and the trie-building functor $\mathcal{R} : \mathbf{Set}_*^{\mathbf{Set}_*} \longrightarrow \mathbf{Set}_*^{\mathbf{Set}_*}$ as

$$\mathcal{R} = \dot{\prod_{\bullet}}^{m} \circ (\circ_{k_1} \times \cdots \times \circ_{k_m}) \circ (\Delta_{k_1} \times \cdots \times \Delta_{k_m}) \circ \Delta_m .$$

Using Equation 11, we may express the composite $\mathcal{R} \circ \mathcal{F}$ as

$$\begin{aligned}\mathcal{R} \circ \mathcal{F} &= \dot{\prod_{\bullet}}^{m} \circ (\circ_{k_1} \times \cdots \times \circ_{k_m}) \circ (\mathcal{F}^{k_1} \times \cdots \times \mathcal{F}^{k_m}) \circ (\Delta_{k_1} \times \cdots \times \Delta_{k_m}) \circ \Delta_m \\ &= \dot{\prod_{\bullet}}^{m} \circ ((\circ_{k_1} \circ \mathcal{F}^{k_1}) \times \cdots \times (\circ_{k_m} \circ \mathcal{F}^{k_m})) \circ (\Delta_{k_1} \times \cdots \times \Delta_{k_m}) \circ \Delta_m .\end{aligned}$$

If we define the composite

$$\Psi \stackrel{\text{def}}{=} \psi_{k_1 \cdots k_m} \circ (\Delta_{k_1} \times \cdots \times \Delta_{k_m}) \circ \Delta_m ,$$

then Proposition 3.7 and the explicit expression for the type of $\psi_{k_1 \cdots k_m}$ given by Formula 10 yield

Proposition 3.8. *We have the natural isomorphism of functors from* $\mathbf{Set}_{\subseteq}$ *to* $\mathbf{Set}_*^{\mathbf{Set}_*}$

$$\Psi : \mathcal{R} \circ \mathcal{F} \cong \mathcal{F} \circ \mathcal{T} .$$

For $Z \in \mathbf{Set}_{\subseteq}$ *and* $A \in \mathbf{Set}_*$, *the effect of* $\Psi_{ZA} : \dot{\prod_{\bullet}}_{i=1}^{m} \mathcal{F}_Z^{(k_i)}(A) \cong \mathcal{F}_{\sum_{i=1}^{m} Z^{k_i}}(A)$ *is given by*

$$\Psi_{ZA}\langle g_1, \ldots, g_m\rangle^{\bullet} = \lambda\langle i, \langle z_i, \ldots, z_{k_i}\rangle\rangle . g_i\, z_1 \cdots z_{k_i} . \quad \Box$$

We may now begin the construction of $\gamma : R \cong \mathcal{F}(T)$. For brevity write $\mathcal{I}$ for the identity functor $I_{\mathbf{Set}_*^{\mathbf{Set}_*}}$. Define what we will show in a moment is a continuous endofunctor of $\mathcal{I}{\downarrow}\mathcal{F}$ by

$$\begin{aligned}&\mathcal{C} : \mathcal{I}{\downarrow}\mathcal{F} \longrightarrow \mathcal{I}{\downarrow}\mathcal{F} \\ &\quad : \langle G, Z, \tau : G \stackrel{\cdot}{\longrightarrow} \mathcal{F}Z\rangle \longmapsto \langle \mathcal{R}(G), \mathcal{T}(Z), \Psi_Z \cdot \mathcal{R}(\tau)\rangle \\ &\quad : \langle \sigma : G \stackrel{\cdot}{\longrightarrow} G', (Z \subseteq \bar{Z})\rangle \longmapsto \langle \mathcal{R}(\sigma), (\mathcal{T}(Z) \subseteq \mathcal{T}(\bar{Z}))\rangle .\end{aligned}$$

Note that $\Psi_Z \cdot \mathcal{R}(\tau)$ is an isomorphism if τ is.

By Lemma A.5, $\mathcal{C}$ is a well-defined functor, taking $D = \mathcal{I}$, $E = \mathcal{F}$, $G = R = \mathcal{R}$, $\rho = 1_{\mathcal{R}}$, $H = \mathcal{T}$, and $\sigma = \Psi$ in the statement of the lemma.

Recall (from Proposition 2.10) the forgetful functor $P : \mathcal{I}{\downarrow}\mathcal{F} \longrightarrow \mathbf{Set}_*^{\mathbf{Set}_*} \times \mathbf{Set}_{\subseteq}$. Since $\mathcal{R}$ and $\mathcal{T}$ are continuous, $P \circ \mathcal{C} = \mathcal{R} \times \mathcal{T}$ is continuous. Consequently, Proposition 2.10 gives that $\mathcal{C}$ is continuous.

Now we will construct a least fixed point $\langle R, T, \gamma\rangle$ of $\mathcal{C}$. The functor $\perp_{\bullet}$ is initial in $\mathbf{Set}_*^{\mathbf{Set}_*}$: for any $G \in \mathbf{Set}_*^{\mathbf{Set}_*}$ we have the unique

$$(A \longmapsto (\bullet \mapsto *_{G(A)})) : \perp_{\bullet} \stackrel{\cdot}{\longrightarrow} G .$$

$\mathcal{F}(\emptyset)$ is also initial, as shown by the natural isomorphism

$$\gamma_0 \stackrel{\text{def}}{=} (A \longmapsto (\bullet \mapsto \lambda x \in \emptyset \,.\, *_A)) : \bot_{\bullet} \cong \mathcal{F}(\emptyset)\,.$$

By initiality, and the fact that every $\mathcal{R}(G)(A)$ is a spot product, and so contains $\bullet$, we have the commutative diagram

$$\text{(12)} \qquad \begin{array}{ccc} \bot_{\bullet} & \xrightarrow{\dot{\subseteq}} & \mathcal{R}(\bot_{\bullet}) \\ {\scriptstyle\gamma_0}\big\downarrow & & \big\downarrow{\scriptstyle\Psi_{\emptyset}\cdot\mathcal{R}(\gamma_0)} \\ \mathcal{F}(\emptyset) & \xrightarrow[\mathcal{F}(\emptyset\subseteq\mathcal{T}(\emptyset))]{} & \mathcal{F}(\mathcal{T}(\emptyset))\,. \end{array}$$

