RIMS-1966

Continuous Functions on Final Comodels of Free Algebraic Theories

By

Tomoya YOSHIDA

September 2022

京都大学　数理解析研究所

RESEARCH INSTITUTE FOR MATHEMATICAL SCIENCES

KYOTO UNIVERSITY, Kyoto, Japan

Continuous Functions on Final Comodels of Free Algebraic Theories

Tomoya Yoshida[1]

Reserach Institute for Mathematical sciences
Kyoto University
Kyoto, Japan

Abstract

In 2009, Ghani, Hancock and Pattinson gave a tree-like representation of stream processors $A^{\mathbb{N}} \to B^{\mathbb{N}}$. In 2021, Garner showed that this representation can be established in terms of *algebraic theory* and *comodels*: the set of infinite streams $A^{\mathbb{N}}$ is the *final comodel* of the *algebraic theory of* A*-valued input* $\mathbb{T}_A$ and the set of stream processors $\mathit{Top}(A^{\mathbb{N}}, B^{\mathbb{N}})$ can be seen as the *final* $\mathbb{T}_A$-$\mathbb{T}_B$*-bimodel*. In this paper, we generalize Garner's results to the case of *free algebraic theories*.

1 Introduction

Writing and verifying programs handling infinite objects such as *streams* and infinite trees are highly non-trivial tasks. To ease it, many attempts to identify the mathematical principles behind infinite computation on infinite data structures have been made. Among them, a most active and well-developed areas are the theory of coalgebras, in which infinite objects are captured as elements of final coalgebras which enjoy nice universality and useful principles for programming and verification. This work can be seen as a contribution in this direction, making use of the recent development of algebraic theories and their comodels.

1.1 Background and Our Result

In computer science, coalgebras for functors appear in various ways [10,11,17]; one is as the way of implementations of infinite data structures, which appear as elements of final coalgebras. The simplest example of infinite data structures is infinite sequences (often called streams). For a set A, the set $A^{\mathbb{N}}$ of A-valued streams is a coalgebra for an endofunctor $A \times (\_)$ on Set with a coalgebra structure map $\alpha : A^{\mathbb{N}} \to A \times A^{\mathbb{N}}$ given by

$$a_0 a_1 a_2 \cdots \mapsto (a_0, a_1 a_2 \cdots).$$

Moreover, as is well known, the coalgebra $(A^{\mathbb{N}}, \alpha)$ is indeed the final coalgebra for the functor $A \times (\_)$, i.e., for every $(A \times (\_))$-coalgebra $(X, \phi : X \to A \times X)$, there exists a unique $(A \times (\_))$-coalgebra homomorphism $(X, \phi) \to (A^{\mathbb{N}}, \alpha)$. For an endofunctor F, we write $\nu X.F(X)$ for the final F-coalgebra when it exists. Then, $A^{\mathbb{N}}$ can be denoted as $\nu X.A \times X$.

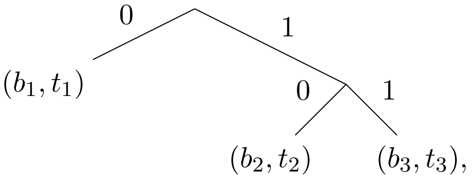
Here, we are interested in functions that translate infinite data to infinite data, e.g., functions from A-streams to B-streams (A and B are sets). One might say stream processors to mean all functions

[1] Email: ytomoya@kurims.kyoto-u.ac.jp

$A^{\mathbb{N}} \to B^{\mathbb{N}}$, but we prefer to define stream processors as *productive* functions $A^{\mathbb{N}} \to B^{\mathbb{N}}$. Productivity of a map $f : A^{\mathbb{N}} \to B^{\mathbb{N}}$ in this context means that f will read only finite information of a input stream $\overrightarrow{a}$ to decide finite information of the output stream $f(\overrightarrow{a})$. This constraint is reasonable from a computational perspective because programs can read only finite information of infinite data. Thus if a program uses the stream $f(\overrightarrow{a})$ then it in fact uses a finite segment of this stream and the finite segment of $f(\overrightarrow{a})$ should be computed using finite information of $\overrightarrow{a}$.

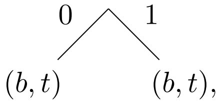
As described in [7,5], this productivity can be characterized as the continuity of functions. The set $A^{\mathbb{N}}$ has a natural topology, the product topology of discrete spaces A, and the set of productive maps $A^{\mathbb{N}} \to B^{\mathbb{N}}$ coincides with the set of continuous maps $A^{\mathbb{N}} \to B^{\mathbb{N}}$. Though this characterization may be mathematically elegant, by computational requirements, we want a coalgebraic characterization of these continuous functions.

There already have been several studies on coalgebraic representation of continuous functions between final coalgebras [7,6,2]. In particular, Ghani, Hancock and Pattinson [7] studied stream processors $A^{\mathbb{N}} \to B^{\mathbb{N}}$ and showed that they can be represented as elements (trees) of another final coalgebra $\nu X.T_A(B \times X)$, where $T_A(V)$ is the set of finite-depth A-ary branching trees with V-labelled leaves. Let $I_{AB} \coloneqq \nu X.T_A(B \times X)$. Then each element of I_{AB} can be seen (coinductively) as a tree in $T_A(B \times I_{AB})$. For example, when $A = \{0, 1\}$, the following tree t belongs to I_{AB}



where $b_i \in B$ and t_i's are trees in I_{AB} $(i = 1, 2, 3)$. Here we give an overview of how t expresses a function $\{0, 1\}^{\mathbb{N}} \to B^{\mathbb{N}}$. For a given stream $a_0 a_1 \cdots \in \{0, 1\}^{\mathbb{N}}$, this tree t outputs a B-valued stream stepwise as follows. First, t consumes a_0. If $a_0 = 0$, it reaches the leaf (b_1, t_1). Thus it outputs b_1 and then the computation continues in the same way with t_1 and $a_1 a_2 \cdots$ (instead of t and $a_0 a_1 \cdots$). On the other hand, if $a_0 = 1$, it does not reach leaves. Therefore t also consumes a_1 and reaches the leaf (b_2, t_2) or (b_3, t_3) if a_1 is 0 or 1, respectively. Then the first output is b_2 or b_3 and computation continues with either t_2 and $a_2 a_3 \cdots$ or t_3 and $a_2 a_3 \cdots$. Eventually, t will give a continuous map $\{0, 1\}^{\mathbb{N}} \to B^{\mathbb{N}}$ because the above procedure produces each digit of the resulting stream using a finite initial segment of the input stream. However, this representation is *not bijective* in the sense that different trees may give the same function. For instance, consider trees t and t' such that

$t =$



$t' = (b, t')$.

Then both of them give a constant map which sends every stream to $bbb \cdots$ (in particular, the tree t', which has no branches, does not consume elements to output the first digit b and thus it does not consume an input stream at all to output the result $bbb \cdots$).

Garner [5] paid attention to this non-bijectivity and he gave a bijective characterization of stream processors in terms of *algebraic theories* and their *comodels*. The set $A^{\mathbb{N}}$ arises as the *final comodel* of the algebraic theory of *A-valued input* $\mathbb{T}_A$ and the set of stream processors $A^{\mathbb{N}} \to B^{\mathbb{N}}$ appears as the *final* $\mathbb{T}_A$-$\mathbb{T}_B$-*bimodel*. He also gave a comodel-theoretic expression of the result of [7]; the final coalgebra $\nu X.T_A(B \times X)$ is the *final* $\mathbb{T}_A$-*residual* $\mathbb{T}_B$-*comodel* and, because of their finalities, there are canonical maps between the final $\mathbb{T}_A$-residual $\mathbb{T}_B$-comodel and the final $\mathbb{T}_A$-$\mathbb{T}_B$-bimodel such that their composition is the identity function on the final $\mathbb{T}_A$-$\mathbb{T}_B$-bimodel.

In this paper, we generalize Garner's result to the case of *free* algebraic theories, which are algebraic theories with no *equational axioms*. We analyse continuous functions on final comodels of free algebraic theories, investigate residual comodels and bimodels of free algebraic theories, and relate them to each other. If a free algebraic theory $\mathbb{T}$ has the *signature* $\Sigma = \{\sigma_1, \ldots, \sigma_n\}$, its final comodel $S_{\mathbb{T}}$ comprises infinite-depth n-ary branching trees with labels determined by the signature Σ. This is a generalization of the case of A-valued streams in the sense that A-valued streams can be seen as infinite-depth 1-ary branching trees with labels in A. Thus continuous functions $S_{\mathbb{T}} \to S_{\mathbb{T}'}$ between final comodels of free algebraic theories $\mathbb{T}, \mathbb{T}'$ translate trees to trees and they also observe only finite information of input trees to decide finite information of output trees (topologies on $S_{\mathbb{T}}$ and $S_{\mathbb{T}'}$ are defined in Section 3).

The key point in this generalization is that these continuous functions can observe *parallel* information such as sibling nodes in trees. On the other hand, elements of the final $\mathbb{T}$-residual $\mathbb{T}'$-comodel can observe only *serial* (in other words, *straight* or *successive*) finite information of input trees. Therefore we must restrict ourselves to consider only such functions (we call them *straight functions* tentatively). Here, as the case of streams, different elements can give the same straight function. This non-bijectiveness will be eliminated when we consider the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. Consequently, we get the following diagram:

$$\begin{tikzcd}
& \text{straight functions } S_{\mathbb{T}} \to S_{\mathbb{T}'} \arrow[dr, equal] & \\
\text{the final } \mathbb{T}\text{-residual } \mathbb{T}'\text{-comodel} \arrow[ur, "\mathsf{reflect}"] & & \text{the final } \mathbb{T}\text{-}\mathbb{T}'\text{-bimodel.} \arrow[ll, "\mathsf{reify}"']
\end{tikzcd}$$

And we will show that the composition $\mathsf{reflect} \circ \mathsf{reify}$ is the identity on the final $\mathbb{T}$-$\mathbb{T}'$-bimodel when it is identified with the set of straight functions. Although the restriction that we consider only straight functions might seem artificial, the correspondence we will construct is in fact a generalization of [5] because there is no parallel information in the case of streams. Note that, to realize our restriction for maps, we will consider a category *Sub* instead of the usual category *Top* of topological spaces and continuous functions. Then the set of straight functions $S_{\mathbb{T}} \to S_{\mathbb{T}'}$ can be identified with a hom-set $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$.

1.2 Outline of The Paper

This paper is organized as follows.

In Section 2, we review basic notions of algebraic theories, their models, and comodels. Especially, we give examples of comodels to help with our intuition.

In Section 3, we discuss topological notions on comodels. Following Garner [5]'s *operational topology*, we define the *operational sub-basis*. Since we mainly use sub-basis, we define the category *Sub* of sets with sub-basis. Additionally, we observe that final comodels in *Set* become final comodels in *Sub*.

In Section 4, we disucuss residual comodels and bimodels. Each elements of the final residual comodel constructs a function between two final comodels. This construction will be named $\mathsf{reflect}$. We also define a map reify from the final bimodel to the final residual comodel.

In Section 5, we show the main result of this paper: $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ can appear as the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. Though the proof is long and might look complex, the technique and key ideas are similar to the case of stream processors [5]. Difficulty comes from the existence of parallel information in trees, which constitute final comodels of free theories. At the end of this section, we explain that the composition $\mathsf{reflect} \circ \mathsf{reify}$ becomes the identity on $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$.

In Section 6, we summarize our results and describe directions for future work.

We use the following notations.

