



**HAL**  
open science

# A unified treatment of structural definitions on syntax for capture-avoiding substitution, context application, named substitution, partial differentiation, and so on

Tom Hirschowitz, Ambroise Lafont

## ► To cite this version:

Tom Hirschowitz, Ambroise Lafont. A unified treatment of structural definitions on syntax for capture-avoiding substitution, context application, named substitution, partial differentiation, and so on. 2022. hal-03633933

**HAL Id: hal-03633933**

**<https://hal.science/hal-03633933>**

Preprint submitted on 7 Apr 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A unified treatment of structural definitions on syntax for capture-avoiding substitution, context application, named substitution, partial differentiation, and so on

TOM HIRSCHOWITZ, Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000, France  
AMBROISE LAFONT, University of Cambridge, United Kingdom

We introduce a category-theoretic abstraction of a syntax with auxiliary functions, called an admissible monad morphism. Relying on an abstract form of structural recursion, we then design generic tools to construct admissible monad morphisms from basic data. These tools automate ubiquitous standard patterns like (1) defining auxiliary functions in successive, potentially dependent layers, and (2) proving properties of auxiliary functions by induction on syntax. We cover significant examples from the literature, including the standard lambda-calculus with capture-avoiding substitution, a lambda-calculus with binding evaluation contexts, the lambda-mu-calculus with named substitution, and the differential lambda-calculus.

Additional Key Words and Phrases: syntax ; variable binding ; substitution ; category theory

## 1 INTRODUCTION

*Motivation.* The literature offers several initial-algebra semantics frameworks for generating and reasoning about syntax with variable binding (e.g., [Fiore and Hur 2009; Fiore et al. 1999; Gabbay and Pitts 1999; Hofmann 1999]). Still, even state-of-the-art frameworks lack some expressiveness to suit the working operational semanticist’s needs. One typical limitation, which is the topic of active research [Coraglia and Di Liberti 2021; Gratzner and Sterling 2021], concerns some complex typing features like dependent types.

In this paper, we are concerned with a different limitation: although existing frameworks do explain capture-avoiding substitution satisfactorily, they largely ignore the numerous similar auxiliary operations on syntax, like context application (e.g., in ML [Pierce 2004, Chapter 10]), named substitution in the  $\lambda\mu$ -calculus [Parigot 1992] (a.k.a. named application [Vaux 2007]), partial differentiation in the differential  $\lambda$ -calculus [Ehrhard and Regnier 2003], and so on.

One way of dealing with such auxiliary operations is to build them into the syntax, and view their defining equations as an equational theory, by which to quotient the initial model (see, e.g., [Fiore and Hur 2009; Gratzner and Sterling 2021]). The problem with this approach is that it lacks the basic, inductive construction of the initial model. Or in other words, it misses the fact that auxiliary operations are... auxiliary, i.e., that they are admissible (= encodable) in the initial model.

*Contributions.* In this paper, we propose a categorical foundation for such admissible operations, called **admissible monad morphisms**, together with a general framework for constructing them and reasoning about them. In particular, the framework offers tools to

- (a) define successive layers of auxiliary operations, each layer potentially depending on previous ones (as, e.g., in the differential  $\lambda$ -calculus [Ehrhard and Regnier 2003]),
- (b) automatically derive **benign** equations, i.e., equations involving auxiliary operations that are satisfied by the syntax (such as, e.g., associativity of substitution  $e[\sigma][\theta] = e[\sigma[\theta]]$ ).

**Remark 1.** *The framework does not yet allow the user to automatically prove that auxiliary operations are compatible with **severe** equations, i.e., equations not involving auxiliary operations (as, again, in the differential  $\lambda$ -calculus), whose initial model thus may be a proper quotient of the syntax.*

---

Authors’ addresses: Tom Hirschowitz, Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000, Chambéry, France; Ambroise Lafont, University of Cambridge, Cambridge, United Kingdom.

We illustrate the expressiveness of the framework by reconstructing (1) Fiore et al.'s [1999] approach as a special case, and (2) a few concrete examples from the literature, notably a  $\lambda$ -calculus with explicit substitution [Accattoli 2019] (which involves variable capture by evaluation contexts) and the differential  $\lambda$ -calculus [Ehrhard and Regnier 2003] (without quotienting by structural equations). We also demonstrate that our framework is not tied to any particular setting for modelling variable binding, by reconstructing pure  $\lambda$ -calculus with capture-avoiding substitution in De Bruijn representation (§4.6).

The framework consists of the following components.

1. First, we offer a fundamental, very easy construction of admissible monad morphisms from suitable monad distributive laws in the sense of Beck [1969]. A monad distributive law is a natural transformation  $TS \rightarrow ST$ , where we think of  $S$  as the monad of basic operations, hence call it the **basic** monad, and of  $T$  as the one of auxiliary operations, hence call it the **auxiliary** monad. We show that, whenever  $T$  preserves the initial object, any monad distributive law induces an admissible monad morphism  $S \rightarrow ST$  to the composite monad.
2. Then, we design a toolbox for constructing such distributive laws from more basic data:
  - (i) We first introduce a simple notion of signature for distributive laws. A signature consists of an endofunctor  $\Sigma$ , equipped with an abstract analogue of a structurally recursive definition, called a **simple structural law**. From such a signature, we generate a distributive law of the desired form, whose basic monad is the free monad  $\Sigma^*$  on  $\Sigma$ . We also establish a simple characterisation of **augmented algebras**, i.e.,  $\Sigma$ -algebras featuring the specified auxiliary operations in a compatible way.
 

Simple structural laws are expressive enough to equip (potentially binding) syntax with capture-avoiding substitution, but not to tackle Features (a) and (b) above.
  - (ii) We then introduce a second notion of signature, which is an incremental form of the first. A signature consists of a distributive law  $\delta: TS \rightarrow ST$  whose basic monad  $S$  is free on some endofunctor  $\Sigma$ , equipped with a so-called **incremental structural law** over it. The idea is that we have already specified a few auxiliary operations over  $S$ , bundled as the monad  $T$ , and an incremental structural law is like a structurally recursive definition by pattern-matching on basic operations, whose body may use all auxiliary operations. From this, we construct a new distributive law with the same basic monad, but whose auxiliary monad contains new operations. We again present a simple explicit description of augmented algebras.
 