Let $R_0 = \bot_{\bullet}$ and $T_0 = \emptyset$. For brevity, write $\sqsubseteq$ for the inequalities $\langle\dot{\subseteq}, \subseteq\rangle$ of $\mathcal{I}{\downarrow}\mathcal{F}$. Then (12) is

$$\langle R_0, T_0, \gamma_0\rangle \sqsubseteq \mathcal{C}\langle R_0, T_0, \gamma_0\rangle$$

drawn as a diagram in $\mathbf{Set}_*^{\mathbf{Set}_*}$. It allows us to generate the ascending ω-chain in $\mathcal{I}{\downarrow}\mathcal{F}$:

$$\text{(13)} \qquad \langle R_0, T_0, \gamma_0\rangle \sqsubseteq \langle R_1, T_1, \gamma_1\rangle \sqsubseteq \langle R_2, T_2, \gamma_2\rangle \sqsubseteq \cdots$$

where $\langle R_n, T_n, \gamma_n\rangle = \mathcal{C}^{(n)}\langle\bot_{\bullet}, \emptyset, \gamma_0\rangle$ or, in detail,

$$R_{n+1} = \mathcal{R}(R_n), \qquad T_{n+1} = \mathcal{T}(T_n), \qquad \gamma_{n+1} = \Psi_{T_n} \cdot \mathcal{R}(\gamma_n)\,.$$

Then Proposition 2.8 gives that (13) has a l.u.b. $\langle R, T, \gamma\rangle$ where $R = \dot{\bigcup}_n R_n$, $T = \bigcup_n T_n$, and (since, by induction, each γ_n is an isomorphism) $\gamma : R \cong \mathcal{F}(T)$.

Considering $\mathcal{C}$ merely as an ω-continuous mapping on the objects of $\mathcal{I}{\downarrow}\mathcal{F}$, we have found

$$\langle R, T, \gamma\rangle = \bigsqcup_n \mathcal{C}^{(n)}\langle\bot_{\bullet}, \emptyset, \gamma_0\rangle\,.$$

We pause to note the following fact about cpos, useful for finding least fixed points when there is no least element.

Fact 3.9. *In any cpo K, if a function $F : K \longrightarrow K$ is ω-continuous, and if $k \in K$ is such that $F(k') \sqsupseteq k$ for all k', then $\bigsqcup_n F^{(n)}(k)$ is the least fixed point of F.*

To see this, recall the familiar fact that $\bigsqcup_n F^{(n)}(k)$ is the least fixed point of F above k; the hypothesis ensures that every fixed point of F is above k.

Now we may observe that $\bot_{\bullet} \mathrel{\dot{\subseteq}} \mathcal{R}(G)$ for every $G \in \mathbf{Set}_*^{\mathbf{Set}_*}$ and $\emptyset \subseteq \mathcal{T}(Z)$ for every $Z \in \mathbf{Set}_{\subseteq}$, so the initiality of $\bot_{\bullet}$ gives that $\langle\bot_{\bullet}, \emptyset, \gamma_0\rangle \sqsubseteq \mathcal{C}\langle G, Z, \tau\rangle$ for every $\langle G, Z, \tau\rangle \in \mathcal{I}{\downarrow}\mathcal{F}$. Hence, by Fact 3.9, $\langle R, T, \gamma\rangle$ is the least fixed point of $\mathcal{C}$. Moreover, if $\langle R, T, \gamma'\rangle$ is any fixed point of $\mathcal{C}$ connecting R to T, then Fact 2.9 gives that $\gamma' = \gamma$. In other words, γ is the unique natural transformation from R to $\mathcal{F}(T)$ satisfying

$$\gamma = \Psi_T \cdot \mathcal{R}(\gamma)\,.$$

(The uniqueness of γ can be proved using only the initiality of $\langle R_0, T_0, \gamma_0\rangle$ [3, Proposition 2.31], but the partial order in $\mathcal{I}{\downarrow}\mathcal{F}$ simplifies the proof here. Reynolds noted a similar uniqueness for the relational functor which he constructed in [10].)

Recovering the look-up algorithm

We may now unpack the fixed-point equation characterizing γ to recover the look-up algorithm "ap" which was written down but not justified in Section 1. Let $A \in \mathbf{Set}_*$ be any pointed set and let $\langle r_1, \ldots, r_m\rangle^\bullet \in R(A)$ be any T-indexed, A-valued generalized trie. We have

$$\begin{aligned}
\gamma_A\langle r_1, \ldots, r_m\rangle^\bullet &= \big(\Psi_T \cdot \prod_{i=1}^{m}{}_\bullet \gamma^{(k_i)}\big)_A \langle r_1, \ldots, r_m\rangle^\bullet \\
&= \big(\Psi_{TA} \circ \prod_{i=1}^{m}{}_\bullet (\gamma^{(k_i)})_A\big)\langle r_1, \ldots, r_m\rangle^\bullet \\
&= (\lambda\langle g_1, \ldots, g_m\rangle^\bullet \,.\, \lambda\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle \,.\, g_i\, t_1 \cdots t_{k_i}) \\
&\qquad\qquad \langle(\gamma^{(k_1)})_A r_1, \ldots, (\gamma^{(k_m)})_A r_m\rangle^\bullet \\
&= \lambda\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle \,.\, (\gamma^{(k_i)})_A r_i\, t_1 \cdots t_{k_i}\,. \qquad (14)
\end{aligned}$$

Recall that for any functors $F, G : M \longrightarrow L$ and $F', G' : L \longrightarrow K$ and any natural transformations $\tau : F \stackrel{\cdot}{\longrightarrow} G$ and $\tau' : F' \stackrel{\cdot}{\longrightarrow} G'$, the horizontal composition $\tau' \circ \tau : F' \circ F \stackrel{\cdot}{\longrightarrow} G' \circ G$ is given [7, p. 43 Formula 3] by

$$\tau' \circ \tau = (G' \circ \tau) \cdot (\tau' \circ F)\,.$$

For endofunctors $F, G : L \longrightarrow L$, $\tau : F \stackrel{\cdot}{\longrightarrow} G$, and any $k \geq 0$, this yields by an easy induction the formula for $\tau^{(k)} : F^{(k)} \stackrel{\cdot}{\longrightarrow} G^{(k)}$

$$\tau^{(k)} = (G^{(k-1)} \circ \tau) \cdot (G^{(k-2)} \circ \tau \circ F) \cdot \ldots \cdot (G \circ \tau \circ F^{(k-2)}) \cdot (\tau \circ F^{(k-1)})\,. \qquad (15)$$

So for any $k \geq 0$ and $r \in R^{(k)}(A)$, we have, since $\gamma^{(k)} : R^{(k)} \stackrel{\cdot}{\longrightarrow} \mathcal{F}_T^{(k)}$,

$$\begin{aligned}
(\gamma^{(k)})_A\, r &= [(\mathcal{F}_T^{(k-1)} \circ \gamma) \cdot (\mathcal{F}_T^{(k-2)} \circ \gamma \circ R) \cdot \ldots \cdot (\mathcal{F}_T \circ \gamma \circ R^{(k-2)}) \cdot (\gamma \circ R^{(k-1)})]_A\, r \\
&= [(\mathcal{F}_T^{(k-1)}(\gamma_A)) \circ (\mathcal{F}_T^{(k-2)}(\gamma_{R(A)})) \circ \cdots \circ (\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})) \circ \gamma_{R^{(k-1)}(A)}]\, r \\
&= \mathcal{F}_T^{(k-1)}(\gamma_A)(\mathcal{F}_T^{(k-2)}(\gamma_{R(A)})(\cdots(\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})(\gamma_{R^{(k-1)}(A)} r))\cdots))\,.
\end{aligned}$$