- $\mathbb{N} := \{0, 1, 2, \ldots\}$ is the set of natural numbers.
- For a set A, $A^{\mathbb{N}}$ denotes the set of infinite sequences of elements of A and A^* denotes the set of finite sequences of elements of A. Their (disjoint) union $A^* \cup A^{\mathbb{N}}$ is denoted by $A^{\leq \mathbb{N}}$. The empty sequence in A^* is written as ϵ.
- For a finite sequence $s \in A^*$ and a finite or infinite sequence $t \in A^{\leq \mathbb{N}}$, $s \cdot t$ denotes the concatenation of s, t.
- Usually we use V, W as sets of variables, and variables are denoted by v, w, v', v_1, v_2, etc.

1.3 Related Work

References for (co)algebras include [10,11,17], in which various usages of coalgebras in computer science are explained; one is as implementations of infinite objects and another one is as representations of transition systems. On the other hand, comodels also can be seen as transition systems. As described in [16,15,18], comodels of an algebraic theory $\mathbb{T}$ provide environments for evaluating $\mathbb{T}$-terms. Applications of algebraic theories in the study of computational effects originated in Plotkin and Power [14], and originally, such categorical studies of computational effects were founded by Moggi [13], in which he advocated that various kinds of computational effects, such as exception and nondeterminism, can be modeled by monads.

Researches on coalgebraic characterizations of continuous functions on final coalgebras originated in Ghani et al. [7], which is about stream processors. After that, in [6], they generalized their results to the case of final coalgebras of functors called *containers*. In their work, remaining problems were non-bijectiveness of representations and verification of *completeness* in the latter case, i.e., it is not resolved whether all continuous functions can be expressed by their representations (however, for the case of stream processors, completeness was already proved). Then Garner [5] gave a bijective characterization of stream processors and reformulated the result of [7] in terms of algebraic theories and their comodels (in particular, he used notions of residual comodels and bimodels). Our study generalizes Garner's techniques to the case of free algebraic theories. An advantage of comodel-theoretic characterization is the easy verification of completeness, which will be simply done by the universality of final objects. Therefore we expect that this technique is useful to give complete characterizations in more general cases.

As for stream processors, there exists another well-known related concept called transducers [8], which is a generalization of automata. In [1], Beal and Carton argued when functions $A^{\mathbb{N}} \to B^{\mathbb{N}}$ realized by transducers become continuous (in their paper, A and B were assumed to be finite). In a recent work [9], Hyvernat used type-theoretic transducers to represent continuous functions between coinductive types (note that completeness of representations is not verified yet). According to Hyvernat, observing parallel information in trees is equivalent to the backtracking of transducers. This insight leads us to an idea that, if there is a comodel-theoretic characterization of backtracking, we might be able to characterize not only straight functions but also arbitrary continuous functions between final comodels.

2 Algebraic Theories and Their (Co)models

In this section, we review basic concepts and notations for algebraic theories and their models as well as comodels. We mostly follow Garner's treatment in [5,4].

Definition 2.1 An *algebraic theory* $\mathbb{T}$ is a pair $(\Sigma_{\mathbb{T}}, E_{\mathbb{T}})$, where $\Sigma_{\mathbb{T}}$ is a *signature* and $E_{\mathbb{T}}$ is a set of *equations* over $\Sigma_{\mathbb{T}}$. A *signature* comprises a set Σ of operation symbols, and for each $\sigma \in \Sigma$ a set $|\sigma|$, its *arity*. Given a signature Σ and a set V, we define the set $\Sigma(V)$ of Σ*-terms with variables in* V by the inductive clauses

$$v \in V \Rightarrow v \in \Sigma(V),$$
$$\sigma \in \Sigma,\ t_i \in \Sigma(V)\ (i \in |\sigma|) \Rightarrow \sigma(\lambda i.t_i) \in \Sigma(V).$$

An *equation over a signature* Σ is a formal equality $t = u$ between terms in the same set of free variables. We say $\mathbb{T}$ is *free* if it has no equation, i.e., if $E_{\mathbb{T}} = \emptyset$.

We usually say "theory" to mean "algebraic theory".

Definition 2.2 For a signature Σ and a term $t \in \Sigma(V)$ and terms $u_v \in \Sigma(W)$ $(v \in V)$, we define the *substitution* $t(\lambda v.u_v) \in \Sigma(W)$ by recursion on t:

$$v \in V \Rightarrow v(\lambda v.u_v) := u_v,$$
$$\sigma \in \Sigma,\ t_i \in \Sigma(V)\ (i \in |\sigma|) \Rightarrow (\sigma(\lambda i.t_i))(\lambda v.u_v) := \sigma(\lambda i.t_i(\lambda v.u_v)).$$

Given a theory $\mathbb{T}$, we define $\mathbb{T}$-*equivalence* as the smallest family of substitution-congruence $\equiv_{\mathbb{T}}$ on the sets $\Sigma_{\mathbb{T}}(V)$ such that $t \equiv_{\mathbb{T}} u$ for all equations $t = u \in E_{\mathbb{T}}$. The set $T(V)$ of $\mathbb{T}$-*terms with variables in* V is the quotient $\Sigma(V)/\equiv_{\mathbb{T}}$.

When writing $\sigma(\lambda i.t_i)$ for a symbol σ, we assume that the variable i ranges over $|\sigma|$.

When we see a theory $\mathbb{T}$ as specifying a computational effect as advocated in [14], $T(V)$ is seen as the set of computations with effects from $\mathbb{T}$ returning a value in V. Well-known examples are theories for effects of output, state, exception, nondeterminism, and so on. In this article, we are mainly interested in the theory of input and its expansions (in short, we basically consider free theories).

Example 2.3 [5] Given a set A, the theory $\mathbb{T}_A$ of *A-valued input* comprises a single A-ary operation symbol $\mathbf{read}_A$ and no equations. The set of terms $T_A(V)$ is the initial algebra $\mu X.V + X^A$. Its elements may be seen as A-ary branching trees with leaves labelled in V; or, from another perspective, they can be seen as programs which request A-values and use them to determine a return value in V.

Example 2.4 Consider a free theory $\mathbb{T}$ with n operation symbols $\sigma_1, \dots, \sigma_n$. For all i, we write $|\sigma_i| = A_i$. ($\mathbb{T}_A$ is the case $n = 1$ and $|\sigma_1| = A$.) The set of terms $T(V)$ is the initial algebra $\mu X.V + \coprod_{\sigma \in \Sigma_\mathbb{T}} X^{|\sigma|}$, where $\coprod$ denotes coproduct or direct sum. Its elements can be thought of as trees such that each node is labelled by a symbol σ_i and such a node has A_i-ary branches and finally, their leaves are labelled in V (or nullary operation symbols if they exist). Computationally, they are programs which request n-sorted values and return a value in V depending on inputs.

In particular, we write $\mathbb{T}_A^{(n)}$ for the free theory with n symbols and $|\sigma_1|, \dots, |\sigma_n|$ are all the same set A. As trees, elements of $T_A^{(n)}(V)$ has the same form as elements of $T_A(V)$ but their nodes have labels in $\{1, \dots, n\}$. Differences between $\mathbb{T}_A^{(n)}$ for $n \geq 2$ and $\mathbb{T}_A$ will become more significant when we consider their *comodels*.

Definition 2.5 Let Σ be a signature and $\mathscr{C}$ be a category with powers. A *Σ-structure* $\boldsymbol{X}$ in $\mathscr{C}$ is an object $X \in \mathscr{C}$ with an *operation* $[\![\sigma]\!]_{\boldsymbol{X}} : X^{|\sigma|} \to X$ for each $\sigma \in \Sigma$. For each $t \in \Sigma(V)$ the *derived operation* $[\![t]\!]_{\boldsymbol{X}} : X^V \to X$ is determined by the recursive clauses:

$$[\![v]\!]_{\boldsymbol{X}} = \pi_v \quad \textit{and} \quad [\![\sigma(\lambda i.t_i)]\!]_{\boldsymbol{X}} = X^V \xrightarrow{([\![t_i]\!]_{\boldsymbol{X}})_{i \in |\sigma|}} X^{|\sigma|} \xrightarrow{[\![\sigma]\!]_{\boldsymbol{X}}} X.$$

Definition 2.6 Given a theory $\mathbb{T}$, a $\mathbb{T}$-*model* in $\mathscr{C}$ is a Σ-structure $\boldsymbol{X}$ which satisfies $[\![t]\!]_{\boldsymbol{X}} = [\![u]\!]_{\boldsymbol{X}}$ for all equations $t = u$ of $\mathbb{T}$. $\mathbb{T}$-models in $\mathscr{C}$ form a category with morphisms $f : X \to Y$ in $\mathscr{C}$ such that the following diagram commutes for all $\sigma \in \Sigma$:

$$\begin{tikzcd}
X^{|\sigma|} \arrow[r, "{[\![\sigma]\!]_{\boldsymbol{X}}}"] \arrow[d, "f^{|\sigma|}"'] & X \arrow[d, "f"] \\
Y^{|\sigma|} \arrow[r, "{[\![\sigma]\!]_{\boldsymbol{Y}}}"] & Y
\end{tikzcd}$$

The unqualified term "model" will mean "model in *Set*". We write $\mathit{Mod}(\mathbb{T}, \mathscr{C})$ for the category of $\mathbb{T}$-models in $\mathscr{C}$, and $\mathit{Mod}(\mathbb{T})$ for $\mathit{Mod}(\mathbb{T}, \mathit{Set})$.

The set of terms $T(V)$ has a $\mathbb{T}$-model structure given by substitution. This structure has the following universal property.

Lemma 2.7 *The set of terms $T(V)$ is the free $\mathbb{T}$-model on V by the inclusion of variables $\eta_V : V \to T(V)$. That is, for any $\mathbb{T}$-model $\boldsymbol{X}$ and any function $f : V \to X$ to the underlying set of $\boldsymbol{X}$, there is the unique $\mathbb{T}$-model morphism $f^\dagger : \boldsymbol{T}(V) \to \boldsymbol{X}$ with $f^\dagger \circ \eta_V = f$. Spelling out the detail, we have $f^\dagger(t) = [\![t]\!]_{\boldsymbol{X}}(\lambda v.f(v))$.*

This lemma allows us to define the *Kleisli category* of $\mathbb{T}$.

Definition 2.8 For an algebraic theory $\mathbb{T}$, the *Kleisli category* $\mathit{Kl}(\mathbb{T})$ of $\mathbb{T}$ has sets as objects. For sets A, B, the hom-set $\mathit{Kl}(\mathbb{T})(A, B)$ is defined as $\mathit{Set}(A, T(B))$. The identity at A is $\eta_A : A \to T(A)$. Composition of $f : A \to T(B)$ and $g : B \to T(C)$ is $g^\dagger \circ f$ with $g^\dagger$ as in Lemma 2.7.

There are well-known functors related to $\mathit{Kl}(\mathbb{T})$; the free functor $F_\mathbb{T} : \mathit{Set} \to \mathit{Kl}(\mathbb{T})$ is the identity on objects and sends $f \in \mathit{Set}(X, Y)$ to $\eta_Y \circ f \in \mathit{Kl}(\mathbb{T})(X, Y)$. the comparison functor $I_\mathbb{T} : \mathit{Kl}(\mathbb{T}) \to \mathit{Mod}(\mathbb{T})$ acts as $A \mapsto \boldsymbol{T}(A)$ and $f \mapsto f^\dagger$.

We now turn to *comodel* which is the dual notion of model.