Incremental structural laws are expressive enough to equip syntax with incremental layers of auxiliary operations, hence implement Feature (a) above.
  - (iii) We finally address the benign equations feature (b). For this, we again start from a distributive law with a free basic monad, and introduce **structural equational systems**, which, roughly, consist of:
    - (an abstract form of) equations on terms, potentially using auxiliary operations, and
    - data that ensures that the equations will be satisfied by the syntax.

From this, we construct a distributive law with the same basic monad.

*Plan.* We start in §2 by introducing admissible monad morphisms and showing how they may be derived from suitable monad distributive laws. We then study simple structural laws in §3, incremental structural laws in §4, and benign equations in §5, giving applications along the way. Finally, we conclude and give some perspectives in §6.

*Related work.* Our framework abstracts over the idea of a definition by structural recursion, to categories other than sets. The abstraction emphasises the idea that the syntax equipped with auxiliary functions is initial in a category of augmented algebras. Our framework is directly inspired

by (and abstracts over) Fiore et al.'s [Fiore et al. 1999; Fiore 2008]. A notable difference is that Fiore et al. rely on a technical notion called **pointed strong** endofunctors, which intuitively amounts to assuming that the argument of the endofunctor already has part of the desired structure – in this case, variables. The corresponding instance of our abstract framework avoids this astute trick by taking the free structure. Technically, this is visible in the presence of  $X \otimes S(Y)$  in the codomain of our simple structural law in §3.5. Indeed, taking  $S(Y)$  instead of merely  $Y$  is what allows us to use  $\sigma^\uparrow$  in the definition.

Our work is related to the idea of traversals [Allais et al. 2018]. The latter are however limited to categories of families of sets indexed by some fixed set of types, and tailored for capture-avoiding substitution.

*Notation and prerequisites.* We often conflate natural numbers with the corresponding finite ordinals, or some choice of equipotent set, hopefully made clear by the context.

We assume basic knowledge of category theory [Mac Lane 1998] and locally finitely presentable categories [Adámek and Rosicky 1994] (although the latter may be ignored on a superficial reading). By default, our categories are locally small, although we occasionally repeat it for emphasis. We let **CAT** denote the category of (locally small) categories.

We generally denote initial objects by  $\emptyset$ , relying on context to infer the corresponding category. For any category  $\mathbf{C}$  and object  $c \in \mathbf{C}$ , we denote by  $\mathbf{C}/c$  the slice category over  $c$ .

On any category  $\mathbf{C}$  with binary coproducts, for any object  $E \in \mathbf{C}$ , we denote the corresponding **option** functor by  $O_E$ , i.e.,  $O_E(C) = C + E$ , for some choice of coproducts.

For any endofunctor  $F$ , we denote its category of algebras by  $F\text{-alg}$ . For any monad  $T$ , we denote its category of monad algebras by  $T\text{-Alg}$ ; it is then a full subcategory of  $T\text{-alg}$ . We denote the coproduct of two monads  $T$  and  $T'$  by  $T \oplus T'$ , to distinguish it from the coproduct  $T + T'$  of underlying endofunctors.

Any finitary endofunctor  $F$  on a locally finitely presentable category  $\mathbf{C}$  generates a free monad, that we denote by  $F^*$ . In particular, the initial algebra is denoted by  $F^*\emptyset$ . As is well-known, this may often be computed as a directed colimit which we denote by  $\mu E.F(E)$ . We furthermore denote the unit  $F \rightarrow F^*$  by  $\eta_F$ . Please also note that  $F^*\text{-Alg}$  is isomorphic to  $F\text{-alg}$ : the isomorphism maps an algebra  $F^*c \rightarrow c$  to  $Fc \rightarrow F^*c \rightarrow c$ .

We write  $\Delta$  for the diagonal functor  $\mathbf{C} \rightarrow \mathbf{C} \times \mathbf{C}$  mapping an object  $c$  of a category  $\mathbf{C}$  to  $(c, c)$ . We use the letters  $\Gamma$  and  $\Theta$  to denote bifunctors  $\mathbf{C} \times \mathbf{C} \rightarrow \mathbf{C}$ . We sometimes write  $\Gamma$  instead of  $\Gamma\Delta$ . E.g., we talk about  $\Gamma$ -algebras instead of  $\Gamma\Delta$ -algebras, and denote  $(\Gamma\Delta)^*$  by  $\Gamma^*$ . We furthermore sometimes denote  $\Gamma(X, Y)$  by  $\Gamma_Y X$ . If  $F$  is a functor to  $\mathbf{C}$ , the bifunctor mapping  $(X, Y)$  to  $\Gamma(X, FY)$  is denoted by  $\Gamma_F$ .

Moreover, for denoting the components of a natural transformation  $\alpha: F \rightarrow G$ , we freely switch between  $\alpha_C$  and  $\alpha C$ , and we denote horizontal composition in **CAT** by mere juxtaposition. E.g., given any  $f: C \rightarrow D$  in the domain category, we also write  $\alpha_f$ , or  $\alpha f$ , for either composite  $\alpha_D \circ F(f) = G(f) \circ \alpha_C$ .

The middle dot  $\cdot$  is overloaded: it may have the following three meanings, depending on context:

- for any presheaf  $F: \mathbf{C}^{op} \rightarrow \mathbf{Set}$ , morphism  $f: c \rightarrow d$  in  $\mathbf{C}$ , and element  $x \in F(d)$ , we define  $x \cdot f := F(f)(x)$ ;
- for any set  $X$  and object  $c$  of any category with enough coproducts, we denote by  $X \cdot c$  the  $X$ -fold coproduct  $\sum_{x \in X} c$ ; and finally,
- the middle dot is used in the syntax of differential  $\lambda$ -calculus.