From Proposition 3.1 we have, for any $j \geq 0$, for an appropriately typed morphism h of $\mathbf{Set}_*$ and function f, and for $t_1, \ldots, t_j \in T$,

$$\mathcal{F}_T^{(j)} h\, f\, t_1 \cdots t_j = h(f\, t_1 \cdots t_j)\,.$$

Let $t_1, \ldots, t_k$ be any terms, and apply this in turn for $j = k-1, k-2, \ldots, 1$, yielding

$$
\begin{aligned}
&(\gamma^{(k)})_A r\, t_1 \cdots t_k \\
&\qquad = \mathcal{F}_T^{(k-1)}(\gamma_A)(\mathcal{F}_T^{(k-2)}(\gamma_{R(A)})(\cdots(\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})(\gamma_{R^{(k-1)}(A)} r))\cdots))t_1 \cdots t_k \\
&\qquad = \gamma_A(\mathcal{F}_T^{(k-2)}(\gamma_{R(A)})(\cdots(\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})(\gamma_{R^{(k-1)}(A)} r))\cdots)t_1 \cdots t_{k-1})t_k \\
&\qquad = \gamma_A(\gamma_{R(A)}((\cdots(\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})(\gamma_{R^{(k-1)}(A)} r))\cdots)t_1 \cdots t_{k-2})t_{k-1})t_k \\
&\qquad\ \vdots \\
&\qquad = \gamma_A(\gamma_{R(A)}(\cdots(\mathcal{F}_T(\gamma_{R^{(k-2)}(A)})(\gamma_{R^{(k-1)}(A)} r)t_1 t_2)\cdots)t_{k-1})t_k \\
&\qquad = \gamma_A(\gamma_{R(A)}(\cdots\gamma_{R^{(k-2)}(A)}(\gamma_{R^{(k-1)}(A)} r\, t_1)t_2 \cdots)t_{k-1})t_k\,.
\end{aligned}
$$

So Equation 14 becomes

$$\gamma_A\langle r_1, \ldots, r_m\rangle^\bullet = \lambda\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle\,.\,\gamma_A(\gamma_{R(A)}(\cdots(\gamma_{R^{(k-1)}(A)} r\, t_1)\cdots)t_{k_i-1})t_{k_i}\,.$$

Since γ_A is base-point preserving, we may expand this as

$$
\begin{aligned}
&\gamma_A \bullet = *_{A[T]} = \lambda t\,.\,*_A \\
&\gamma_A\langle r_1, \ldots, r_m\rangle = \lambda\langle i, \langle t_1, \ldots, t_{k_i}\rangle\rangle\,.\,\gamma_A(\gamma_{R(A)}(\cdots(\gamma_{R^{(k-1)}(A)} r\, t_1)\cdots)t_{k_i-1})t_{k_i}\,.
\end{aligned}
$$

This is precisely the recursive definition proposed in Section 1 for ap.

A more realistic set of terms

In a typical application of generalized tries, such as to a table of common subexpressions in a compiler, one would be likely to find that the "terms" to be looked up were not quite an instance of the term algebra T we have been discussing, but rather were defined by an equation like

$$T' = T'^{k_1} + \cdots + T'^{k_m} + V$$

with V being a large, possibly infinite, set of unstructured elements such as identifiers or numerals. We sketch here how any reasonable (that is, functorial) data structure for V-indexed look-up tables can be incorporated with the trie idea.

Suppose then that T' is as just described, that is, the least fixed point of a functor $\mathcal{T}' : \mathbf{Set}_\subseteq \longrightarrow \mathbf{Set}_\subseteq$ defined by

$$\mathcal{T}'(Z) = Z^{k_1} + \cdots + Z^{k_m} + V.$$

We suppose that a functor $B : \mathbf{Set}_* \longrightarrow \mathbf{Set}_*$ encapsulates some data structure for V-indexed tables, its look-up function being a natural transformation $\beta : B \stackrel{\cdot}{\longrightarrow} \mathcal{F}(V)$. (For example, B might assign to each pointed set A the set of all V-indexed, A-valued binary search trees. With such possibilities in mind, we refrain from supposing that β is an isomorphism, which would be to suppose a unique representing data structure for each finite function.)

The elements of $R'(A)$, the T'-indexed, A-valued tries, will correspondingly have an $m+1$st field in each tuple, containing an element of $B(A)$ or of $B(R'(A))$ or $\ldots$. That is, we define

$$\mathcal{R}'(G) = G^{(k_1)} \times_\bullet \cdots \times_\bullet G^{(k_m)} \times_\bullet B\,,$$
$$\mathcal{R}'(\tau) = \tau^{(k_1)} \times_\bullet \cdots \times_\bullet \tau^{(k_m)} \times_\bullet 1_B\,.$$

Suitable adjustments to the constructions used for Propositions 3.7 and 3.8 above will then produce a natural transformation (not an isomorphism unless β is)

$$\Psi' : \mathcal{R}' \circ \mathcal{F} \stackrel{\cdot}{\longrightarrow} \mathcal{F} \circ \mathcal{T}',$$

with the effect of

$$\Psi'_{ZA} : \prod_{i=1}^{m}{}_\bullet\, \mathcal{F}_Z^{(k_i)}(A) \times_\bullet B(A) \longrightarrow \mathcal{F}_{\sum_{i=1}^m Z^{k_i}+V}(A)$$

given by

$$\Psi'_{ZA}\langle g_1, \ldots, g_m, b\rangle^\bullet = \mu^\bullet_{m+1}\langle \nu_{k_1}(g_1), \ldots, \nu_{k_m}(g_m), \beta_A(b)\rangle^\bullet\,,$$

that is,

$$\Psi'_{ZA}\langle g_1, \ldots, g_m, b\rangle^\bullet\langle i, \langle z_1, \ldots, z_{k_i}\rangle\rangle = g_i\, z_1 \cdots z_{k_i} \qquad \text{if } i \leq m,$$
$$\Psi'_{ZA}\langle g_1, \ldots, g_m, b\rangle^\bullet\langle m+1, v\rangle = \beta_A b\, v\,.$$

The construction of $\mathcal{C}' : \mathcal{I}{\downarrow}\mathcal{F} \longrightarrow \mathcal{I}{\downarrow}\mathcal{F}$ and its least fixed point $\langle R', T', \gamma'\rangle$, with $\gamma' : R' \stackrel{\cdot}{\longrightarrow} \mathcal{F}(T')$ and $\gamma' = \Psi'_{T'} \cdot \mathcal{R}'(\gamma')$, may then proceed as before.