Definition 2.9 Let $\mathbb{T}$ be a theory. A $\mathbb{T}$-*comodel* $\boldsymbol{S}$ in a category $\mathscr{C}$ with copowers is a model of $\mathbb{T}$ in $\mathscr{C}^{op}$, i.e. an object $S \in \mathscr{C}$ with co-operations $[\![\sigma]\!]^{\boldsymbol{S}} : S \to |\sigma| \cdot S$ satisfying the equations of $\mathbb{T}$. Morphisms between comodels $\boldsymbol{S}$, $\boldsymbol{S'}$ are morphisms $f : S \to S'$ in $\mathscr{C}$ such that the following diagram commutes for each symbol σ in $\Sigma_{\mathbb{T}}$:

$$\begin{tikzcd}
S \arrow[r, "{[\![\sigma]\!]^{\boldsymbol{S}}}"] \arrow[d, "f"'] & {|\sigma| \cdot S} \arrow[d, "{|\sigma|\cdot f}"] \\
S' \arrow[r, "{[\![\sigma]\!]^{\boldsymbol{S'}}}"] & {|\sigma| \cdot S'}
\end{tikzcd}$$

The unqualified term "comodel" will mean "comodel in *Set*". We write $\mathit{Comod}(\mathbb{T}, \mathscr{C})$ for the category of $\mathbb{T}$-comodels in $\mathscr{C}$, and $\mathit{Comod}(\mathbb{T})$ for $\mathit{Comod}(\mathbb{T}, \mathit{Set})$. We here note that $\mathit{Comod}(\mathbb{T}, \mathscr{C}) \cong \mathit{Mod}(\mathbb{T}, \mathscr{C}^{op})^{op}$, the opposite of the category of $\mathbb{T}$-models in the opposite of $\mathscr{C}$.

Example 2.10 A comodel $\boldsymbol{S}$ of the theory $\mathbb{T}_A$ of A-valued input is a set S with a map $[\![\mathbf{read}_A]\!]^{\boldsymbol{S}} : S \to A \times S$. This map can be decomposed into two maps: the output map $o^{\boldsymbol{S}} : S \to A$ and the transition map $\partial^{\boldsymbol{S}} : S \to S$.

We usually call elements of comodels *states*. Then a comodel $\boldsymbol{S}$ of $\mathbb{T}_A$ can be seen as a *state machine* which answers to requests for A-value and transition to the next state, determined by $o^{\boldsymbol{S}}$ and $\partial^{\boldsymbol{S}}$, respectively.

As explained in [16,15,18], when a theory $\mathbb{T}$ presents a computational effect, its comodels provide deterministic environments for evaluating computations with effects from $\mathbb{T}$. In general, given a $\mathbb{T}$-comodel $\boldsymbol{S}$ and a term $t \in T(V)$, we have *derived co-operation* $[\![t]\!]^{\boldsymbol{S}} : S \to V \times S$ as the dual of derived operation in Definition 2.5:

$$[\![v]\!]^{\boldsymbol{S}}(s) = \iota_v(s) = (v, s)$$
$$[\![\sigma(\lambda i.t_i)]\!]^{\boldsymbol{S}}(s) = [\![t_i]\!]^{\boldsymbol{S}}(s') \text{ where } [\![\sigma]\!]^{\boldsymbol{S}}(s) = (i, s').$$

Intuitively, evaluating a term (or a tree) t with an initial state s is selection of a path to a value in t depending on the behavior of s as follows: first, if $t = \sigma(\lambda i.t_i)$ and $[\![\sigma]\!]^{\boldsymbol{S}}(s) = (i, s')$, s chooses the i-th branch and transition to the next state s'; then continue the computation by evaluating t_i with s'; finally, if the term under the chosen branch is a value, then the evaluation terminates and returns that value. Clearly, return values of a term t appear in t as variables.

We focus on the final comodel of a theory. The final comodel of $\mathbb{T}$ is the final object of $\mathit{Comod}(\mathbb{T})$. We will also consider the final comodel in a category $\mathscr{C}$ other than *Set*, that is, the final object of $\mathit{Comod}(\mathbb{T}, \mathscr{C})$.

Example 2.11 The final comodel of $\mathbb{T}_A$ is $\boldsymbol{A}^{\mathbb{N}}$, the set of infinite sequences of elements in A. Its co-operation $[\![\mathbf{read}]\!]^{\boldsymbol{A}^{\mathbb{N}}} : A^{\mathbb{N}} \to A \times A^{\mathbb{N}}$ is composed of

$$o^{A^{\mathbb{N}}} : a_0 a_1 a_2 \cdots \mapsto a_0 \quad \partial^{A^{\mathbb{N}}} : a_0 a_1 a_2 \cdots \mapsto a_1 a_2 \cdots .$$

In order to help understanding the next example, we see a sequence in $A^{\mathbb{N}}$ in a slightly different way. Firstly, we see a sequence $\overrightarrow{a} \in A^{\mathbb{N}}$ as the function $a : \mathbb{N} \to A$ with $a(k) = a_k$. We can get the k-th element $a(k)$ by applying $[\![\mathbf{read}]\!]^{\boldsymbol{A}^{\mathbb{N}}}$ (more precisely, applying its component $\partial^{A^{\mathbb{N}}}$) k-times and taking the head element of the resulting sequence. So, it is reasonable to see the domain of the function a as $\{\mathbf{read}\}^* = \{\epsilon, \mathbf{read}, \mathbf{read} \cdot \mathbf{read}, \ldots\}$, the set of finite repeats of the symbol $\mathbf{read}$. Then components of $[\![\mathbf{read}]\!]^{\boldsymbol{A}^{\mathbb{N}}}$ can be written as:

$$o^{A^{\mathbb{N}}} : a \mapsto a(\epsilon) \quad \partial^{A^{\mathbb{N}}} : a \mapsto a(\mathbf{read}\_)$$

where $(a(\mathbf{read}\_))(\mathbf{read}^k) = a(\mathbf{read} \cdot \mathbf{read}^k) = a(\mathbf{read}^{k+1})$.

We recognize comodels and the final comodel of a free theory in a similar way to the case of the theory of input.

Example 2.12 Let $\mathbb{T}$ be a free theory. A comodel of $\mathbb{T}$ is a state machine that answers to requests for elements of $|\sigma|$ for each $\sigma \in \Sigma_{\mathbb{T}}$, and then transition to a next state depending on the requested operation symbol. This comprises a set of states S and operations $[\![\sigma]\!]^{\boldsymbol{S}} = (o^{\boldsymbol{S}}_{\sigma}, \partial^{\boldsymbol{S}}_{\sigma}) : S \to |\sigma| \times S$ giving for each state $s \in S$ an output $o^{\boldsymbol{S}}_{\sigma}(s) \in |\sigma|$ and a next state $\partial^{\boldsymbol{S}}_{\sigma}(s) \in S$. By taking the product of $[\![\sigma]\!]^{\boldsymbol{S}}$'s, we can regard the comodel $\boldsymbol{S}$ as a coalgebra of the functor $H_{\mathbb{T}} := \prod_{\sigma\in\Sigma_{\mathbb{T}}}(|\sigma| \times (\_)) \cong (\prod_{\sigma\in\Sigma_{\mathbb{T}}}|\sigma|) \times (\_)^{\Sigma_{\mathbb{T}}}$. Moreover, morphisms between $\mathbb{T}$-comodels coincide with those between $H_{\mathbb{T}}$-coalgebras. Therefore the final $\mathbb{T}$-comodel $\boldsymbol{S}_{\mathbb{T}}$ is the final $H_{\mathbb{T}}$-coalgebra $\nu X.(\prod_{\sigma\in\Sigma_{\mathbb{T}}}|\sigma|) \times X^{\Sigma_{\mathbb{T}}} = (\prod_{\sigma\in\Sigma_{\mathbb{T}}}|\sigma|)^{\Sigma_{\mathbb{T}}^{*}}$. The comodel structure of $\boldsymbol{S}_{\mathbb{T}}$ is given by

$$o^{\boldsymbol{S}_{\mathbb{T}}}_{\sigma}(s) = \pi_{\sigma}(s(\epsilon)) \qquad \partial^{\boldsymbol{S}_{\mathbb{T}}}_{\sigma}(s) = s(\sigma\_)$$

where π_σ denotes the projection map $(\prod_{\sigma\in\Sigma_{\mathbb{T}}}|\sigma|) \to |\sigma|$.

We omit indices $\boldsymbol{X}$, $\boldsymbol{S}$ of $[\![\sigma]\!]_{\boldsymbol{X}}$, $[\![\sigma]\!]^{\boldsymbol{S}}$, $o^{\boldsymbol{S}}_{\sigma}$ and $\partial^{\boldsymbol{S}}_{\sigma}$, if it is clear from contexts.

An important property of the final comodel is that it describes "observable behaviors" of states in comodels. As described in [4], the final comodel of the theory $\mathbb{T}$ can be characterized as the set of all possible states of all possible comodels modulo *operational equivalence*.

Definition 2.13 Let $\mathbb{T}$ be an algebraic theory. For states $s_1 \in \boldsymbol{S}_1$, $s_2 \in \boldsymbol{S}_2$ of two $\mathbb{T}$-comodels, we say that they are *operationally equivalent* if for all $\mathbb{T}$-terms t, $\pi_V([\![t]\!]^{\boldsymbol{S}_1}(s_1)) = \pi_V([\![t]\!]^{\boldsymbol{S}_2}(s_2))$.

Lemma 2.14 ([4]) *States $s_1 \in \boldsymbol{S}_1$ and $s_2 \in \boldsymbol{S}_2$ of two $\mathbb{T}$-comodels are operationally equivalent iff they become equal under each unique map to the final $\mathbb{T}$-comodel.*

3 Operational Sub-basis on Comodels

Garner [5] defined the *operational topology* on a comodel of an algebraic theory $\mathbb{T}$ as the topology whose basic open sets describe those states which are indistinguishable with respect to a finite set of $\mathbb{T}$-computations.

Definition 3.1 [5] Let $\boldsymbol{S}$ be a $\mathbb{T}$-comodel. The *operational topology* on S is generated by sub-basic open sets

$$[t \mapsto v]_{\boldsymbol{S}} := \{s \in S \mid [\![t]\!]^{\boldsymbol{S}}(s) = (v, s') \text{ for some } s'\} \qquad \text{for all } t \in T(V) \text{ and } v \in V.$$

We omit the subscript $\boldsymbol{S}$ of $[t \mapsto v]_{\boldsymbol{S}}$ if the comodel is clear.

We call the sub-basis $\{[t \mapsto v]_{\boldsymbol{S}} \mid t : \text{term}, v : \text{variable}\}$ of the operational topology, the *operational sub-basis* of the comodel $\boldsymbol{S}$. In particular, for the final comodel of a theory $\mathbb{T}$, we write $\Phi_{\mathbb{T}}$ for its operational sub-basis. In the sequel, we mainly use this operational sub-basis, not the operational topology. Thus we define a category whose objects are sets with sub-basis.

Definition 3.2 We define the category *Sub* of *sets with sub-basis and functions continuous on sub-basis.* Objects are pairs (X, Φ), X is a set and Φ is a subset of $\mathcal{P}(X)$ which contains $\emptyset$ and X. A morphism $(X, \Phi) \to (Y, \Psi)$ is a mapping $f : X \to Y$ such that for each set $U \in \Psi$, $f^{-1}(U) \in \Phi$. We call such maps as *continuous functions on sub-basis.*

We can show that *Sub* is a cocomplete category and its constructions of colimits are similar to colimits in *Top*. As for comodels in *Sub*, the following is valid because we can show that, if a set with sub-basis (X, Φ) has a $\mathbb{T}$-comodel structure in *Sub*, then the sub-basis Φ contains all sets of the form $[t \mapsto v]$.