*Definition 1.1.* A **monad morphism**  $S \rightarrow T$  between monads  $S$  and  $T$  on a given category  $\mathbf{C}$  is a natural transformation  $\alpha: S \rightarrow T$  commuting with multiplication and unit, i.e., making the following diagrams commute.

$$\begin{array}{ccc}
SS & \xrightarrow{\alpha\alpha} & TT \\
\mu^S \downarrow & & \downarrow \mu^T \\
S & \xrightarrow{\alpha} & T
\end{array}
\qquad
\begin{array}{ccc}
& & \text{id}_{\mathbf{C}} \\
& \eta^S \swarrow & \searrow \eta^T \\
S & \xrightarrow{\alpha} & T
\end{array}$$

We let  $\mathbf{Mnd}(\mathbf{C})$  denote the category of monads in  $\mathbf{C}$  and monad morphisms between them, and  $\mathbf{Mnd}_f(\mathbf{C})$  denote its full subcategory spanned by finitary monads.

## 2 ADMISSIBLE MONAD MORPHISMS AND DISTRIBUTIVE LAWS

In this section, we introduce admissible monad morphisms, and show how to construct them from monad distributive laws.

### 2.1 Admissible monad morphisms

*Definition 2.1.* Let  $\mathbf{C}$  have an initial object. A morphism  $\alpha: R \rightarrow S$  of monads on  $\mathbf{C}$  is **admissible** iff its component  $\alpha_0: R0 \rightarrow S0$  at the initial object is an isomorphism.

**Remark 2.** *Intuitively, thinking of  $R$  and  $S$  as algebraic structures, or families of operations,  $R(0)$  and  $S(0)$  are the initial algebras, i.e., morally the syntaxes of both languages. The monad morphism translates each operation from  $R$  to some (derived) operation from  $S$ . Admissibility then amounts to the translation being an isomorphism.*

We will give the paradigmatic example of admissible morphism just below, but before that, let us mention a different point of view on admissible morphisms.

*Definition 2.2* ([Mac Lane 1998, §VI.3]). We call **monadic** (over  $\mathbf{C}$ ) any functor isomorphic in  $\mathbf{CAT}/\mathbf{C}$  to some forgetful functor  $T - \mathbf{Alg} \rightarrow \mathbf{C}$ , and let  $\mathbf{Monadic}/\mathbf{C}$  denote the full subcategory of  $\mathbf{CAT}/\mathbf{C}$  spanned by monadic functors.

Given a category  $\mathbf{C}$ , the assignment mapping any monad  $T$  on  $\mathbf{C}$  to the forgetful functor  $U^T: T - \mathbf{Alg} \rightarrow \mathbf{C}$  extends to a functor  $\mathbf{sem}: \mathbf{Mnd}(\mathbf{C})^{op} \rightarrow \mathbf{Monadic}/\mathbf{C}$ , and we have:

LEMMA 2.3. *The functor  $\mathbf{sem}$  is an equivalence of categories.*

PROOF. The functor  $\mathbf{sem}$  is essentially surjective by definition of monadic functors. It is also full and faithful by [Barr 1970, Proposition 5.3].  $\square$

PROPOSITION 2.4. *Given a category  $\mathbf{C}$  with initial object, monads  $S$  and  $T$  on  $\mathbf{C}$ , and a monad morphism  $\alpha: S \rightarrow T$ , the following are equivalent:*

- (i)  $\alpha$  is admissible;
- (ii)  $\mathbf{sem}(\alpha): T - \mathbf{Alg} \rightarrow S - \mathbf{Alg}$  preserves the initial object;
- (iii)  $\mathbf{sem}(\alpha): T - \mathbf{Alg} \rightarrow S - \mathbf{Alg}$  creates the initial object in the sense of [Mac Lane 1998, §V.1], which in this case means that the initial  $S$ -algebra  $S0$  possesses a unique  $T$ -algebra structure  $a: TS0 \rightarrow S0$  making the triangle

$$\begin{array}{ccc}
SS0 & \xrightarrow{\alpha_{S0}} & TS0 \\
& \searrow \mu_0^S & \swarrow a \\
& & S0
\end{array}$$

commute, which furthermore makes it initial in  $T - \mathbf{Alg}$ .

PROOF. See §A.  $\square$

*Example 2.5.* Our motivating example is in fact a class of examples. For any finitary, **pointed strong** endofunctor  $\Sigma$  on any nice monoidal category  $(\mathbf{C}, \otimes, I, \alpha, \lambda, \rho)$  (the impatient reader may consult Definitions 5.14 and 5.16 below for details, but these are not yet needed), Fiore et al. [1999] introduce a category  $\Sigma$  - **Mon** of  $\Sigma$ -**monoids**, which are objects  $X \in \mathbf{C}$  with both  $\Sigma$ -algebra structure  $\Sigma X \rightarrow X$  and monoid structure  $I \rightarrow X \leftarrow X \otimes X$ , satisfying a standard coherence condition (Definition 5.15 below).

They then show that the initial  $(I + \Sigma)$ -algebra, or equivalently the free  $\Sigma$ -algebra on  $I$ , admits a unique  $\Sigma$ -monoid structure, which makes it initial in  $\Sigma$  - **Mon**. Furthermore, both forgetful functors from  $(I + \Sigma)$  - **alg** and  $\Sigma$  - **Mon** to  $\mathbf{C}$  are monadic, and there is an obvious forgetful functor  $\Sigma$  - **Mon**  $\rightarrow$   $(I + \Sigma)$  - **alg**.

Thus, denoting by  $(I + \Sigma)^*$  and  $\Sigma^\circledast$  the corresponding monads, their result may be read as proving that the induced monad morphism

$$(I + \Sigma)^* \rightarrow \Sigma^\circledast$$

is admissible.

## 2.2 From distributive laws

Let us now show how to construct admissible monad morphisms from distributive laws. We first recall from [Beck 1969] that

- (i) a **monad distributive law** of  $S$  over  $T$  is a natural transformation  $\delta: TS \rightarrow ST$ , commuting with unit and multiplication of both monads  $S$  and  $T$ , and that
- (ii) any such distributive law equips the composite functor  $ST$  with monad structure, which in particular makes the natural transformation  $S\eta^T: S \rightarrow ST$  into a monad morphism.