Section 4. Many-sorted Tries.

In extending the result of Section 3 to many-sorted term algebras the greatest difficulties are notational. For many-sorted algebras we follow Goguen, Thatcher, Wagner, and Wright [5] in substance, although our notion of signature is arranged differently from theirs in order to follow our treatment of the one-sorted case more closely.

Let S be a set of "sorts". We call a finite sequence of sorts, that is an element w of S^*, the free monoid over S, a "word" on S; we denote its length as $|w|$. A signature for an S-sorted algebra should provide for each operator a result sort and a finite sequence, that is a word, of argument sorts. It is necessary to our trie construction that for each $s \in S$, only finitely many operators have result sort s (this is our only substantial departure from the notion of S-sorted algebra in [5]). Accordingly we suppose that an S-sorted signature is a pair $(\overline{m}, \kappa)$ where $\overline{m}$ is an S-indexed family of non-negative integers and, for each $s \in S$, $\kappa(s) : \{1, \ldots, \overline{m}(s)\} \longrightarrow S^*$. For $s \in S$, $\overline{m}(s)$ is the number of operators having result sort s, and for $1 \leq i \leq \overline{m}(s)$, $\kappa(s)_i$ gives the arity (a word on S) of the ith operator of result sort s.

To give as familiar an example as possible of a two-sorted term algebra, we take ordered trees and ordered forests, with the tree nodes again labeled by a single bit; the mutually recursive definition is

$$T_{\mathrm{T}} = T_{\mathrm{F}} + T_{\mathrm{F}},$$
$$T_{\mathrm{F}} = \{\langle\rangle\} + T_{\mathrm{T}} \times T_{\mathrm{F}}\,.$$

We would like the T_{T}- and T_{F}-indexed tries to come out satisfying the corresponding equations

$$R_{\mathrm{T}}(A) = R_{\mathrm{F}}(A) \times R_{\mathrm{F}}(A),$$
$$R_{\mathrm{F}}(A) = A \times R_{\mathrm{T}}(R_{\mathrm{F}}(A))\,.$$

As is well known, ordered forests are in one-to-one correspondence with binary trees. Following the example may be facilitated by observing that a T_{F}-indexed trie comes out as a reformatting, via the template $\langle\ ,\ ,\ \rangle \mapsto \langle\ ,\langle\ ,\ \rangle\rangle$, of a T_{B}-indexed one-sorted trie.

The same two-sorted syntax of terms may be expressed more opaquely according to our definition of many-sorted signature by the choices $S = \{\mathrm{T}, \mathrm{F}\}$ (for "tree" and "forest", not the truth values), $\overline{m}(\mathrm{T}) = \overline{m}(\mathrm{F}) = 2$, $\kappa(\mathrm{T})_1 = \kappa(\mathrm{T})_2 = \mathrm{F}$, $\kappa(\mathrm{F})_1 = \epsilon$ (the empty word), and $\kappa(\mathrm{F})_2 = \mathrm{TF}$.

We follow [5] in a convenient generalization of the exponential notation: if X is an S-indexed family of sets and $w \in S^*$ is a word, $w = w_1 w_2 \ldots w_k$ say, then X^w denotes $X(w_1) \times \cdots \times X(w_k)$.

Fixing now on any arbitrarily chosen S, $\overline{m}$, and κ, we may make the construction of the term algebra look very much like the one-sorted case. Define the functor $\overline{T} : \mathbf{Set}_{\subseteq}^{S} \to \mathbf{Set}_{\subseteq}^{S}$ by, for each $s \in S$ and $Z \in \mathbf{Set}_{\subseteq}^{S}$,

$$\overline{T}(Z)(s) = \sum_{i=1}^{\overline{m}(s)} Z^{\kappa(s)_i}.$$

Then let $\overline{T} \in \mathbf{Set}_{\subseteq}^{S}$ be given by

$$\overline{T} = \bigcup_n \overline{T}^{(n)}(\lambda s \in S\,.\,\emptyset)\,.$$

(Slightly abusing notation, we write plain $\subseteq$ and $\bigcup$ for the componentwise extension of inclusion and union to S-indexed families of sets. In particular $Z \subseteq Z'$, where Z and Z' are S-indexed families of sets, if and only if $Z(s) \subseteq Z'(s)$ for all $s \in S$. Analogously we extend $\dot{\subseteq}$ and $\dot{\bigcup}$ to S-indexed families of functors.)

Similarly generalize the notation for n-fold composition: if G is an S-indexed family of endofunctors of $\mathbf{Set}_*$, that is an object of $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$, and w is a word on S with $|w| = k$, let $G^{(w)}$ denote $G(w_1) \circ \cdots \circ G(w_k)$. Likewise, if τ is an S-indexed family of natural transformations of such functors, that is a morphism of $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$, let $\tau^{(w)}$ denote $\tau(w_1) \circ \cdots \circ \tau(w_k)$.

Now we may define $\bar{R}$, the $\overline{T}$-indexed trie functor, much as before:

$$\bar{R} = \dot{\bigcup_n} \overline{\mathcal{R}}^{(n)}(\lambda s \in S.\bot.)$$

where $\overline{\mathcal{R}} : \left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S \longrightarrow \left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$ is given, for objects and morphisms G and τ of $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$ and for $s \in S$, by

$$\overline{\mathcal{R}}(G)(s) = \dot{\prod_{i=1}^{\overline{m}(s)}}{}_{\bullet} G^{(\kappa(s)_i)}, \qquad \overline{\mathcal{R}}(\tau)(s) = \dot{\prod_{i=1}^{\overline{m}(s)}}{}_{\bullet} \tau^{(\kappa(s)_i)}.$$

There is nothing new in the verification that $\overline{T}$ and $\overline{\mathcal{R}}$ are continuous endofunctors of $\mathbf{Set}_{\subseteq}^S$ and $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$ respectively.

The sets of terms and of tries which for readability we called T_{T}, etc. have official designations $\overline{T}(\mathrm{T})$, etc.

Using the notation of [7, p. 45], we have the continuous functor

$$\begin{aligned}
&\mathcal{F}^S : \mathbf{Set}_*^S \longrightarrow \left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S,\\
&\mathcal{F}^S(Z)(s) = \mathcal{F}(Z(s)),\\
&\mathcal{F}^S(Z \subseteq Z')(s) = \mathcal{F}(Z(s) \subseteq Z'(s)),
\end{aligned}$$

for S-indexed families of sets Z, Z' and for $s \in S$.

Applying once more the notational idea from [5] of generalizing from non-negative integers to words, we may, for any category L and $w \in S^*$ with $|w| = k$, define

$$\begin{aligned}
&\Delta_w : L^S \longrightarrow L^k,\\
&\Delta_w(x) = \langle x(w_1), \ldots, x(w_k)\rangle
\end{aligned}$$

for both objects and morphisms of L^S. As earlier with Δ_n, we shall leave the category L to be determined by context.