Proposition 3.3 *There is an adjunction*

$$\begin{tikzcd} \mathit{Comod}(\mathbb{T}, \mathit{Sub}) \arrow[r, shift right, "U"'] \arrow[r, phantom, "\top"] & \mathit{Comod}(\mathbb{T}) \arrow[l, shift right, "op"'] \end{tikzcd}$$

where op gives a comodel $\boldsymbol{S}$ the operational sub-basis, and U forgets sub-basis.

As a corollary, the functor *op* preserves the final object (A similar result for the category *Top* is in [5]).

Corollary 3.4 *For each theory $\mathbb{T}$, the final $\mathbb{T}$-comodel in Set is, when endowed with its operational sub-basis, the final comodel in Sub.*

4 Residual Comodel and Continuous Functions

Residual comodels allow us to describe stateful translations between different notions of computation. The word *residual* comes from [12].

Definition 4.1 [5] Let $\mathbb{T}$ and $\mathbb{T}'$ be theories. An $\mathbb{T}$-*residual* $\mathbb{T}'$-*comodel* is a comodel of $\mathbb{T}'$ in the Kleisli category $\mathit{Kl}(\mathbb{T})$.

Spelling out the detail, an $\mathbb{T}$-residual $\mathbb{T}'$-comodel $\boldsymbol{S}$ comprises an underlying set S and a co-operation $[\![\sigma]\!]^{\boldsymbol{S}} : S \to T(|\sigma| \times S)$ for each symbol $\sigma \in \Sigma_{\mathbb{T}'}$. That is, for each state $s \in S$ and each symbol σ, we need to deal with an $\mathbb{T}$-computation in order to decide what state we should transition to and to extract an index in $|\sigma|$ from $[\![\sigma]\!]^{\boldsymbol{S}}(s)$. The derived co-operation $[\![t]\!]^{\boldsymbol{S}} : S \to T(V \times S)$ of a term $t \in T'(V)$ is calculated using composition in the Kleisli category:

$$\begin{aligned}
&[\![v]\!]^{\boldsymbol{S}}(s) = (v, s) \in V \times S \subseteq T(V \times S)\\
&[\![\sigma(\lambda i.t_i)]\!]^{\boldsymbol{S}}(s) = [\![\sigma]\!]^{\boldsymbol{S}}(s)(\lambda(i, s').[\![t_i]\!]^{\boldsymbol{S}}(s')),
\end{aligned}$$

where the term $[\![\sigma]\!]^{\boldsymbol{S}}(s)(\lambda(i, s').[\![t_i]\!]^{\boldsymbol{S}}(s')) \in T(V \times S)$ is the substitution of $([\![t_i]\!]^{\boldsymbol{S}}(s'))_{(i,s')} \in T(V \times S)^{|\sigma| \times S}$ to $[\![\sigma]\!]^{\boldsymbol{S}}(s) \in T(|\sigma| \times S)$.

By the above intuition about $\mathbb{T}$-residual $\mathbb{T}'$-comodels, we expect that when we have a state s of an $\mathbb{T}$-residual $\mathbb{T}'$-comodel $\boldsymbol{S}$ and a state m of an $\mathbb{T}$-comodel, then, for each term $t \in T'(V)$, we can evaluate $[\![t]\!]^{\boldsymbol{S}}(s) \in T(V \times S)$ with the initial state m to a value-state pair in $V \times S$. This idea is formalized as:

Definition 4.2 [5] Let $\mathbb{T}$, $\mathbb{T}'$ be theories. Let $\boldsymbol{S}$ be an $\mathbb{T}$-residual $\mathbb{T}'$-comodel, and let $\boldsymbol{M}$ be an $\mathbb{T}$-comodel. The *tensor product* $\boldsymbol{S} \cdot \boldsymbol{M}$ is the $\mathbb{T}'$-comodel with underlying set $S \times M$ and co-operations

$$[\![\sigma]\!]^{\boldsymbol{S}\cdot\boldsymbol{M}} : S \times M \xrightarrow{[\![\sigma]\!]^{\boldsymbol{S}} \times M} T(|\sigma| \times S) \times M \xrightarrow{(t,m) \mapsto [\![t]\!]^{\boldsymbol{M}}(m)} |\sigma| \times S \times M.$$

This construction can be generalized to the case that $\boldsymbol{M}$ is a $\mathbb{T}$-comodel in a category $\mathscr{C}$ with copowers. For an $\mathbb{T}$-residual $\mathbb{T}'$-comodel $\boldsymbol{S}$ and an $\mathbb{T}$-comodel $\boldsymbol{M}$ in $\mathscr{C}$, there is a $\mathbb{T}'$-comodel in $\mathscr{C}$ whose underlying object is the copower $S \cdot M$.

If there is the final $\mathbb{T}'$-comodel $\boldsymbol{S}_{\mathbb{T}'}$ in $\mathscr{C}$, $\mathit{Comod}(\mathbb{T}', \mathscr{C})(\boldsymbol{S}\cdot\boldsymbol{M}, \boldsymbol{S}_{\mathbb{T}'})$ has only one map e. Now we consider the case of $\mathscr{C} = \mathit{Sub}$ and $\boldsymbol{M}$ is the final $\mathbb{T}$-comodel $\boldsymbol{S}_{\mathbb{T}}$. By Corollary 3.4, we have this $e : S \times S_{\mathbb{T}} \to S_{\mathbb{T}'}$ called the extent of $\boldsymbol{S}$. Then its currying $\lambda s \in S.e(s, \_) : S \to \mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ translates elements of a residual comodel to continuous functions between two final comodels, in particular, these functions are continuous on sub-basis.

One of our goals is to show that, for free theories $\mathbb{T}$ and $\mathbb{T}'$, the currying of the extent of the *final* $\mathbb{T}$-*residual* $\mathbb{T}'$-*comodel* is surjective. To define the final object, we first define the notion of morphism between residual comodels. This is different from the usual morphism between comodels in the Kleisli category.

Definition 4.3 [5] Let $\mathbb{T}$ and $\mathbb{T}'$ be theories, $\boldsymbol{S}$ and $\boldsymbol{U}$ be $\mathbb{T}$-residual $\mathbb{T}'$-comodels. *A map of residual comodels* $\boldsymbol{S} \to \boldsymbol{U}$ is a function $f : S \to U$ such that the following diagram commutes for each symbol $\sigma \in \Sigma_{\mathbb{T}'}$:

$$\begin{tikzcd}
S \arrow[r, "{[\![\sigma]\!]^{\boldsymbol{S}}}"] \arrow[d, "f"'] & T(|\sigma| \times S) \arrow[d, "T(|\sigma| \times f)"] \\
U \arrow[r, "{[\![\sigma]\!]^{\boldsymbol{U}}}"'] & T(|\sigma| \times U)
\end{tikzcd}$$

Now we define the final $\mathbb{T}$-residual $\mathbb{T}'$-comodel $\boldsymbol{I}_{\mathbb{T},\mathbb{T}'}$ as the final object of the category of residual comodels and maps between residual comodels, which is different from the final $\mathbb{T}'$-comodel in $\mathit{Kl}(\mathbb{T})$.

We call the currying of the extent of $\boldsymbol{I}_{\mathbb{T},\mathbb{T}'}$ the *reflection*.

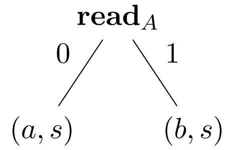
Definition 4.4 Let $\mathbb{T}$ and $\mathbb{T}'$ be free theories, $\boldsymbol{I}_{\mathbb{T},\mathbb{T}'}$ be the final $\mathbb{T}$-residual $\mathbb{T}'$-comodel. The *reflection* function is defined as currying of the extent e of $\boldsymbol{I}_{\mathbb{T},\mathbb{T}'}$:

$$\mathsf{reflect} : I_{\mathbb{T},\mathbb{T}'} \to \mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'}) \qquad s \mapsto e(s, \_) : S_{\mathbb{T}} \to S_{\mathbb{T}'}.$$

Following examples explain how a state of residual comodel implements a function and why we restrict our target to maps in *Sub* (or straight functions in the introduction).

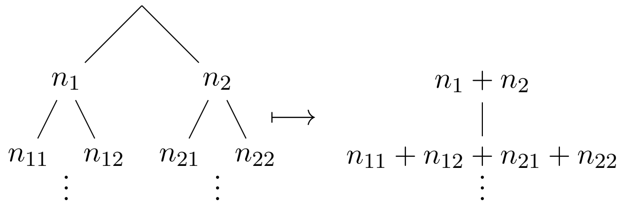
Example 4.5 Let $A = \{0, 1\}$ and $B = \{a, b\}$. Consider a function $A^{\mathbb{N}} \to B^{\mathbb{N}}$ which rewrites 0 to a and 1 to b. We take theories $\mathbb{T}$ and $\mathbb{T}'$ as $\mathbb{T}_A$ and $\mathbb{T}_B$. The final $\mathbb{T}$-residual $\mathbb{T}'$-comodel $I_{\mathbb{T},\mathbb{T}'}$ has its residual comodel structure $[\![\mathbf{read}_B]\!] : I_{\mathbb{T},\mathbb{T}'} \to T(B \times I_{\mathbb{T},\mathbb{T}'})$. Take a state $s \in I_{\mathbb{T},\mathbb{T}'}$ as

$[\![\mathbf{read}_B]\!](s) =$



Then for a given stream, such as $1011\cdots$ in $A^{\mathbb{N}}$, this state constructs a stream $y_1 y_2 \cdots$ in $B^{\mathbb{N}}$ as follows: To compute the first digit y_1, it uses $\mathbf{read}_B$ one time. The tree $[\![\mathbf{read}_B]\!](s)$ requires reading an A-element and this requirement is met by the given stream. So it consumes the first digit 1 of the input and it determines that $y_1 = b$. To compute y_2, it uses $\mathbf{read}_B$ twice. The first $\mathbf{read}_B$ is computed as above and it reaches the leaf (b, s). The second one is applied to this new s and now it consumes the second digit 0 of the input. Thus it reaches (a, s) and y_2 becomes a. The computation continues similarly and it will implement the function considered as above. We can implement more complex maps by deepening the tree or by using other states in leaves.

Example 4.6 This example exhibits a function which cannot be implemented by states of residual comodels. Let $\mathbb{T} = \mathbb{T}^2_{\mathbb{N}}$ and $\mathbb{T}' = \mathbb{T}_{\mathbb{N}}$. Consider a function between final comodels $S_{\mathbb{T}} \to S_{\mathbb{T}'}$ which takes the sum of each depth:



If this can be implemented by a state $s \in I_{\mathbb{T},\mathbb{T}'}$, the first digit $n_1 + n_2$ of the output is computed by the term $[\![\mathbf{read}_{\mathbb{N}}]\!](s) \in T^{(2)}_{\mathbb{N}}(\mathbb{N} \times I_{\mathbb{T},\mathbb{T}'})$. There are three cases; (i) $[\![\mathbf{read}_{\mathbb{N}}]\!](s)$ is a variable $(n, s') \in \mathbb{N} \times I_{\mathbb{T},\mathbb{T}'}$, (ii) $[\![\mathbf{read}_{\mathbb{N}}]\!](s)$ is of the form $\mathbf{read}^1_{\mathbb{N}}(\lambda n.u_n)$ and (iii) $[\![\mathbf{read}_{\mathbb{N}}]\!](s)$ is of the form $\mathbf{read}^2_{\mathbb{N}}(\lambda n.u_n)$. When (i), its output is always n and thus this cannot depend on n_1, n_2. When (ii), it reads n_1 of the input tree, selects the term u_{n_1} and, if u_{n_1} requires further input, it uses the tree under n_1. So, in this case, the output cannot depend on n_2. Similarly, when (iii), the output cannot depend on n_1. Consequently, the term $[\![\mathbf{read}_{\mathbb{N}}]\!](s)$ cannot observe both of n_1 and n_2 and thus the function summing up each depth cannot be implemented by residual comodels.