The idea is to start from a distributive law with a suitable constraint on  $T$ , namely that  $T(\emptyset) \cong \emptyset$ . Intuitively,  $T$  may have many operations, but no constants to feed them with. We name such monads accordingly:

*Definition 2.6.* A monad  $T$  on a category with an initial object  $\emptyset$  is **constant-free** iff its unit at  $\emptyset$ ,  $\eta_\emptyset^T: \emptyset \rightarrow T(\emptyset)$ , is an isomorphism.

**PROPOSITION 2.7.** *For any distributive law  $\delta: TS \rightarrow ST$  with  $T$  constant-free, the monad morphism  $S\eta^T: S \rightarrow ST$  is admissible.*

**PROOF.** Immediate. □

This result might seem purely academic, but in fact all of our applications arise in this way. The technical core of the paper thus consists in designing tools to construct such distributive laws over constant-free monads.

## 3 SIMPLE STRUCTURAL LAWS

In this section, we introduce our first construction of distributive laws with constant-free auxiliary monad, from what we call simple structural laws. Let us start by working on a simple (perhaps surprising) example, and abstract over it in the following subsection.

### 3.1 On a simple example

We consider the unary (Peano) natural numbers, viewed as the initial algebra of the endofunctor  $\Sigma$  on sets defined by  $\Sigma(X) = 1 + X$ . We use standard syntax, i.e.,  $0$  for the constant and  $s$  for successor. Let us now consider the category, say  $\Sigma$  - **alg<sup>add</sup>**, of  $\Sigma$ -algebras  $X$  equipped with a binary operation, denoted by  $(x, y) \mapsto x + y$ , satisfying the following equations.

$$s(x) + y = s(x + y) \qquad 0 + y = y$$

We call such an operation an addition.

**PROPOSITION 3.1.** *The initial algebra  $\Sigma^*(\emptyset)$  is equipped with a unique addition, which makes it initial in the category of  $\Sigma$ -algebras with addition.*

**PROOF.** An easy induction. □

Furthermore, the forgetful functor  $U^{add}: \Sigma\text{-alg}^{add} \rightarrow \mathbf{Set}$  is monadic, and the corresponding monad, say  $S^{add}$  maps any set  $X$  of variables to terms generated from  $0$  and all  $x \in X$  by  $s$  and  $+$ , modulo the above equations.

Now, for each set  $X$ , there is an obvious inclusion  $SX \hookrightarrow S^{add}X$ . This family of inclusions induces a monad morphism  $S \rightarrow S^{add}$ , and it is admissible by Propositions 3.1 and 2.4.

Let us now describe the monad  $S^{add}$  and its relation to  $S$  more carefully, which will lead us to distributive laws.

Orienting equations, terms generated by  $0$ ,  $s$ , and  $+$  from a given set of variables, quotiented by the equations, have a normal form in which the first argument of an addition is either a variable, or a further addition. Otherwise said, normal forms with variables in  $X$  are generated by the following grammar:

$$\begin{aligned} e & ::= 0 \mid s(e) \mid a \\ a & ::= x \mid a + e. \end{aligned} \tag{1}$$

Clearly, if no variable is available, only the first two cases can occur in a term  $e$ , which together correspond precisely to the syntax generated by  $S$ . We thus recover the fact that  $S\emptyset \rightarrow S^{add}\emptyset$  is an isomorphism.

Let us replay this reasoning, at a slightly more abstract level. The starting point is to refine the arity of the auxiliary function (addition), here the endofunctor  $X \mapsto X^2$ , into a bifunctor

$$\begin{aligned} \Gamma: \mathbf{Set}^2 & \rightarrow \mathbf{Set} \\ (X, Y) & \mapsto X \times Y. \end{aligned}$$

Intuitively, this allows us to distinguish the “decreasing” occurrence of the argument.

By mimicking the recursive definition of addition on  $S\emptyset$ , we then define the following natural transformation

$$\begin{aligned} d_{X,Y}: \Gamma(\Sigma(X), Y) & \rightarrow S(\Gamma(X, Y) + Y) \\ (0, y) & \mapsto \eta^S(\text{in}_2(y)) \\ (s(x), y) & \mapsto s(\eta^S(\text{in}_1(x, y))), \end{aligned} \tag{2}$$

where we write elements of  $\Sigma(Z)$  as terms of depth 1 in  $S(Z)$ , for any set  $Z$ . (We will use a slightly more general codomain in the abstract case.) Elements of the domain are thought of as patterns in the first argument, and the natural transformation maps them to “definition bodies”, which are basic terms generated from the auxiliary arguments in  $Y$ , and potentially a “recursive call” in  $\Gamma(X, Y)$  — hence with “strictly smaller” main argument.

We furthermore define a monad  $T := \Gamma_S^*$ , which, we recall from §1, denotes  $(\Gamma_S \Delta)^*$ . Concretely,  $T(X) := \mu A.(X + A \times S(A))$  corresponds to the syntactic category  $a$  above: terms consist of additions in normal form, i.e., additions whose first argument may be a variable or some further addition, inductively, and whose second argument is an arbitrary expression. We thus have  $S^{add} \cong S \circ T$ .

Finally, we construct a distributive law  $TS \rightarrow ST$ , by induction from the natural transformation (2), which repeatedly applies oriented equations until some normal form is reached. We will be done if we prove  $T(\emptyset) \cong \emptyset$ , for then

$$S^{add}(\emptyset) \cong ST(\emptyset) \cong S(\emptyset),$$

as desired. But  $T(\emptyset) \cong \emptyset$  follows from the next lemma with  $\Theta = \Gamma_S$ .

LEMMA 3.2. *For any category  $\mathbf{C}$  and functor  $\Theta: \mathbf{C}^2 \rightarrow \mathbf{C}$ , if  $\Theta$  is cocontinuous in its first argument and finitary in its second argument, then the initial object possesses a unique  $\Theta\Delta$ -algebra structure, which furthermore makes it initial in  $\Theta\Delta$ -alg.*

PROOF. By cocontinuity,  $\Theta_0\emptyset$  is initial, hence there is a unique morphism  $\Theta_0\emptyset \rightarrow \emptyset$ . The rest follows easily.  $\square$

It may not be entirely obvious that  $\Gamma_S$  is cocontinuous in its first argument, but  $X \times Y$  is isomorphic to the coproduct  $\sum_{y \in Y} X$ , which is cocontinuous by interchange of colimits. Most of our examples below follow the same pattern.