Now, combining for each result sort its own version of the natural isomorphism $\psi_{k_1 \cdots k_m}$ introduced in Proposition 3.7, we may define a natural isomorphism $\overline{\Psi}$ of functors from $\mathbf{Set}_{\subseteq}^S$ to $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$,

$$\overline{\Psi} : \overline{\mathcal{R}} \circ \mathcal{F}^S \cong \mathcal{F}^S \circ \overline{T},$$

by, for any $Z \in \mathbf{Set}_{\subseteq}^S$ and $s \in S$,

$$\begin{aligned}
\overline{\Psi}_Z(s) &= [\psi_{|\kappa(s)_1| \cdots |\kappa(s)_{\overline{m}(s)}|} \circ (\Delta_{\kappa(s)_1} \times \cdots \times \Delta_{\kappa(s)_{\overline{m}(s)}}) \circ \Delta_{\overline{m}(s)}]_Z\\
&: \overline{\mathcal{R}}(\mathcal{F}^S Z)(s) \cong \mathcal{F}^S(\overline{T} Z)(s) = \mathcal{F}(\overline{T} Z(s)).
\end{aligned}$$

For $A \in \mathbf{Set}_*$, the effect of $\overline{\Psi}_Z(s)_A$ is given by

$$\begin{aligned}
&\overline{\Psi}_Z(s)_A : \dot{\prod_{i=1}^{\overline{m}(s)}}{}_{\bullet} (\mathcal{F}_Z^S)^{(\kappa(s)_i)} A = \dot{\prod_{i=1}^{\overline{m}(s)}}{}_{\bullet} A^{[Z((\kappa(s)_i)_{|\kappa(s)_i|})] \cdots [Z((\kappa(s)_i)_1)]} \longrightarrow \mathcal{F}_{\sum_{i=1}^{\overline{m}(s)} Z^{\kappa(s)_i}}(A),\\
&\overline{\Psi}_Z(s)_A \langle g_1, \ldots, g_{\overline{m}(s)}\rangle^\bullet = \lambda \langle i, \langle z_1, \ldots, z_{|\kappa(s)_i|}\rangle\rangle \,.\, g_i\, z_1 \cdots z_{|\kappa(s)_i|}\,.
\end{aligned}$$

Write $\mathcal{I}^S$ for the identity functor on $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$. Then we may define a continuous endofunctor of the comma category $\mathcal{I}^S{\downarrow}\mathcal{F}^S$ by

$$\begin{aligned}
&\overline{\mathcal{C}} : \mathcal{I}^S{\downarrow}\mathcal{F}^S \longrightarrow \mathcal{I}^S{\downarrow}\mathcal{F}^S\\
&\quad : \langle G, Z, \tau : G \stackrel{.}{\longrightarrow} \mathcal{F}^S Z\rangle \longmapsto \langle \overline{\mathcal{R}}(G), \overline{\mathcal{T}}(Z), \overline{\Psi}_Z \cdot \overline{\mathcal{R}}(\tau)\rangle\\
&\quad : \langle \sigma : G \stackrel{.}{\longrightarrow} G', (Z \subseteq Z')\rangle \longmapsto \langle \overline{\mathcal{R}}(\sigma), (\overline{\mathcal{T}}(Z) \subseteq \overline{\mathcal{T}}(Z'))\rangle\,.
\end{aligned}$$

Lemma A.5 shows, taking $D = \mathcal{I}^S,\ \ E = \mathcal{F}^S,\ \ G = R = \overline{\mathcal{R}},\ \ \rho = 1_{\overline{\mathcal{R}}},\ \ H = \overline{\mathcal{T}}$, and $\sigma = \overline{\Psi}$, that $\overline{\mathcal{C}}$ is a well-defined functor; $\overline{\mathcal{C}}$ is continuous because $\overline{\mathcal{R}} \times \overline{\mathcal{T}}$ is.

The S-indexed family $\overline{\bot}_{\bullet} \stackrel{\text{def}}{=} \lambda s \,.\, \bot_{\bullet}$ is an initial object in $\left(\mathbf{Set}_*^{\mathbf{Set}_*}\right)^S$, and we have the natural isomorphism

$$\overline{\gamma}_0 \stackrel{\text{def}}{=} \lambda s \,.\, \gamma_0 : \overline{\bot}_{\bullet} \cong \lambda s \,.\, \mathcal{F}(\emptyset)\,.$$

We obtain, as in the one-sorted case,

$$\langle \overline{R}, \overline{T}, \overline{\gamma}\rangle = \bigsqcup_n \overline{\mathcal{C}}^{(n)}\langle \overline{\bot}_{\bullet}, \lambda s \,.\, \mathcal{F}(\emptyset), \overline{\gamma}_0\rangle,$$

with $\overline{\gamma}$ a natural isomorphism, and the unique natural transformation satisfying $\overline{\gamma} = \overline{\Psi}_{\overline{T}} \cdot \overline{\mathcal{R}}(\overline{\gamma})$.

To uncover the look-up algorithm in $\overline{\gamma}$, we may begin by writing an analogue to Formula 14, for $s \in S,\ \ A \in \mathbf{Set}_*$, and $\langle r_1, \ldots, r_{\overline{m}(s)}\rangle \in \overline{R}(s)(A)$ any $\overline{T}(s)$-indexed, A-valued trie,

$$\overline{\gamma}(s)_A\langle r_1, \ldots, r_{\overline{m}(s)}\rangle^{\bullet} = \lambda\langle i, \langle t_1, \ldots, t_{|\kappa(s)_i|}\rangle\rangle \,.\, (\overline{\gamma}^{(\kappa(s)_i)})_A r_i\, t_1 \cdots t_{|\kappa(s)_i|}\,. \tag{16}$$

The formula for horizontal composition generalizes just as well to k pairs of functors which need not all be the same; so for any category L, any two S-indexed families F and G of endofunctors of L, that is any two objects of $(L^L)^S$, any morphism $\tau : F \stackrel{.}{\longrightarrow} G$ of $(L^L)^S$, and any $w \in S^*$ with $|w| = k$ we have the S-sorted analogue of Formula 15:

$$\begin{aligned}
\tau^{(w)} = {}&(G^{(w_1\cdots w_{k-1})} \circ \tau(w_k)) \cdot (G^{(w_1\cdots w_{k-2})} \circ \tau(w_{k-1}) \circ F^{(w_k)}) \cdot \ \ldots\\
&\ldots \cdot (G^{(w_1)} \circ \tau(w_2) \circ F^{(w_3\cdots w_k)}) \cdot (\tau(w_1) \circ F^{(w_2\cdots w_k)})\,.
\end{aligned}$$