We will show the surjectivity of the reflection by characterizing $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ as the *final* $\mathbb{T}$-$\mathbb{T}'$*-bimodel*. Here, for theories $\mathbb{T}$ and $\mathbb{T}'$, the category of $\mathbb{T}$-$\mathbb{T}'$-bimodels is the category of $\mathbb{T}'$-comodels in $\mathit{Mod}(\mathbb{T})$, i.e., $\mathit{Comod}(\mathbb{T}', \mathit{Mod}(\mathbb{T}))$. We only describe the most important properties for our purpose; bimodels can be seen as residual comodels.

Lemma 4.7 *Let $\mathbb{T}$ be any theory. For any $\mathbb{T}$-model $\boldsymbol{X} = (X, [\![\_]\!]_{\boldsymbol{X}})$ and set B, the copower $B \cdot \boldsymbol{X}$ is the quotient of the free model $\boldsymbol{T}(B \times X)$ for an $\mathbb{T}$-congruence relation.*

Especially, if $\mathbb{T}$ is a free theory, we can take a canonical representative of each equivalence class and thus $B \cdot X$ may be regarded as a subset of $T(B \times X)$. In detail, the set of canonical representatives coincides with the set of terms in $T(B \times X)$ which have no non-trivial sub-terms whose variables are labelled by the *same element of B, in other words, banned sub-terms are of the form $\sigma(\lambda i.(b, x_i))$ for $b \in B$. In this case, the $\mathbb{T}$-model structure of $B \cdot \boldsymbol{X}$ is that of $T(B \times X)$ except that $[\![\sigma]\!]_{B\cdot\boldsymbol{X}}(\lambda i.(b, x_i)) = (b, [\![\sigma]\!]_{\boldsymbol{X}}(\lambda i.x_i))$.*

Proof. Define a $\mathbb{T}$-congruence $\sim$ on $\boldsymbol{T}(B \times X)$ as the minimal congruence satisfying

$$\sigma(\lambda i.(b, x_i)) \sim (b, [\![\sigma]\!]_{\boldsymbol{X}}(\lambda i.x_i)). \tag{1}$$

for all symbols $\sigma \in \Sigma_{\mathbb{T}}$. The quotient $\boldsymbol{T}(B \times X)/\sim$ satisfies universality of the copower $B \cdot \boldsymbol{X}$.

If $\mathbb{T}$ is a free theory, by orienting read (1) from left to right, this determines a strongly normalizing rewrite system on $T(B \times X)$; if there is a sub-term of the form $\sigma(\lambda i.(b, x_i))$ then rewrite this into the variable $(b, [\![\sigma]\!]_{\boldsymbol{X}}(\lambda i.x_i))$. Thus we can take the normal forms as representatives of equivalence classes. □

Remark 4.8 The latter of this lemma is justified because the set of $\mathbb{T}$-terms $T(B \times X)$ coincides with the set of $\Sigma_{\mathbb{T}}$-terms $\Sigma_{\mathbb{T}}(B \times X)$, whose elements are trees with $\Sigma_{\mathbb{T}}$-labelled nodes and $(B \times X)$-labelled leaves. When $\mathbb{T}$ has non-trivial equations, $T(B \times X)$ is a quotient of $\Sigma_{\mathbb{T}}(B \times X)$. Thus we must argue about rewriting systems on a quotient set and cannot generalize this lemma simply.

Definition 4.9 Let $\mathbb{T}$ and $\mathbb{T}'$ be free theories and $\boldsymbol{K}$ be a $\mathbb{T}$-$\mathbb{T}'$-bimodel. We define the $\mathbb{T}$-residual $\mathbb{T}'$-comodel $\check{\boldsymbol{K}} = (K, [\![\_]\!]^{\check{\boldsymbol{K}}})$ whose co-operations are the composites

$$[\![\sigma]\!]^{\check{\boldsymbol{K}}} : K \xrightarrow{[\![\sigma]\!]^{\boldsymbol{K}}} |\sigma| \cdot K \hookrightarrow T(|\sigma| \times K)$$

of the $\mathbb{T}'$-comodel structure map with the inclusion of Lemma 4.7.

Definition 4.10 Let $\mathbb{T}$ and $\mathbb{T}'$ be free theories and $\boldsymbol{E}_{\mathbb{T},\mathbb{T}'}$ be the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. Then we have the unique $\mathbb{T}$-residual $\mathbb{T}'$-comodel map $\check{\boldsymbol{E}}_{\mathbb{T},\mathbb{T}'} \to \boldsymbol{I}_{\mathbb{T},\mathbb{T}'}$ and we define the *reification* function as its underlying map:

$$\mathsf{reify} : E_{\mathbb{T},\mathbb{T}} \to I_{\mathbb{T},\mathbb{T}'}.$$

If we characterize $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ as the final $\mathbb{T}$-$\mathbb{T}'$-bimodel and if we show $\mathsf{reflect} \circ \mathsf{reify}$ is the identity map, we can conclude that $\mathsf{reflect}$ is surjective.

5 The Final Bimodel of Free Theories

In this section, we assume that $\mathbb{T}$ and $\mathbb{T}'$ are free theories (the only exception is in Proposition 5.1). We write $\boldsymbol{S}_{\mathbb{T}}$ and $\boldsymbol{S}_{\mathbb{T}'}$ for their final comodels and we regard them as objects of Sub with their operational sub-bases $\Phi_{\mathbb{T}}$ and $\Phi_{\mathbb{T}'}$.

Our goal in this section is to prove that $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ is the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. This is verified by proving that the functor $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_) : \mathit{Sub} \to \mathit{Mod}(\mathbb{T})$ preserves the final $\mathbb{T}'$-comodel. Here we assert that $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_)$ is actually a functor to $\mathit{Mod}(\mathbb{T})$. For each object $X \in \mathit{Sub}$, the $\mathbb{T}$-model structure on $\mathit{Sub}(S_{\mathbb{T}}, X)$ is

$$\begin{array}{rcccc} [\![\sigma]\!] = \mathit{split}_\sigma : & \mathit{Sub}(S_{\mathbb{T}}, X)^{|\sigma|} & \to & \mathit{Sub}(S_{\mathbb{T}}, X) \\ & \rotatebox[origin=c]{90}{$\in$} & & \rotatebox[origin=c]{90}{$\in$} \\ & (f_i)_{i \in |\sigma|} & \mapsto & \begin{array}{ccc} S_{\mathbb{T}} & \to & X \\ \rotatebox[origin=c]{90}{$\in$} & & \rotatebox[origin=c]{90}{$\in$} \\ s & \mapsto & f_{o_\sigma s}(\partial_\sigma s) \end{array} \end{array} \tag{2}$$

for each $\sigma \in \Sigma_{\mathbb{T}}$.

The outline of the proof is as follows:

(I) The functor $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_) : \mathit{Sub} \to \mathit{Mod}(\mathbb{T})$ has a left adjoint.

(II) $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_)$ preserves copowers of objects which have a *simple* (see Definition 5.3) $\mathbb{T}'$*-comodel* structure.

(III) The final $\mathbb{T}'$-comodel $\boldsymbol{S}_{\mathbb{T}'}$ (endowed with $\Phi_{\mathbb{T}'}$) is a *simple* $\mathbb{T}'$-comodel.

(IV) Conclude the claim by using adjointness in (I).

(I) is established in [3] and [5]. We cite the statement from [5].

Proposition 5.1 *Let $\mathbb{T}$ be a theory (which is not necessarily free). Let $\mathscr{C}$ be a category with copowers and $\boldsymbol{S}$ a $\mathbb{T}$-comodel in $\mathscr{C}$. For any object $C \in \mathscr{C}$, the hom-set $\mathscr{C}(S, C)$ bears a structure of $\mathbb{T}$-model $\mathscr{C}(\boldsymbol{S}, C)$ with operations*

$$[\![\sigma]\!]_{\mathscr{C}(\boldsymbol{S},C)}(\lambda i.S \xrightarrow{f_i} C) = S \xrightarrow{[\![\sigma]\!]^{\boldsymbol{S}}} |\sigma| \cdot S \xrightarrow{\langle f_i \rangle_{i\in|\sigma|}} C$$

where $\langle f_i \rangle_{i\in|\sigma|}$ is the copairing of the f_i's. As C varies, this assignment underlies a functor $\mathscr{C}(\boldsymbol{S}, \_) : \mathscr{C} \to Mod(\mathbb{T})$. If $\mathscr{C}$ is cocomplete, this functor has a left adjoint $(\_) \otimes \boldsymbol{S} : Mod(\mathbb{T}) \to \mathscr{C}$.

Remark 5.2 The $\mathbb{T}$-model structure (2) on $Sub(S_\mathbb{T}, X)$ is given by this proposition with the ordinary $\mathbb{T}$-comodel structure of $\boldsymbol{S}_\mathbb{T}$.

(III) is clear from the definition of *simplicity* and (IV) is shown as follows:
When (I),(II) and (III) have been verified, then we have the adjunction in (I) as

$$(\_) \otimes \boldsymbol{S}_\mathbb{T} \dashv Sub(\boldsymbol{S}_\mathbb{T}, \_) : Sub \to Mod(\mathbb{T}).$$

Since the left adjoint $(\_) \otimes \boldsymbol{S}_\mathbb{T}$ preserves copowers and $Sub(\boldsymbol{S}_\mathbb{T}, \_)$ also preserves copowers of objects X in *Sub* with a *simple* $\mathbb{T}'$-comodel structure, we have the following isomorphism for an arbitrary object $\boldsymbol{Y}$ in $Comod(\mathbb{T}', Mod(\mathbb{T}))$,

$$Comod(\mathbb{T}', Mod(\mathbb{T}))(\boldsymbol{Y}, Sub(\boldsymbol{S}_\mathbb{T}, \boldsymbol{X})) \cong Comod(\mathbb{T}', Sub)(\boldsymbol{Y} \otimes \boldsymbol{S}_\mathbb{T}, \boldsymbol{X}).$$

The final $\mathbb{T}'$-comodel $\boldsymbol{S}_{\mathbb{T}'}$ is *simple*. So we let $\boldsymbol{X} = \boldsymbol{S}_{\mathbb{T}'}$ in above, then

$$Comod(\mathbb{T}', Mod(\mathbb{T}))(\boldsymbol{Y}, Sub(\boldsymbol{S}_\mathbb{T}, \boldsymbol{S}_{\mathbb{T}'})) \cong Comod(\mathbb{T}', Sub)(\boldsymbol{Y} \otimes \boldsymbol{S}_\mathbb{T}, \boldsymbol{S}_{\mathbb{T}'}).$$

By finality of $\boldsymbol{S}_{\mathbb{T}'}$, the right hand side is a singleton. Thus, $Sub(\boldsymbol{S}_\mathbb{T}, \boldsymbol{S}_{\mathbb{T}'})$ is final in $Comod(\mathbb{T}', Mod(\mathbb{T}))$.

Therefore we will concentrate on (II). First, we define the notion of *simple comodels* appearing in (II).

Definition 5.3 We say a comodel $\boldsymbol{S}$ is *simple* if observationally equivalent states are actually identical, or explicitly, if $\boldsymbol{S}$ satisfies following condition for all states s, s':

$$\begin{aligned}&\text{if } o_\sigma(\partial_{\sigma_n} \cdots \partial_{\sigma_1}(s)) = o_\sigma(\partial_{\sigma_n} \cdots \partial_{\sigma_1}(s'))\\ &\text{for all sequences of symbols } \sigma_1, \ldots, \sigma_n \text{ and for all } \sigma,\\ &\text{then } s = s'.\end{aligned}$$

This definition says that a comodel is simple iff it has no proper quotient (this is the original definition of simple comodels in [17]). The final comodel is clearly a simple comodel (this is a justification of (III)).