### 3.2 The abstract case

In this subsection, we introduce the general notion of simple structural law, and construct, from any such law, a monad distributive law with constant-free auxiliary monad.

*Definition 3.3.* A **simple structural law** on a given locally finitely presentable category  $\mathbf{C}$  consists of

- a **basic** finitary endofunctor  $\Sigma: \mathbf{C} \rightarrow \mathbf{C}$ ,
- an **auxiliary** functor  $\Gamma: \mathbf{C}^2 \rightarrow \mathbf{C}$  which is cocontinuous in its first argument and finitary in its second argument, and
- a natural transformation

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow S(\Gamma_{S(Y)}(X) + X + Y),$$

where  $S := \Sigma^*$  denotes the free monad generated by  $\Sigma$ .

**Remark 3.** *Comparing with the example (2) of the previous section, we have added a new base case  $X$ , and replaced  $Y$  with the more general  $S(Y)$  in the recursive call.*

Let us now state the construction result, relying on the following lemma.

LEMMA 3.4. *For any finitary bifunctor  $B: \mathbf{C}^2 \rightarrow \mathbf{C}$  which is cocontinuous in its first argument,  $B^*$  is constant-free.*

PROOF. More generally, if an endofunctor  $F$  preserves the initial object  $\emptyset$ , then  $\emptyset$  equipped with the isomorphism  $F(\emptyset) \rightarrow \emptyset$  is easily seen to be the initial  $F$ -algebra, hence is isomorphic to  $F^*\emptyset$ .  $\square$

THEOREM 3.5. *Any simple structural law*

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow S(\Gamma_{S(Y)}(X) + X + Y)$$

(with again  $S = \Sigma^*$ ) induces a monad distributive law

$$TS \rightarrow ST,$$

where  $T := \Gamma_S^*$ , making the following diagram commute.

$$\begin{array}{ccc}
 \Gamma_X \Sigma X & \xrightarrow{d_{X,X}} & S(\Gamma_{S(X)} X + X + X) \\
 \Gamma_{\eta_X^S} \eta_{\Sigma, X} \downarrow & & \downarrow S[\eta_{T_S, X}, \eta_X^T, \eta_X^T] \\
 \Gamma_{S(X)} \Sigma X & & \\
 \Gamma_{\eta_{S(X)}^S} \Sigma X \downarrow & & \\
 \Gamma_{S(S(X))} \Sigma X & & \\
 \eta_{T_S, S(X)} \downarrow & & \\
 TSX & \xrightarrow{\delta_X} & STX
 \end{array} \tag{3}$$

Furthermore,  $T$  being constant-free by Lemma 3.4, the monad morphism  $S \rightarrow ST$  is admissible.

PROOF. This is a special case of Theorem 4.10 below, instantiating  $T$  with the identity monad.  $\square$

**Remark 4.** By the usual characterisation of free algebras for finitary endofunctors [Reiterman 1977], we have

$$T(X) = \mu A.(X + \Gamma_{SA}(A)).$$

### 3.3 Augmented algebras

As a bonus, we may characterise algebras for the generated monad  $ST$ , which we call augmented algebras.

*Definition 3.6.*

- An **algebra** for a simple structural law  $L = (\Sigma, \Gamma, d)$  on  $\mathbf{C}$  consists of an object  $X \in \mathbf{C}$ , equipped with
  - $\Sigma$ -algebra structure  $\mathbf{a}: \Sigma(X) \rightarrow X$  and
  - $\Gamma\Delta$ -algebra structure  $\mathbf{b}: \Gamma_X(X) \rightarrow X$ ,
 making the following diagram commute,

$$\begin{array}{ccc}
 \Gamma_X(\Sigma(X)) & \xrightarrow{d_{X,X}} & S(\Gamma_{SX}(X) + X + X) \\
 \downarrow \Gamma_X(\mathbf{a}) & & \downarrow S(\Gamma_{\bar{\mathbf{a}}}(X) + X + X) \\
 \Gamma_X(X) & & S(\Gamma_X(X) + X + X) \\
 & & \downarrow S[\mathbf{b}, X, X] \\
 \Gamma_X(X) & & SX \\
 \searrow \mathbf{b} & & \swarrow \bar{\mathbf{a}} \\
 & X &
 \end{array}$$

where  $\bar{\mathbf{a}}: S(X) \rightarrow X$  is freely induced by  $\mathbf{a}$ .

- A morphism  $X \rightarrow Y$  of algebras for  $L = (\Sigma, \Gamma, d)$  is a morphism between underlying objects which is both a morphism of  $\Sigma$ - and  $\Gamma\Delta$ -algebras.
- Let  $L$ -**alg** denote the category of algebras for  $L$ , or  $L$ -algebras.

PROPOSITION 3.7. Let  $L = (\Sigma, \Gamma, d)$  be any simple structural law on  $\mathbf{C}$ , and let  $T = \Gamma_S^*$ . Then we have

$$L\text{-alg} \cong ST\text{-Alg}$$

over  $\mathbf{C}$ , where we recall from the basic notations of §1 that capital **Alg** denotes monad algebras.

PROOF. This is a special case of Theorem 4.13 below, instantiating  $T$  with the identity monad.  $\square$

### 3.4 Application: evaluation contexts

Let us present a first application of Theorem 3.5, to generate a simple language with context application. The language is generated by the following grammar

$$\begin{aligned}
 e, f &::= x \mid e f \mid \lambda x. e \\
 E &::= \square \mid E e
 \end{aligned}$$

and context application is defined inductively by

$$\begin{aligned}
 \square[e] &= e \\
 (E e)[f] &= E[f] e.
 \end{aligned}$$

We now define a simple structural law, whose basic monad is the term monad, including contexts, and whose associated auxiliary monad will account for context application.