Thus for $r \in \overline{R}^{(w)}(A)$ we have, since $\overline{\gamma}^{(w)} : \overline{R}^{(w)} \stackrel{.}{\longrightarrow} (\mathcal{F}^S_{\overline{T}})^{(w)}$,

$$\begin{aligned}
(\overline{\gamma}^{(w)})_A r = {}&(\mathcal{F}^S_{\overline{T}})^{(w_1\cdots w_{k-1})}(\overline{\gamma}(w_k)_A)(\mathcal{F}^S_{\overline{T}})^{(w_1\cdots w_{k-2})}(\overline{\gamma}(w_{k-1})_{\overline{R}^{(w_k)}(A)})(\cdots\\
&\cdots\ (\mathcal{F}^S_{\overline{T}})^{(w_1)}(\overline{\gamma}(w_2)_{\overline{R}^{(w_3\cdots w_k)}(A)})(\overline{\gamma}(w_1)_{\overline{R}^{(w_2\cdots w_k)}(A)} r))\ \cdots))\,.
\end{aligned}$$

For any prefix $w^- = w_1 \cdots w_j$ of w, for any morphism $h : B \longrightarrow B'$ of $\mathbf{Set}_*$, for $f \in B^{[\overline{T}(w_j)]\cdots[\overline{T}(w_1)]}$, and for $t_1 \in \overline{T}(w_1),\ \ldots\ ,\ t_j \in \overline{T}(w_j)$, Proposition 3.1 yields

$$(\mathcal{F}^S_{\overline{T}})^{(w^-)} h\, f\, t_1 \cdots t_j = h(f\, t_1 \cdots t_j)\,.$$

Then for $A \in \mathbf{Set}_*$, $r \in \overline{R}^{(w)}(A)$, and $t_i \in \overline{T}(w_i)$, $i = 1, \ldots, k$, we obtain

$$(\overline{\gamma}^{(w)})_A r\, t_1 \cdots t_k = \overline{\gamma}(w_k)_A(\overline{\gamma}(w_{k-1})_{\overline{R}^{(w_k)}(A)}(\cdots \\ \overline{\gamma}(w_2)_{\overline{R}^{(w_3 \cdots w_k)}(A)}(\overline{\gamma}(w_1)_{\overline{R}^{(w_2 \cdots w_k)}(A)} r\, t_1)\, t_2 \cdots)\, t_{k-1})\, t_k\,.$$

So we may write the S-indexed family of mutually recursive routines implicit in Formula 16 as

$$\overline{\gamma}(s)_A \bullet = \lambda t \in \overline{T}(s)\,.\, *_A$$
$$\overline{\gamma}(s)_A \langle r_1, \ldots, r_{\overline{m}(s)} \rangle = \lambda \langle i, \langle t_1, \ldots, t_{|\kappa(s)_i|} \rangle \rangle \,.\, \overline{\gamma}((\kappa(s)_i)_{|\kappa(s)_i|})_A(\cdots \\ \cdots\ (\overline{\gamma}((\kappa(s)_i)_1)_{\overline{R}^{((\kappa(s)_i)_2 \cdots (\kappa(s)_i)_{|\kappa(s)_i|})}(A)} r\, t_1) \cdots)\, t_{|\kappa(s)_i|}\,.$$

This is not really as horrible as the general notation makes it look. For our example of ordered trees and forests, calling the two look-up functions T-ap and F-ap rather than $\overline{\gamma}(\mathrm{T})$ and $\overline{\gamma}(\mathrm{F})$, it works out to

$$\text{T-ap}_A\ \bullet\ t = *_A,$$
$$\text{T-ap}_A \langle r_1, r_2 \rangle \langle 1, \langle f \rangle \rangle = \text{F-ap}_A\, r_1\, f,$$
$$\text{T-ap}_A \langle r_1, r_2 \rangle \langle 2, \langle f \rangle \rangle = \text{F-ap}_A\, r_2\, f;$$
$$\text{F-ap}_A\ \bullet\ f = *_A,$$
$$\text{F-ap}_A \langle r_1, r_2 \rangle \langle 1, \langle \rangle \rangle = r_1,$$
$$\text{F-ap}_A \langle r_1, r_2 \rangle \langle 2, \langle t, f \rangle \rangle = \text{F-ap}_A(\text{T-ap}_{R_{\mathrm{F}}(A)}\, r_2\, t)\, f\,.$$

Appendix. Colimits in Comma Categories.

Below are two theorems which show the creation of a colimit by parameters in a category of functors and in a comma category. The first is MacLane's "(co)limits with parameters" theorem [7, Theorem V.3.2] which we state here as a fact. The second is Bierle's theorem, which shows how to construct a colimit by components in a comma category [1, Fact I.4]. This theorem was independently discovered by Connelly [3] and by Casley, et. al. [2]. The presentation here uses MacLane's concept of "creating colimits" [7, p. 108], defined as follows.

A functor $V : K \longrightarrow M$ *creates colimits* for a functor $F : J \longrightarrow K$ if and only if for any colimit cone $\tau : V \circ F \stackrel{\cdot}{\longrightarrow} m$ there is a unique object k in K and unique cone $\sigma : F \stackrel{\cdot}{\longrightarrow} k$ such that $V \circ \sigma = \tau$ and, further, this σ is a colimit cone.

The functor V is often a forgetful functor, sending every morphism of K to itself as a morphism of M. In this case we may describe the creation of colimits by V for F as the existence, for any colimit cone $\tau : V \circ F \stackrel{\cdot}{\longrightarrow} m$, of a unique $k \in K$ such that $Fk = m$, and $\tau : F \stackrel{\cdot}{\longrightarrow} k$ is also a cone in K, and τ is also a colimit in K.

The definition of "creates colimits" yields the following obvious fact.

Fact A.1. *Let $\sigma : F \stackrel{\cdot}{\longrightarrow} k$ be a colimit cone created by V. For any cone $\eta : F \stackrel{\cdot}{\longrightarrow} k'$, the unique mediating morphism (u.m.m.) f from $V \circ \sigma$ to $V \circ \eta$ is the image of the u.m.m. g from σ to η (that is, $Vg = f$).*

In the case of V a forgetful functor, Fact A.1 says that any u.m.m. in K from the created colimit σ to another cone on F is the same as the u.m.m. in M between the composites of the two cones with V.

The next lemma gives a condition under which a functor that creates colimits can be used to prove that a second functor preserves colimits.