The statement (II) says that, for a $\mathbb{T}'$-comodel $((X, \Phi), [\![\_]\!]^{\boldsymbol{X}})$ in *Sub* whose underlying comodel $(X, [\![\_]\!]^{\boldsymbol{X}})$ is simple, we have the copower $(\iota_i : X \to I \cdot X)_{i\in I}$ in *Sub*, then maps given by applying the functor $Sub(\boldsymbol{S}_\mathbb{T}, \_)$ to each ι_i

$$(\iota_i \circ (\_) : Sub(\boldsymbol{S}_\mathbb{T}, X) \to Sub(\boldsymbol{S}_\mathbb{T}, I \cdot X))_{i\in I} \tag{3}$$

constitute a copower cocone in $Mod(\mathbb{T})$. We prove this in two steps: uniqueness of the mediating morphism and existence of it.

The difficult part is to prove its uniqueness and this is rephrased as following.

Theorem 5.4 (uniqueness of the mediating morphism) *Let $((X, \Phi), [\![\_]\!]^{\boldsymbol{X}})$ be a $\mathbb{T}'$-comodel in Sub whose underlying comodel $(X, [\![\_]\!]^{\boldsymbol{X}})$ is simple. For its copower $(\iota_i : X \to I \cdot X)_{i\in I}$ in Sub, the family of maps given by applying the functor $Sub(\boldsymbol{S}_\mathbb{T}, \_)$ to each ι_i*

$$(\iota_i \circ (\_) : Sub(\boldsymbol{S}_\mathbb{T}, X) \to Sub(\boldsymbol{S}_\mathbb{T}, I \cdot X))_{i\in I}$$

is jointly epimorphic in the category $Mod(\mathbb{T})$ (that is, if $\mathbb{T}$-model maps $f, g : Sub(\boldsymbol{S}_\mathbb{T}, I \cdot X) \to \boldsymbol{Y}$ satisfy $f \circ (\iota_i \circ (\_)) = g \circ (\iota_i \circ (\_))$ for all $i \in I$, then $f = g$).

Here we describe an overview of the proof:

(i) Let M be the subset of $\mathit{Sub}(S_{\mathbb{T}}, I \cdot X)$ which is generated by the image of maps (3) and $\mathbb{T}$-model structure maps on $\mathit{Sub}(S_{\mathbb{T}}, I \cdot X)$ (2).

(ii) Then it suffices to show that $M = \mathit{Sub}(S_{\mathbb{T}}, I \cdot X)$, i.e., each map $f \in \mathit{Sub}(S_{\mathbb{T}}, I \cdot X)$ can be expressed by split_σ's and maps g whose image $g(S_{\mathbb{T}})$ is a subset of $\iota_i(X)$ for some i.

(iii) We show (ii) by induction on the number of indices $i \in I$ such that $\iota_i(X) \cap f(S_{\mathbb{T}}) \neq \emptyset$ (we define I_f as the set of such indexes).

(iv) When $|I_f| = 1$ (in particular $|I| = 1$), there is nothing to do.

(v) When $|I_f| > 1$, we will show that f can be expressed by split_σ and maps g which satisfy $I_g \subsetneq I_f$.

(vi) If I_f is finite, the above argument in fact shows $f \in M$.

(vii) To complete the proof, we will show that even if I_f is infinite, the situation comes to the finite case.

(We ignore the case of $|I_f| = 0$ since it is equivalent to $|I| = 0$.)

The key points are how to express f by using split_σ as (v) and how to show (vii). Firstly, argue about (v). When the theory $\mathbb{T}$ is $\mathbb{T}_A$ (the case Garner dealt with), this has only one symbol $\mathbf{read}_A$ and the final comodel $S_{\mathbb{T}}$ is $A^{\mathbb{N}}$. For a given map f, the map $\mathit{split}(\lambda a \in A.f(a\_))$ is equal to f: for a given sequence $\overrightarrow{a}$, the operation split separates it into the head a_0 and the tail $a_1 a_2 \cdots$, then $\lambda a.f(a\_)$ simply reconnects them (and apply f).

When the final comodel consists of trees (when $\mathbb{T}$ has more than two symbols), the situation is not so simple. For a given f, we want an expression $f = \mathit{split}_\sigma(\lambda i \in |\sigma|.f_i)$. There are many problems. First, we have to select σ. Second, we should define appropriate f_i's. Finally, for a given tree s, split_σ takes only $o_\sigma(s)$ and $\partial_\sigma(s)$ and forgets other information. Thus, no matter how we select σ and f_i's, we cannot fully recover original tree s. So, we focus on recovering the image $f(s)$.

Our solution is taking an index $i \in I$ such that $\iota_i(X) \cap f(S_{\mathbb{T}}) \neq \emptyset$ and judging whether $f(s) \in \iota_i(X)$ or not. Now $\iota_i(X)$ is in the sub-basis of the copower $I \cdot X$. Therefore, $f^{-1}(\iota_i(X))$ has the form $[t^{(i)} \mapsto v^{(i)}]$ for some term $t^{(i)}$ and some valuable $v^{(i)}$. This says that we can judge whether $s \in f^{-1}(\iota_i(X))$ or not by observing *the path of* $t^{(i)}$ *along* s.

Definition 5.5 For a free theory $\mathbb{T}$ and a term $t \in T(V)$, the set of *paths* of the term t, $\mathit{Path}(t)$, is defined as follows. Each path is a sequence of pairs (σ, i) of a symbol σ in $\Sigma_{\mathbb{T}}$ and $i \in |\sigma|$. If t is a variable v, then the only path of v is the empty sequence. A sequence $(\sigma_1, i_1) \cdots (\sigma_n, i_n)$ is in $\mathit{Path}(t)$ if t is of the form $\sigma_1(\lambda i.t_i)$ and $(\sigma_2, i_2) \cdots (\sigma_n, i_n) \in \mathit{Path}(t_{i_1})$.

Definition 5.6 Let $t \in T(V)$ and s be a state in $S_{\mathbb{T}}$. Define inductively *the path of* t *along* s, $\mathit{path}(t, s) \in \mathit{Path}(t)$:

$$\begin{aligned}
\mathit{path}(v, s) &:= \epsilon, \\
\mathit{path}(\sigma(\lambda i.t_i), s) &:= (\sigma, o_\sigma(s)) \cdot \mathit{path}(t_{o_\sigma(s)}, \partial_\sigma(s)).
\end{aligned}$$

Obviously, if $\mathit{path}(t^{(i)}, s) = \mathit{path}(t^{(i)}, s')$ then $f(s) \in \iota_i(X)$ iff $f(s') \in \iota_i(X)$. Thus, we select σ and f_i's as remembering the information about $\mathit{path}(t^{(i)}, s)$ for given s.

Definition 5.7 Let $f \in \mathit{Sub}(S_{\mathbb{T}}, X)$, σ be a symbol and $i \in |\sigma|$. Define $f^{\sigma,i} \in \mathit{Sub}(S_{\mathbb{T}}, X)$ as, for a given $s \in S_{\mathbb{T}}$, constructing a new tree s' such that

$$o_\sigma(s') = i, \ \partial_\sigma(s') = s$$

and applying f to s'. Concretely, $f^{\sigma,i}(s) := f({}^{\sigma,i}s)$, where ${}^{\sigma,i}s$ is the state such that

$$o_\sigma({}^{\sigma,i}s) = i, \ \partial_\sigma({}^{\sigma,i}s) = s$$

and for other symbols τ,

$$o_\tau({}^{\sigma,i}s) = o_\tau(s), \ \partial_\tau({}^{\sigma,i}s) = \partial_\tau(s).$$

For a sequence $p = (\sigma_1, i_1) \cdots (\sigma_n, i_n)$, f^p denotes the map $(f^{\sigma_1,i_1})^{\cdots\sigma_n,i_n}$ and ${}^p s$ denotes the state ${}^{\sigma_1,i_1}({}^{\cdots\sigma_n,i_n} s)$ (if $p = \epsilon$ then $f^\epsilon := f$ and ${}^\epsilon s := s$). Then $f^p(s) = f({}^p s)$.

Remark 5.8 For $f \in \mathit{Sub}(S_{\mathbb{T}}, X)$, the function $f^{\sigma,i}$ is indeed continuous on sub-basis because for a sub-basis U of X, $s \in (f^{\sigma,i})^{-1}(U) \Leftrightarrow s^{\sigma,i} \in f^{-1}(U)$ and $s^{\sigma,i}$ behaves much like s. The nontrivial case is when $f^{-1}(U) = [\sigma(\lambda i.t_i) \mapsto v]$, but we can easily see that $[\![\sigma(\lambda i.t_i)]\!](s^{\sigma,i}) = [\![t_i]\!](s)$. Thus, in this case, $s \in (f^{\sigma,i})^{-1}(U) \Leftrightarrow s \in [t_i \mapsto v]$.

Definition 5.9 For $f \in \mathit{Sub}(S_{\mathbb{T}}, X)$ and for a term t, we define $[f, t] \in \mathit{Sub}(S_{\mathbb{T}}, X)$ inductively:

$$\begin{aligned}[f, v] &:= f \\ [f, \sigma(\lambda i.t_i)] &:= \mathit{split}_\sigma(\lambda i.[f^{\sigma,i}, t_i])\end{aligned}$$

Example 5.10 When $t = \sigma_1(\lambda k.\sigma_2(\lambda l.(k, l)))$, for a given state s such that $o_{\sigma_1}(s) = k_0$ and $o_{\sigma_2}(\partial_{\sigma_1}(s)) = l_0$,

$$\begin{aligned}[f, t](s) &= [f^{(\sigma_1,k_0)}, \sigma_2(\lambda l.(k_0, l))](\partial_{\sigma_1}(s)) \\ &= [f^{(\sigma_1,k_0)(\sigma_2,l_0)}, (k_0, l_0)](\partial_{\sigma_2}(\partial_{\sigma_1}(s))) \\ &= f^{(\sigma_1,k_0)(\sigma_2,l_0)}(\partial_{\sigma_2}(\partial_{\sigma_1}(s))) \\ &= f^{(\sigma_1,k_0)}({}^{(\sigma_2,l_0)}\partial_{\sigma_2}(\partial_{\sigma_1}(s))) \\ &= f({}^{(\sigma_1,k_0)(\sigma_2,l_0)}\partial_{\sigma_2}(\partial_{\sigma_1}(s))).\end{aligned}$$

The state $s' := {}^{(\sigma_1,k_0)(\sigma_2,l_0)}\partial_{\sigma_2}(\partial_{\sigma_1}(s))$ has the same behavior as s on t i.e. $o_{\sigma_1}(s') = o_{\sigma_1}(s) = k_0$ and $o_{\sigma_2}(\partial_{\sigma_1}(s')) = o_{\sigma_2}(\partial_{\sigma_1}(s)) = l_0$. Additionally, s' behaves completely in the same way as s after t, i.e., $\partial_{\sigma_2}(\partial_{\sigma_1}(s')) = \partial_{\sigma_2}(\partial_{\sigma_1}(s))$.