- Following Fiore et al. [1999], at least in spirit, we first choose as ambient category the category  $[\mathbf{Set}, \mathbf{Set}^2]_f$  of finitary functors  $\mathbf{Set} \rightarrow \mathbf{Set}^2$ , or equivalently  $[\mathbb{F}, \mathbf{Set}^2]$ , where  $\mathbb{F}$  denotes the category of finite ordinals and all maps between them.

We call  $\mathbf{p}$  and  $\mathbf{c}$  the two elements of 2, respectively for “program” and “context”.

For any object  $X \in [\mathbb{F}, \mathbf{Set}^2]$  and  $n \in \mathbb{F}$ ,  $X(n)$  is a pair of sets, which we denote by  $(X(n)_{\mathbf{p}}, X(n)_{\mathbf{c}})$ . We think of

- $X(n)_{\mathbf{p}}$  as a set of **programs** with  $n$  free variables, and of
- $X(n)_{\mathbf{c}}$  as a set of **contexts** with  $n$  free variables.

- The basic endofunctor is defined by:

$$\begin{aligned} \Sigma(X)(n)_{\mathbf{p}} &= n + X(n)_{\mathbf{p}}^2 + X(n+1)_{\mathbf{p}} \\ (e, f) &::= x \mid e f \mid \lambda x. e \\ \Sigma(X)(n)_{\mathbf{c}} &= 1 + X(n)_{\mathbf{c}} \times X(n)_{\mathbf{p}} \\ (E) &::= \square \mid E e. \end{aligned}$$

- The auxiliary bifunctor is

$$\begin{aligned} \Gamma(X, Y)(n)_{\mathbf{p}} &= X(n)_{\mathbf{c}} \times Y(n)_{\mathbf{p}} \\ \Gamma(X, Y)(n)_{\mathbf{c}} &= \emptyset. \end{aligned}$$

- For our simple structural law, which is trivial by construction at  $\mathbf{c}$ , we take at  $\mathbf{p}$ :

$$\begin{aligned} \Sigma(X)(n)_{\mathbf{c}} \times Y(n)_{\mathbf{p}} &\rightarrow S(\Gamma(X, S(Y)) + X + Y)(n)_{\mathbf{p}} \\ (\square, e) &\mapsto \eta^S(\mathit{in}_3(e)) \\ (E f, e) &\mapsto \eta^S(\mathit{in}_1(E, \eta^S(e))) \eta^S(\mathit{in}_2(f)). \end{aligned}$$

### 3.5 Application: capture-avoiding substitution

In this section, we present a third application of Theorem 3.5, to pure  $\lambda$ -calculus with capture-avoiding substitution. This will be subsumed by §5.5, but we find it instructive to unfold the development on a concrete example.

- We take as ambient category the category  $[\mathbf{Set}, \mathbf{Set}]_f$  of finitary endofunctors of sets, or equivalently  $[\mathbb{F}, \mathbf{Set}]$ .
- The basic endofunctor on  $[\mathbb{F}, \mathbf{Set}]$  is defined by

$$\begin{aligned} \Sigma(X)(n) &= n + X(n)^2 + X(n+1) \\ (e, f) &::= x \mid e f \mid \lambda e. \end{aligned}$$

- The auxiliary bifunctor is [Fiore et al. 1999]’s substitution tensor product

$$(X \otimes Y)(n) = \int^{\mathbf{p}} X(\mathbf{p}) \times Y(n)^{\mathbf{p}},$$

or, expressed in  $[\mathbf{Set}, \mathbf{Set}]_f$ ,  $(X \otimes Y)(n) = X(Y(n))$ .

- For our simple structural law, we take (sometimes omitting  $\eta^S$  for readability)

$$\begin{aligned} \int^{\mathbf{p}} \Sigma(X)(\mathbf{p}) \times Y(n)^{\mathbf{p}} &\rightarrow S(X \otimes S(Y)) + X + Y)(n) \\ (x, \sigma) &\mapsto \mathit{in}_3(\sigma(x)) \\ (e f, \sigma) &\mapsto \mathit{in}_1(e, \eta^S \circ \sigma) \mathit{in}_1(f, \eta^S \circ \sigma) \\ (\lambda e, \sigma) &\mapsto \lambda(\mathit{in}_1(e, \sigma^\uparrow)), \end{aligned}$$

where  $\sigma^\uparrow: p+1 \rightarrow S(Y)(n+1)$  denotes the copairing of the following two maps.

$$p \xrightarrow{\sigma} Y(n) \xrightarrow{Y(in_1)} Y(n+1) \xrightarrow{\eta_{Y,n+1}^S} S(Y)(n+1) \quad 1 \xrightarrow{in_2} n+1 \xrightarrow{in_1} \Sigma(Y)(n+1) \hookrightarrow S(Y)(n+1)$$

**Remark 5.** *In the final case, we crucially rely on the recursive call being in  $\Gamma(X, SY)$ , rather than just  $\Gamma(X, Y)$ . In Fiore et al. [1999], this is done by taking  $Y$  “pointed”, in the sense of being equipped with a natural transformation  $n \rightarrow Y(n)$ .*

**Remark 6.** *We obtain the expected syntax and substitution, but not yet the following standard equations,*

$$e[\sigma][\sigma'] = e[\sigma[\sigma']] \qquad e[\text{id}] = e$$

where  $\text{id}: n \rightarrow S(X)(n)$  picks the variables. We will complete the picture in §5.

### 3.6 Application: binding contexts

For a slightly more involved example, we consider in this subsection the **sharing**  $\lambda$ -calculus [Accattoli 2019, §4.1] (but see also [Hirschowitz et al. 2009; Sewell et al. 2008]).