Lemma A.2. *Let $F : J \longrightarrow K$, $G : K \longrightarrow L$, and $V : L \longrightarrow M$ be functors. If V creates colimits for $G \circ F$, and $V \circ G$ preserves colimits of F, then G preserves colimits of F.*

Proof. Let $\sigma : F \stackrel{\cdot}{\longrightarrow} k$ be any colimit cone. By hypothesis, $V \circ G \circ \sigma$ is a colimit cone. Since V creates colimits of $G \circ F$, $G \circ \sigma$ must be the colimit cone created by V for $V \circ G \circ \sigma$. □

MacLane's theorem gives a way to create a colimit by parameters in a functor category. It uses the following notation. Let $|L|$ denote the discrete category whose objects are those of L, and let $i : |L| \longrightarrow L$ denote the insertion functor. Define $i^* : K^L \longrightarrow K^{|L|}$ as the functor sending $H \mapsto H \circ i$ for each object (functor) and $\eta \mapsto \eta \circ i$ for each morphism (natural transformation).

Fact A.3. [7, Theorem V.3.2] *Let K and L be any categories. The functor $i^* : K^L \longrightarrow K^{|L|}$ creates colimits (for any functor $F : J \longrightarrow K^L$).*

Lemma A.2 and Fact A.3 mean that for any functors $F : J \longrightarrow M$ and $G : M \longrightarrow K^L$, if $i^* \circ G$ preserves colimits of F, then G preserves colimits of F. This is especially useful because $|L|$ is discrete and so it is sufficient to show that $i^* \circ G$ preserves colimits for each object l of L.

Beierle's theorem uses the following notation. Let $T : L \longrightarrow K$ and $S : M \longrightarrow K$ be any functors. For any object $z \in T{\downarrow}S$, $z_\downarrow$ will denote the morphism in z (i.e., the third component). Consider the following forgetful functor and natural transformation (the latter is taken from [1]).

$$\begin{aligned} P : T{\downarrow}S &\longrightarrow L \times M \\ : \langle l, m, f\rangle &\longmapsto \langle l, m\rangle \\ : \langle u, v\rangle &\longmapsto \langle u, v\rangle \end{aligned} \qquad\qquad \begin{aligned} P_\downarrow : T \circ \Pi_1 \circ P &\stackrel{\cdot}{\longrightarrow} S \circ \Pi_2 \circ P \\ : z &\longmapsto z_\downarrow \end{aligned}$$

The composites $\Pi_1 \circ P$ and $\Pi_2 \circ P$ will be abbreviated by P_1 and P_2 respectively.

Here is Beierle's Fact I.4. The statement of the theorem here is more detailed than the references [1, 2, 3], but the proof method is the same.

Theorem A.4. *Let $T : L \longrightarrow K$ and $S : M \longrightarrow K$ be any functors. Let $F : J \longrightarrow T{\downarrow}S$ be any functor such that T preserves colimits of $P_1 \circ F$. Then the functor $P : T{\downarrow}S \longrightarrow L \times M$ creates colimits for F. Specifically, if $\tau : P \circ F \stackrel{\cdot}{\longrightarrow} \langle l, m\rangle$ is any colimit cone of $P \circ F$, then in the created colimit cone $\tau : F \stackrel{\cdot}{\longrightarrow} \langle l, m, f\rangle$, f is the u.m.m. in K from the colimit cone $T \circ P_1 \circ \tau$ to the cone $(S \circ P_2 \circ \tau) \cdot (P_\downarrow \circ F)$. Further, if S preserves colimits of $P_2 \circ F$, and $P_\downarrow \circ F$ is a natural isomorphism, then f is an isomorphism.*

Proof. Let $\tau : P \circ F \stackrel{\cdot}{\longrightarrow} \langle l, m\rangle$ be a colimit cone. Since T preserves $\Pi_1 \circ \tau$ as a colimit cone, $T \circ \Pi_1 \circ \tau$ is a colimit cone. Let $f : T(l) \longrightarrow S(m)$ denote the u.m.m. from $T \circ \Pi_1 \circ \tau$ to $(S \circ \Pi_2 \circ \tau) \cdot (P_\downarrow \circ F)$. That is, f satisfies the following diagram for each $j \in J$.

$$\begin{array}{ccc} T(F(j)_1) & \xrightarrow{T(\tau(j)_1)} & T(l) \\ {\scriptstyle F(j)_\downarrow}\Big\downarrow & & \Big\downarrow{\scriptstyle f} \\ S(F(j)_2) & \xrightarrow[S(\tau(j)_2)]{} & S(m). \end{array}$$

The diagram gives that $\tau(j)$ is a morphism in $T{\downarrow}S$ from $F(j)$ to $\langle l, m, f\rangle$ for each $j \in J$. Consequently, since τ is a cone from $P \circ F$ to $\langle l, m\rangle$, it is also a cone from F to $\langle l, m, f\rangle$. To see that f is unique in the vertex $\langle l, m, f\rangle$, suppose $\langle l, m, f'\rangle$ to be any other vertex of τ in $T{\downarrow}S$. For each $j \in J$, the morphism $\tau(j) : F(j) \longrightarrow \langle l, m, f'\rangle$ implies that the diagram continues to commute if f is replaced by f'. Consequently, f' mediates from $T \circ \Pi_1 \circ \tau$ to $(S \circ \Pi_2 \circ \tau) \cdot (P_\downarrow \circ F)$ and must be the u.m.m. f. When S preserves colimits of $P_2 \circ F$ and $P_\downarrow \circ F$ is an isomorphism, $(S \circ \Pi_2 \circ \tau) \cdot (P_\downarrow \circ F)$ is a colimit cone and so f is an isomorphism.

It remains to prove that τ is a colimit cone in $T{\downarrow}S$. Let $\sigma : F \stackrel{.}{\longrightarrow} \langle \bar{l}, \bar{m}, \bar{f}\rangle$ be any cone and let $\nu : \langle l, m\rangle \longrightarrow \langle \bar{l}, \bar{m}\rangle$ denote the u.m.m. from $P \circ \tau$ to $P \circ \sigma$. It must be proved that ν is a morphism in $T{\downarrow}S$ from $\langle l, m, f\rangle$ to $\langle \bar{l}, \bar{m}, \bar{f}\rangle$ and that it uniquely mediates from τ to σ. The only part of this that is any work is to show that ν is a morphism at all, as follows.

Since $\tau(j) : F(j) \longrightarrow \langle l, m, f\rangle$ for each $j \in J$ and since ν_2 mediates from $P_2 \circ \tau$ to $P_2 \circ \sigma$, the following equations are valid for each $j \in J$:

$$S(\nu_2)\circ f\circ T(\tau(j)_1) = S(\nu_2)\circ S(\tau(j)_2)\circ F(j)_\downarrow = S(\nu_2\circ\tau(j)_2)\circ F(j)_\downarrow = S(\sigma(j)_2)\circ F(j)_\downarrow.$$

Since ν_1 mediates from $P_1 \circ \tau$ to $P_1 \circ \sigma$ and since $\sigma(j) : F(j) \longrightarrow \langle \bar{l}, \bar{m}, \bar{f}\rangle$ for each $j \in J$, the following equations also hold for each $j \in J$:

$$\bar{f} \circ T(\nu_1) \circ T(\tau(j)_1) = \bar{f} \circ T(\nu_1 \circ \tau(j)_1) = \bar{f} \circ T(\sigma(j)_1) = S(\sigma(j)_2) \circ F(j)_\downarrow.$$

These two sequences of equations show that $S(\nu_2) \circ f$ and $\bar{f} \circ T(\nu_1)$ both mediate from colimit cone $T \circ P_1 \circ \tau$ to cone $(S \circ P_2 \circ \sigma) \cdot (P_\downarrow \circ F)$ and so must be equal. Consequently, ν is a morphism in $T{\downarrow}S$ from $\langle l, m, f\rangle$ to $\langle \bar{l}, \bar{m}, \bar{f}\rangle$.