When we write the state constructed by $[f, t^{(i)}]$ from a state s as $\langle t^{(i)}, s\rangle$, we can show that $[f, t^{(i)}](s) = f(\langle t^{(i)}, s\rangle)$ and $f(\langle t^{(i)}, s\rangle) = f(s)$. The former is shown by easy induction on $t^{(i)}$ (we have to describe $\langle t^{(i)}, s\rangle$ concretely). The latter is established as follows (the formal proof is too long to describe here).

- Write $f(s)$,$f(\langle t^{(i)}, s\rangle) \in I \times X$ as (j_0, x_0),(j_1, x_1).
- Since f is continuous on sub-basis, the index j_0 and behaviors of x_0 as a state of $\mathbb{T}'$-comodel $\boldsymbol{X}$ are determined by behaviors of s. Similarly, j_1 and behaviors of x_1 are determined by $\langle t^{(i)}, s\rangle$. (That is, if $f^{-1}(\iota_{i_0}(X)) = [t \mapsto v]$ then $j_0 = i_0 \Leftrightarrow s \in [t \mapsto v]$. Behaviors of x_0 is examined by observing whether $x \in [t' \mapsto v']$ for some t', v' and its inverse image $f^{-1}([t' \mapsto v'])$ is also of the form $[t'' \mapsto v'']$.)
- With effort, we can show that required behaviors of s and $\langle t^{(i)}, s\rangle$ are those on $t^{(i)}$ (i.e., we should ask values of $[\![t^{(i)}]\!](s)$ and $[\![t^{(i)}]\!](\langle t^{(i)}, s\rangle)$) or those after $t^{(i)}$ (i.e., we should observe $[\![t]\!](s)$ and $[\![t]\!](\langle t^{(i)}, s\rangle)$ for some t compatible with $t^{(i)}$).
- Behaviors of $\langle t^{(i)}, s\rangle$ are completely the same as those of s on $t^{(i)}$ and after $t^{(i)}$, as described in Example 5.10.
- Therefore, $j_0 = j_1$ and behaviors of x_0 and x_1 are completely the same.
- Since $\boldsymbol{X}$ is a simple comodel, the behavior determines the state (if $\boldsymbol{X}$ is not simple, there may be different states with the same behavior).

Now we get the following lemma:

Lemma 5.11 *Let $f \in \mathit{Sub}(S_{\mathbb{T}}, I \cdot X)$ such that $I_f := \{i \in I \mid \iota_i(X) \cap f(S_{\mathbb{T}}) \neq \emptyset\}$ has at least two elements. Take an index $i \in I_f$, an appropriate $\mathbb{T}$-term $t^{(i)}$ and a variable $v^{(i)}$ as $f^{-1}(\iota_i(X)) = [t^{(i)} \mapsto v^{(i)}]$. Then $f = [f, t^{(i)}]$.*

Moreover, by definition, $[f, t^{(i)}]$ consists of split_σ's and f^p's ($p \in \mathit{Path}(t^{(i)})$).

Lemma 5.12 *In the situation of Lemma 5.11, maps f^p's appearing in $[f, t^{(i)}]$ satisfy $I_{f^p} \subsetneq I_f$ (where $I_f := \{i \in I \mid \iota_i(X) \cap f(S_{\mathbb{T}}) \neq \emptyset\}$).*

Intuition is that when calculation reaches f^p, it has already identified whether a given state belongs to $f^{-1}(\iota_i(X))$ or not. Thus $I_{f^p} = \{i\}$ or $i \notin I_{f^p}$.

So far, we completed the part (v) of the overview. If I_f is finite, by induction, we can express f by using operations split_σ and maps g such that $g : S_{\mathbb{T}} \to \iota_{i_g}(X)$ for an index $i_g \in I$, and conclude $f \in M$ (complete (vi) of the overview).

It remains to consider the case that I_f is infinite. First, specify the induction in the finite case (we assume $I \neq \emptyset$ for simplicity):

(1) If $|I_f| > 1$ then we choose an index $l_0 \in I_f$.

(2) Take a term t_0 and a variable v_0 as $f^{-1}(\iota_{l_0}(X)) = [t_0 \mapsto v_0]$.

(3) Write f as $[f, t_0]$.

(4) Consider f^p's, $p \in \mathit{Path}(t_0)$ (maps appearing in $[f, t_0]$).

(5) By Lemma 5.12, for all $p \in \mathit{Path}(t_0)$, we have $I_{f^p} \subsetneq I_f$ and they are not empty.

(6) If $|I_{f^p}| = 1$ for all p then the induction is finished.

(7) If there are paths p with $|I_{f^p}| > 1$, we apply this procedure for such maps f^p instead of f (take $l_1^p \in I_{f^p}$, let $(f^p)^{-1}(\iota_{l_1^p}(X)) = [t_1^p \mapsto v_1^p]$, etc.).

We call this procedure the *splitting procedure* of f. Even when I_f is infinite, if we reach the situation such that all maps f^p appearing in the splitting procedure satisfy the condition that I_{f^p} is finite, then we can conclude $f \in M$. Thus our aim is to show that we can always reach this situation, by contradiction.

Suppose that we cannot reach this situation. Then the splitting procedure continues infinitely and we get a infinite sequence $(\sigma_1, i_n)(\sigma_2, i_2)\cdots$ such that for all $n \geq 1$, the set $I_{f^{(\sigma_1,i_1)\cdots(\sigma_n,i_n)}}$ is infinite. Since this sequence is given by the splitting procedure of f, there exists the unique natural number n_1 such that the sequence $p_1 := (\sigma_1, i_1)\cdots(\sigma_{n_1}, i_{n_1}) \in \mathit{Path}(t_0)$ and we apply (7) to the map f^{p_1}; take appropriate $l_1 \in I$ (especially $l_1 \neq l_0$), t_1, v_1 as $f_0^{-1}(\iota_{l_1}(X)) = [t_1 \mapsto v_1]$ then we get the sequence $p_2 := (\sigma_{n_1+1}, i_{n_1+1})\cdots(\sigma_{n_2}, i_{n_2}) \in \mathit{Path}(t_1)$ for the unique n_2. Therefore we are in the following situation:

$\exists$infinite sequence of natural numbers $0 = n_0 < n_1 < n_2 < \cdots$,
$\forall j \geq 0$, $\exists l_j \in I$ (different from each other), $\exists$term t_j, $\exists$variable v_j,
$p_{j+1} := (\sigma_{n_j+1}, i_{n_j+1})\cdots(\sigma_{n_{j+1}}, i_{n_{j+1}}) \in \mathit{Path}(t_j)$,
$(f^{q_j})^{-1}(\iota_{l_j}(X)) = [t_j \mapsto v_j] \neq \emptyset, S_{\mathbb{T}}$,
$f^{q_j} = [f^{q_j}, t_j]$,
$I_{f^{q_j}}$ is infinite.
$(q_j := (\sigma_1, i_1)\cdots(\sigma_{n_j}, i_{n_j}) = p_1 \cdots p_j,\ q_0 = p_0 := \epsilon.)$

If a state s satisfies $\mathit{path}(t_j, s) = p_{j+1}$ then $[\![t_j]\!](s) \neq v_j$ i.e. $f^{q_j}(s) \notin \iota_{l_j}(X)$: because if $[\![t_j]\!](s) = v_j$ for this s, then for all state s', $[\![t_j]\!]({}^{p_{j+1}}s') = v_j$ and this implies $f^{q_{j+1}}(s') = f^{q_j \cdot p_{j+1}}(s) = f^{q_j}({}^{p_{j+1}}s') \in \iota_{l_j}(X)$, contradicts to the assumption that $I_{f^{q_{j+1}}}$ is infinite.

By extending this argument, for a state $s' := {}^{q_{j+1}}s$ (s is an arbitrary state), we can show that $f(s')$ does not belong to any of $\iota_{l_0}(X), \ldots \iota_{l_j}(X)$. For example, when $j = 1$, then $s' = {}^{p_1 \cdot p_2}s$ and it is clear that $f(s') \notin \iota_{l_0}(X)$ by $\mathit{path}(t_0, s') = p_1$; for $\iota_{l_1}(X)$, we can calculate that $f(s') = [f, t_0](s') = f^{p_1}({}^{p_2}s) = f^{q_1}({}^{p_2}s)$, thus $f(s') \in \iota_{l_1}(X)$ iff $f^{q_1}({}^{p_2}s) \in \iota_{l_1}(X)$ but the latter is false.

Then if we can construct a state s' such as ${}^{p_1 \cdot p_2 \cdots}s$ (this notation is informal), we expect that $f(s')$ does not belong to $\iota_{l_j}(X)$ for all j. Assume we have such a state s', and let $f(s') \in \iota_l(X)$ for $l \in I$, which satisfies $l \neq l_j$ for all j. By the construction of the sequence $(\sigma_1, i_1)(\sigma_2, i_2)\cdots$, we may have to observe an infinite behavior of a given state s'' to decide whether $f(s'') \in \iota_l(X)$. This contradicts to the continuity of f on sub-basis.

For lack of space, we only explain the construction of such a state and the outline of derivation of a contradiction.

Definition 5.13 For any state $s \in S_{\mathbb{T}}$ and any infinite sequence $\overrightarrow{(\tau, i)} = (\tau_1, i_1)(\tau_2, i_2)\cdots$, where τ_k is a operation symbol in $\mathbb{T}$ and $i_k \in |\tau_k|$ for all k, we define a new state ${}^{\overrightarrow{(\tau,i)}}s$ as the function $\Sigma_{\mathbb{T}}^* \to \prod_{\sigma\in\Sigma_{\mathbb{T}}} |\sigma|$:

$$
{}^{\overrightarrow{(\tau,i)}}s(\overrightarrow{x}) := \begin{cases} s(\epsilon)[\tau_k \to i_k] & (\overrightarrow{x} = \tau_1 \cdots \tau_{k-1}) \\ s(\overrightarrow{y}) & (\overrightarrow{x} = \tau_1 \cdots \tau_{k-1} \overrightarrow{y},\ y_1 \neq \tau_k)\ (k \geq 1) \end{cases}
$$

We define the infinite sequence $\overrightarrow{(\tau,i)_k}$ for a natural number $k \geq 1$ and a infinite sequence $\overrightarrow{(\tau,i)} = (\tau_n, i_n)_{n\geq 1}$ as:

$$
\overrightarrow{(\tau,i)_k} := (\tau_n, i_n)_{n\geq k}.
$$

Lemma 5.14 *For any state $s \in S_{\mathbb{T}}$ and any infinite sequence $\overrightarrow{(\tau,i)}$,*

$$
o_{\tau_1}({}^{\overrightarrow{(\tau,i)}}s) = i_1,\ \partial_{\tau_1}({}^{\overrightarrow{(\tau,i)}}s) = {}^{\overrightarrow{(\tau,i)_2}}s,
$$

and inductively, for each $k \geq 1$,

$$
\begin{aligned} o_{\tau_k}(\partial_{\tau_{k-1}} \cdots \partial_{\tau_1}({}^{\overrightarrow{(\tau,i)}}s)) &= i_k \\ \partial_{\tau_k} \cdots \partial_{\tau_1}({}^{\overrightarrow{(\tau,i)}}s) &= {}^{\overrightarrow{(\tau,i)_{k+1}}}\ s. \end{aligned}
$$

Consider the infinite sequence $\overrightarrow{(\sigma,i)}$ constructed by the splitting procedure of f_0 and fix a state s. When we abbreviate the composition $\partial_{\sigma_{n_j}} \cdots \partial_{\sigma_{n_1}}$ as ∂_j (∂_0 is the identity on $S_{\mathbb{T}}$), then by the above lemma, for all j,

$$
\mathit{path}(t_j, \partial_j({}^{\overrightarrow{(\sigma,i)}}s)) = (\sigma_{n_j+1}, i_{n_j+1}) \cdots (\sigma_{n_{j+1}}, i_{n_{j+1}}) = p_{j+1}. \tag{4}
$$

This implies $f({}^{\overrightarrow{(\sigma,i)}}s) \notin \iota_{l_j}(X)$ for all j: we can show by easy induction on j that $f({}^{\overrightarrow{(\sigma,i)}}s) = f^{q_j}(\partial_j({}^{\overrightarrow{(\sigma,i)}}s))$ for each j, and then by (4), $f^{q_j}(\partial_j({}^{\overrightarrow{(\sigma,i)}}s)) \notin \iota_{l_j}(X)$.