Following Accattoli, the syntax is given by

$$\begin{aligned} e, f &::= x \mid e f \mid \lambda x. e \mid e \langle x \mapsto f \rangle \\ E &::= \square \mid E \langle x \mapsto f \rangle. \end{aligned}$$

Context application is then defined inductively by

$$\begin{aligned} \square[e] &= e \\ (E \langle x \mapsto f \rangle)[e] &= E[e] \langle x \mapsto f \rangle. \end{aligned}$$

**Remark 7.** *Context application may give rise to variable capture. E.g.,  $(\square \langle x \mapsto f \rangle)[x] = x \langle x \mapsto f \rangle$ .*

In order to model this, we extend the setting of the previous section as follows. In §3.4, for any functor  $X: \mathbb{F} \rightarrow \mathbf{Set}^2$ , we thought of  $X(n)_p$  and  $X(n)_c$  as sets of programs, resp. contexts with  $n$  free variables. We now need to refine this point of view, and index contexts over the number of **capturing** variables, i.e., variables bound above the context hole  $\square$ . Instead of functors  $\mathbb{F} \rightarrow \mathbf{Set}^2$ , we thus consider functors  $\mathbb{F} \rightarrow \mathbf{Set}^{1+\mathbb{N}}$ . Of course, we have  $\mathbb{N} \cong 1 + \mathbb{N}$ , but we write  $1 + \mathbb{N}$  to emphasise the fact that  $in_1(\star)$ , the unique element of the left-hand summand, models terms, while each  $in_2(n)$  models contexts with  $n$  capturing variables.

**Notation 1.** *We abbreviate  $in_1(\star)$  to  $\mathbf{p}$  and  $in_2(n)$  to  $\mathbf{c}_n$ , so that, e.g.,  $X(n)_{\mathbf{c}_m}$  is thought of as a set of contexts with  $n$  free variables and  $m$  capturing variables.*

Accordingly, we specify the syntax by the endofunctor

$$\begin{aligned} \Sigma(X)(n)_{\mathbf{p}} &= n + X(n)_{\mathbf{p}}^2 + X(n+1)_{\mathbf{p}} + X(n+1)_{\mathbf{p}} \times X(n)_{\mathbf{p}} \\ \Sigma(X)(n)_{\mathbf{c}_{m+1}} &= X(n+1)_{\mathbf{c}_m} \times X(n)_{\mathbf{p}} \\ \Sigma(X)(n)_{\mathbf{c}_0} &= 1. \end{aligned}$$

**Remark 8.** *On the second line, the expression  $X(n+1)_{\mathbf{c}_m} \times X(n)_{\mathbf{p}}$  reflects the fact that in the above grammar, in  $E \langle x \mapsto f \rangle$ ,  $E$  may use the bound variable  $x$ , hence the use of  $n+1$ , and has one less capturing variable than  $E \langle x \mapsto f \rangle$ , hence the passing from  $m+1$  to  $m$ . Otherwise said,  $x$  is free in  $E$ , but capturing in  $E \langle x \mapsto f \rangle$ .*

The generated monad  $S = \Sigma^*$  may be presented syntactically by the following rules

$$\frac{k \in X(n)_p}{n \vdash k : p} \quad \frac{i \in n}{n \vdash x_i : p} \quad \frac{n \vdash e : p \quad n \vdash f : p}{n \vdash e f : p} \quad \frac{n+1 \vdash e : p}{n \vdash \lambda(e) : p}$$

$$\frac{n+1 \vdash e : p \quad n \vdash f : p}{n \vdash e\langle f \rangle : p} \quad \frac{K \in X(n)_{c_m}}{n; m \vdash K : c} \quad \frac{}{n; 0 \vdash \square : c} \quad \frac{n+1; m \vdash E : c \quad n \vdash f : p}{n; m+1 \vdash E\langle f \rangle : c},$$

with  $S(n)_p = \{e \mid n \vdash e : p\}$  and  $S(n)_{c_m} = \{E \mid n; m \vdash E : c\}$ . One then straightforwardly defines the functorial action in  $n$ : for any renaming  $\rho: n \rightarrow n'$ , one replaces all  $x_i$  with  $x_{\rho(i)}$ , for  $i \in n$ .

Let us now model context application by a simple structural law:

- the arity functor is defined by

$$\begin{aligned} \Gamma(X, Y)(n)_{c_m} &= \emptyset \\ \Gamma(X, Y)(n)_p &= \sum_{m \in \mathbb{N}} X(n)_{c_m} \times Y(n+m)_p, \end{aligned}$$

reflecting the fact that in a context application  $E[e]$ ,  $e$  has as free variables the disjoint union of the free and capturing variables of  $E$ ;

- the simple structural law is defined componentwise by

$$\begin{aligned} \Sigma(X)(n)_{c_m} \times Y(n+m)_p &\rightarrow S(\Gamma(X, S(Y)) + X + Y)(n)_p \\ (\square, e) &\mapsto \mathit{in}_3(e) && \text{(if } m = 0\text{)} \\ (E\langle f \rangle, e) &\mapsto \mathit{in}_1(E, e)\langle \mathit{in}_2(f) \rangle && \text{(if } m = m' + 1\text{),} \end{aligned}$$

for all  $n, m \in \mathbb{N}$  (omitting  $\eta^S$  again for readability).

**Remark 9.** *The reason  $e$  may be applied to both  $E$  and  $E\langle f \rangle$  is that both contexts have the same number of free or capturing variables ( $(n+1) + m'$  and  $n + (m' + 1)$ , respectively).*

### 3.7 Application: named substitution

Let us now consider a last illustration of simple structural laws: named substitution in  $\lambda\mu$ -calculus [Parigot 1992; Vaux 2007]. Its usual form is as follows. The syntax of  $\lambda\mu$ -calculus is:

$$\begin{aligned} e, f, g &::= x \mid e f \mid \lambda x. e \mid \mu \alpha. c \\ c, d &::= [\alpha]e. \end{aligned}$$

There are two syntactic categories: **programs**, ranged over by  $e, f, g, \dots$ , and **continuations**, ranged over by  $c, d$ . Accordingly, there are two kinds of variables: **program variables**, ranged over by  $x, y, z, \dots$ , and **continuation variables**, ranged over by  $\alpha, \beta, \dots$

**Remark 10.** *In this subsection,  $\mu$  always denote the syntactic operation, as opposed to any monad multiplication, or least fixed-point operator.*