That ν uniquely mediates from σ to τ is true simply because ν uniquely mediates from $P \circ \sigma$ to $P \circ \tau$. $\square$

The following corollary gives a condition under which a functor whose codomain is a comma category will preserve colimits. It is similar to Corollary 2.19.3 in [3].

Corollary A.4.1. *Let $H : Q \longrightarrow T{\downarrow}S$ be any functor. For any functor $F : J \longrightarrow Q$, if $P \circ H$ preserves colimits of F and T preserves colimits of $P_1 \circ H \circ F$, then H preserves colimits of F.*

Proof. By Theorem A.4, P creates colimits of $H \circ F$. Take V to be P and H to be G in Lemma A.2; then H preserves colimits of F. $\square$

The following diagram lemma for the construction of an endofunctor of a comma category, which we apply in Sections 3 and 4, generalizes a construction used in Chapter 7 of [3] that in turn was inspired by the relational functors defined by Reynolds [10]. It seems time it was recorded separately.

Lemma A.5. *Given five functors $D : L \longrightarrow K$, $E : M \longrightarrow K$, $G : L \longrightarrow L$, $H : M \longrightarrow M$, and $R : K \longrightarrow K$, and two natural transformations $\rho : D \circ G \stackrel{.}{\longrightarrow} R \circ D$ and $\sigma : R \circ E \stackrel{.}{\longrightarrow} E \circ H$, the following defines a functor:*

$$\begin{aligned} C &: D{\downarrow}E \longrightarrow D{\downarrow}E \\ &: \langle l, m, f\rangle \longmapsto \langle G(l), H(m), \sigma_m \circ R(f) \circ \rho_l\rangle \\ &: \langle g, h\rangle \longmapsto \langle G(g), H(h)\rangle\,. \end{aligned}$$

Proof. To see that C is a functor, consider any morphism $\langle g, h\rangle : \langle l, m, f\rangle \longrightarrow \langle l', m', f'\rangle$ of $D{\downarrow}E$ as a commutative diagram in K:

$$(*) \qquad \begin{tikzcd} D(l) \arrow[r, "D(g)"] \arrow[d, "f"'] & D(l') \arrow[d, "f'"] \\ E(m) \arrow[r, "E(h)"'] & E(m')\,. \end{tikzcd}$$

This is sent by C to the outer rectangle of the following diagram.

$$(**) \qquad \begin{tikzcd} D(G(l)) \arrow[r, "D(G(g))"] \arrow[d, "\rho_l"'] & D(G(l')) \arrow[d, "\rho_{l'}"] \\ R(D(l)) \arrow[r, "R(D(g))"] \arrow[d, "R(f)"'] & R(D(l')) \arrow[d, "R(f')"] \\ R(E(m)) \arrow[r, "R(E(h))"'] \arrow[d, "\sigma_m"'] & R(E(m')) \arrow[d, "\sigma_{m'}"] \\ E(H(m)) \arrow[r, "E(H(h))"'] & E(H(m'))\,. \end{tikzcd}$$

The upper pane of $(**)$ commutes because ρ is a natural transformation, and the lower pane because σ is. The center pane is simply the image of diagram $(*)$ under R. Consequently, $(**)$ commutes and C is well defined. C preserves identities and composition simply because G and H are functors. $\square$

As we apply the lemma here, we have D always the identity I_K, with $G = R$ and $\rho = 1_R$, so that the definition of C reduces to

$$\begin{aligned} C &: I_K{\downarrow}E \longrightarrow I_K{\downarrow}E \\ &: \langle l, m, f\rangle \mapsto \langle R(l), H(m), \sigma_m \circ R(f)\rangle \qquad : \langle g, h\rangle \mapsto \langle R(g), H(h)\rangle\,. \end{aligned}$$

REFERENCES

1. C. Beierle, *Algebraic Implementations in an Integrated Software Development and Verification System*, Dissertation, University of Kaiserlautern, Dec. 1985.
2. R. Casley, R. F. Crew, J. Meseguer, and V. Pratt, *Temporal Structures*, Category Theory and Computer Science, Lecture Notes in Computer Science 389, Springer-Verlag, 1989.
3. R. H. Connelly, *A Comparison of Semantic Domains for Interleaving*, Dissertation, Syracuse University, Aug. 1990.
4. E. Fredkin, *Trie Memory*, Comm. ACM **3** (Sept. 1960), 490-499.
5. J. A. Goguen, J. W. Thatcher, E. G. Wagner, and J. B. Wright, *Initial Algebra Semantics and Continuous Algebras*, J. ACM **24** (Jan. 1977), 68–95.
6. D. E. Knuth, *The Art of Computer Programming, Vol. 3: Sorting and Searching*, Addison-Wesley, Reading, Mass., 1973.
7. S. MacLane, *Categories for the Working Mathematician*, Graduate Texts in Mathematics 5, Springer-Verlag, 1971.
8. A. Poigné, *Cartesian Closure—Higher Types in Categories*, Category Theory and Computer Programming, Lecture Notes in Computer Science 240, Springer-Verlag, 1985.
9. J. C. Reynolds, *Semantics as a Design Tool*, Class notes for CS15–850A—Advanced Topics in Theory, Carnegie-Mellon University, 1988.
10. ———, *On the Relation between Direct and Continuation Semantics*, Automata, Languages, and Programming—Second Colloquium, Lecture Notes in Computer Science 14, Springer-Verlag, 1974.
11. M. B. Smyth and G. D. Plotkin, *The Category-Theoretic Solution of Recursive Domain Equations*, SIAM J. on Computing **11** (Nov. 1982), 761–783.
12. M. Spivey, *A Categorical Approach to the Theory of Lists*, Mathematics of Program Construction, Lecture Notes in Computer Science 375, Springer-Verlag, 1989, pp. 399–408.

RICHARD CONNELLY, DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, PROVIDENCE COLLEGE, RIVER AND EATON STREETS, PROVIDENCE RI 02918

LOCKWOOD MORRIS, SCHOOL OF COMPUTER AND INFORMATION SCIENCE, 4–116 CENTER FOR SCIENCE AND TECHNOLOGY, SYRACUSE UNIVERSITY, SYRACUSE NY 13244–4100