Let $l \in I$ be the index such that $f({}^{\overrightarrow{(\sigma,i)}}s) \in \iota_l(X)$, then $l \neq l_j$ for all j. By the continuity of f, $f^{-1}(\iota_l(X)) = [t \mapsto v]$ for some t and v. We show, by induction on $j \geq 1$, that each q_j is a prefix of the path $\mathit{path}(t, {}^{\overrightarrow{(\sigma,i)}}s)$ and therefore the length of $\mathit{path}(t, {}^{\overrightarrow{(\sigma,i)}}s)$ is infinite, this is irrational.

We have reached the contradiction. Thus, even if I_f is infinite, the splitting procedure of f comes down to the finite case and therefore $f \in M$. This implies $M = \mathit{Sub}(S_{\mathbb{T}}, X)$, we complete the proof of Theorem 5.4.

To complete the proof of that the functor $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_)$ preserves copowers of simple comodels (the part (II) of the overview), we show the existence of the mediating morphism. In the proof below, we use the fact that $M = \mathit{Sub}(S_{\mathbb{T}}, X)$ we have shown above and thus we assume $X \in \mathit{Sub}$ has a simple $\mathbb{T}'$-comodel structure.

Proposition 5.15 (the existence of mediating morphism) *Assume that $X \in \mathit{Sub}$ has a simple $\mathbb{T}'$-comodel structure. If there is a family of $\mathbb{T}$-model map*

$$
(p_i : \mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, X) \to \boldsymbol{Y})_{i\in I}
$$

then there is a $\mathbb{T}$-model map

$$
\overline{p} : \mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, I \cdot X) \to \boldsymbol{Y}
$$

such that $p_i = \overline{p} \circ (\iota_i \circ (\_))$ for all $i \in I$.

Proof. Let $\boldsymbol{N}$ be the free $\mathbb{T}$-model on the set $\mathit{Sub}(S_{\mathbb{T}}, X) \times I$. By freeness, we have the unique $\mathbb{T}$-model map $\beta : \boldsymbol{N} \to \mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, X)$ with $\beta(f, i) = \iota_i f$ and the unique map $p' : \boldsymbol{N} \to \boldsymbol{Y}$ with $p'(f, i) = p_i(f)$. It suffices to show that there is a factorization p of p' through β i.e. $p' = p \circ \beta$. By the proof of the above theorem, β is epimorphic. So, it suffices to show that if $x, y \in N$ satisfy $\beta(x) = \beta(y)$, then they satisfy $p'(x) = p'(y)$. We can do this by induction on the total number of operation symbols in $\mathbb{T}$-terms x, y. □

Now we have completed all of the steps described in the overview and reached the goal:

Theorem 5.16 *For free theories $\mathbb{T}$, $\mathbb{T}'$, the set $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ appears as the final $\mathbb{T}$-$\mathbb{T}'$-bimodel with $\mathbb{T}$-model structure maps $(\mathit{split}_{\sigma})_{\sigma\in\Sigma_{\mathbb{T}}}$ and $\mathbb{T}'$-comodel structure maps*

$$\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, S_{\mathbb{T}'}) \xrightarrow{[\![\tau]\!]^{\boldsymbol{S}_{\mathbb{T}'}} \circ (\_)} \mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, |\tau| \cdot S_{\mathbb{T}'}) \xrightarrow{\cong} |\tau| \cdot \mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, S_{\mathbb{T}'})$$

for $\tau \in \Sigma_{\mathbb{T}'}$, where the first part is postcomposition with the $\mathbb{T}'$-comodel structure map $[\![\tau]\!]^{\boldsymbol{S}_{\mathbb{T}'}}$ of the final $\mathbb{T}'$-comodel $\boldsymbol{S}_{\mathbb{T}'}$ and the second part is the canonical isomorphism coming from the fact that $\mathit{Sub}(\boldsymbol{S}_{\mathbb{T}}, \_) : \mathit{Sub} \to \mathit{Mod}(\mathbb{T})$ preserves copowers.

Finally, we describe the relation between the final residual comodel $I_{\mathbb{T},\mathbb{T}'}$ and the final bimodel $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$. In section 4, we constructed $\mathsf{reflect} : I_{\mathbb{T},\mathbb{T}'} \to \mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$. We can also define a map of converse direction $\mathsf{reify} : \mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'}) \to I_{\mathbb{T},\mathbb{T}'}$. This is because each $\mathbb{T}$-$\mathbb{T}'$-bimodel equips a $\mathbb{T}$-residual $\mathbb{T}'$-comodel structure. We can show that the composition $\mathsf{reflect} \circ \mathsf{reify}$ is the identity on $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$. This establishes the complete representation of $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ by $I_{\mathbb{T},\mathbb{T}'}$. The argument is the same as [5].

6 Conclusion and Future Work

Our main contribution is giving a comodel-theoretic characterization of $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$, which is a subset of $\mathit{Top}(S_{\mathbb{T}}, S_{\mathbb{T}'})$. Explicitly, $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ can appear as the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. Additionally, we constructed maps

$$\begin{tikzcd} \mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'}) \arrow[r, shift right, "\mathsf{reify}"'] & I_{\mathbb{T},\mathbb{T}'}, \arrow[l, shift right, "\mathsf{reflect}"'] \end{tikzcd}$$

where $I_{\mathbb{T},\mathbb{T}'}$ is the underlying set of the final $\mathbb{T}$-residual $\mathbb{T}'$-comodel and we identify $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$ with the final $\mathbb{T}$-$\mathbb{T}'$-bimodel. Then we showed that the composition $\mathsf{reflect} \circ \mathsf{reify}$ is the identity on $\mathit{Sub}(S_{\mathbb{T}}, S_{\mathbb{T}'})$, this implies the completeness of $\mathsf{reflect}$. A further generalization is required in order to give a complete representation of $\mathit{Top}(S_{\mathbb{T}}, S_{\mathbb{T}'})$. In [6], Ghani et al. gave a coalgebraic representation of continuous functions on final coalgebras of various functors but they did not show its completeness. On the other hand, although we show a kind of complete correspondence, this does not consider the whole set of continuous functions.

Our second contribution is an analysis of the final comodel of a free algebraic theory and functions continuous on sub-basis from it. During the proof in Section 5, we defined several notions such as the continuous function $f^{\sigma,i}$ and the state ${}^{\sigma,i}s$. We expect that they and the ideas underlying them are useful when studying arbitrary continuous functions or investigating the case of non-free algebraic theories.

According to [9], transducers with backtracking characterize continuous functions between the set of trees. After submitting this article, we established a retraction between appropriate transducers and the residual comodels. We hope to report this result elsewhere.

When we try to generalize our argument to the case of non-free algebraic theories, there are many difficulties. One is the question whether bimodels can be seen as residual comodels (the latter of Lemma 4.7). This is a key point to define the map reify. Of course, there can be unnoticed issues. We should carefully analyse our proofs and we would like to identify algebraic theories in which our argument is effective.

Acknowledgements

I am grateful to Soichiro Fujii for suggesting the topic treated in this paper. He and my supervisor Masahito Hasegawa helped me in many ways, and discussion with them were very meaningful. Finally, I would like to thank my advisors and colleagues for broadening and deepening my knowledge.

References

[1] Béal, M.-P. and O. Carton, *Determinization of transducers over finite and infinite words*, Theoretical Computer Science **289** (2002), pp. 225–251.

URL https://www.sciencedirect.com/science/article/pii/S0304397501002717

[2] Capretta, V., G. Hutton and M. Jaskelioff, *Contractive functions on infinite data structures*, in: *Proceedings of the 28th Symposium on the Implementation and Application of Functional Programming Languages*, IFL 2016 (2016).
URL https://doi.org/10.1145/3064899.3064900

[3] Freyd, P., *Algebra valued functors in general and tensor products in particular*, Colloquium Mathematicae **14** (1966), pp. 89–106.
URL https://eudml.org/doc/262988#reverseReferences

[4] Garner, R., *The costructure-cosemantics adjunction for comodels for computational effects*, Mathematical Structures in Computer Science (2021), pp. 1–46.
URL https://doi.org/10.1017/S0960129521000219

[5] Garner, R., *Stream processors and comodels*, in: F. Gadducci and A. Silva, editors, *9th Conference on Algebra and Coalgebra in Computer Science, CALCO 2021, August 31 to September 3, 2021, Salzburg, Austria*, LIPIcs **211** (2021), pp. 15:1–15:17.
URL https://doi.org/10.4230/LIPIcs.CALCO.2021.15

[6] Ghani, N., P. Hancock and D. Pattinson, *Continuous functions on final coalgebras*, Electr. Notes Theor. Comput. Sci. **249** (2009), pp. 3–18.

[7] Ghani, N., P. G. Hancock and D. Pattinson, *Representations of stream processors using nested fixed points*, Logical Methods in Computer Science **5** (2009).
URL http://arxiv.org/abs/0905.4813

[8] Ginsburg, S. and S. Greibach, *Abstract families of languages*, in: *Proceedings of the 8th Annual Symposium on Switching and Automata Theory (SWAT 1967)*, FOCS ’67 (1967), pp. 128–139.
URL https://doi.org/10.1109/FOCS.1967.3

[9] Hyvernat, P., *Representing continuous functions between greatest fixed points of indexed containers*, Logical Methods in Computer Science **Volume 17, Issue 3** (2021).
URL https://lmcs.episciences.org/7712

[10] Jacobs, B., “Introduction to Coalgebra: Towards Mathematics of States and Observation,” Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, 2016.

[11] Jacobs, B. and J. J. M. M. Rutten, *A tutorial on (co)algebras and (co)induction*, Bulletin of The European Association for Theoretical Computer Science (1997).

[12] Katsumata, S.-y., E. Rivas and T. Uustalu, *Interaction laws of monads and comonads*, CoRR **abs/1912.13477** (2019).
URL http://arxiv.org/abs/1912.13477

[13] Moggi, E., *Notions of computation and monads*, Information and Computation **93** (1991), pp. 55–92.
URL https://www.sciencedirect.com/science/article/pii/0890540191900524

[14] Plotkin, G. and J. Power, *Notions of computation determine monads*, Lecture Notes in Computer Science **2303**, 2001.

[15] Plotkin, G. and J. Power, *Tensors of comodels and models for operational semantics*, Electronic Notes in Theoretical Computer Science **218** (2008), pp. 295–311.
URL https://doi.org/10.1016/j.entcs.2008.10.018

[16] Power, J. and O. Shkaravska, *From Comodels to Coalgebras: State and Arrays*, Electronic Notes in Theoretical Computer Science **106** (2004), pp. 297–314.
URL https://www.sciencedirect.com/science/article/pii/S1571066104051813

[17] Rutten, J. J. M. M., *Universal coalgebra: A theory of systems*, Theoretical Computer Science **249** (2000), pp. 3–80.
URL https://www.sciencedirect.com/science/article/pii/S0304397500000566

[18] Uustalu, T., *Stateful Runners of Effectful Computations*, Electronic Notes in Theoretical Computer Science **319** (2015), pp. 403–421.