Intuitively, named substitution  $(t)_\alpha g$  (notation from [Vaux 2007, Definition 6.5]) takes as argument any term  $t$  (program or continuation) with a distinguished continuation variable  $\alpha$ , together with a program  $g$ , and replaces all subterms of the form  $[\alpha]e$  with  $[\alpha](e g)$  in  $t$ . This may be defined recursively by:

$$\begin{aligned} (x)_\alpha g &= x \\ (\lambda x. e)_\alpha g &= \lambda x. ((e)_\alpha g) && (x \notin g) \\ (e f)_\alpha g &= ((e)_\alpha g) ((f)_\alpha g) \\ (\mu \beta. c)_\alpha g &= \mu \beta. ((c)_\alpha g) && (\beta \notin \alpha, g) \\ ([\beta]e)_\alpha g &= \begin{cases} [\alpha](((e)_\alpha g) g) & \text{(if } \alpha = \beta\text{)} \\ [\beta]((e)_\alpha g) & \text{(otherwise).} \end{cases} \end{aligned}$$

In order to model this categorically, since we have two syntactic categories, each with its own set of variables, we work with finitary functors  $\mathbf{Set}^2 \rightarrow \mathbf{Set}^2$ , or equivalently the category  $[\mathbb{F}^2, \mathbf{Set}^2]$ . We write  $\mathbf{p}$  and  $\mathbf{c}$  for the elements of  $2$  (respectively standing for “program” and “continuation”). Thus, for any  $X \in [\mathbb{F}^2, \mathbf{Set}^2]$ , we think of

- $X(m, n)_{\mathbf{p}}$  as a set of programs with  $m$  free program variables and  $n$  continuation variables, and of
- $X(m, n)_{\mathbf{c}}$  as a set of continuations with  $m$  free program variables and  $n$  continuation variables.

The basic syntax is specified by the endofunctor  $\Sigma$  defined by

$$\begin{aligned}\Sigma(X)(m, n)_{\mathbf{p}} &= m + X(m, n)_{\mathbf{p}}^2 + X(m+1, n)_{\mathbf{p}} + X(m, n+1)_{\mathbf{c}} \\ \Sigma(X)(m, n)_{\mathbf{c}} &= n \times X(m, n)_{\mathbf{p}}.\end{aligned}$$

For specifying named substitution, we take as auxiliary bifunctor

$$\begin{aligned}\Gamma(X)(m, n)_{\mathbf{p}} &= X(m, n)_{\mathbf{p}} \times n \times Y(m, n)_{\mathbf{p}} \\ \Gamma(X)(m, n)_{\mathbf{c}} &= X(m, n)_{\mathbf{c}} \times n \times Y(m, n)_{\mathbf{p}}.\end{aligned}$$

**Notation 2.** *Slightly generalising previous notation, we denote by  $x_i$  the  $i$ th program variable, for  $i \in m$ , and by  $\alpha_j$  the  $j$ th continuation variable, for  $j \in n$ .*

Using this notation, we model named substitution by the following simple structural law (omitting  $\eta^S$  again for readability):

$$\begin{aligned}d_{X,Y,m,n,\mathbf{p}}: \Sigma(X)(m, n)_{\mathbf{p}} \times n \times Y(m, n)_{\mathbf{p}} &\rightarrow S(\Gamma(X, SY) + X + Y)(m, n)_{\mathbf{p}} \\ (x_i, j, g) &\mapsto in_2(x_i) \\ (\lambda(e), j, g) &\mapsto \lambda(in_1(e, j, w_{m,n}^{\mathbf{p}} \cdot g)) \\ (e f, j, g) &\mapsto in_1(e, j, g) in_1(f, j, g) \\ (\mu(e), j, g) &\mapsto \mu(in_1(e, j, w_{m,n}^{\mathbf{c}} \cdot g)) \\ \\ d_{X,Y,m,n,\mathbf{c}}: \Sigma(X)(m, n)_{\mathbf{c}} \times n \times Y(m, n)_{\mathbf{p}} &\rightarrow S(\Gamma(X, SY) + X + Y)(m, n)_{\mathbf{c}}, \text{ i.e.,} \\ d_{X,Y,m,n,\mathbf{c}}: n \times X(m, n)_{\mathbf{p}} \times n \times Y(m, n)_{\mathbf{p}} &\rightarrow S(\Gamma(X, SY) + X + Y)(m, n)_{\mathbf{c}} \\ ([\alpha_j]e, j, g) &\mapsto [\alpha_j](in_1(e, j, g) in_3(g)) \\ ([\alpha_{j'}]e, j, g) &\mapsto [\alpha_{j'}]in_1(e, j, g) \quad (\text{for } j' \neq j),\end{aligned}$$

where

- $w_{m,k}^{\mathbf{p}} := (in_1, id_k): (m, k) \hookrightarrow (m+1, k)$ ;
- $w_{m,k}^{\mathbf{c}} := (id_m, in_1): (m, k) \hookrightarrow (m, k+1)$ ; and
- $u \cdot a$  denotes the action of a morphism  $u: (m, k) \rightarrow (m', k')$  in  $\mathbb{F}^2$  on an element  $a$  of some  $A(m, k)_s$ , for some  $A: \mathbb{F}^2 \rightarrow \mathbf{Set}^2$  and  $s \in \{\mathbf{p}, \mathbf{c}\}$ .

#### 4 INCREMENTAL STRUCTURAL LAWS

In this section, we introduce the incremental variant of structural laws. In §4.1, we sketch the idea on a simple example. In §4.2, we introduce **incremental structural laws** in full generality, and prove that they induce admissible morphisms (Theorem 4.2). In §4.3, we characterise the composite monad induced by an incremental structural law, as objects equipped with suitably compatible algebra structures. Then, in §4.4, we explain how to combine several, independent incremental structural laws. Finally, we cover applications: partial differentiation in the differential  $\lambda$ -calculus (§4.5), and De Bruijn’s presentation of capture-avoiding substitution in the  $\lambda$ -calculus (§4.6).

#### 4.1 On a simple example

Recalling the distributive law, say  $\delta: TS \rightarrow ST$  constructed in §3.1 from the simple structural law (2), we now want to extend the language with a second binary operation, multiplication, satisfying

$$\begin{aligned} s(x) \times y &= (x \times y) + y \\ 0 \times y &= 0. \end{aligned}$$

Furthermore, the full quotiented term language yields a monad  $S^{add,mul}$ , and a monad morphism  $S \rightarrow S^{add,mul}$ . As in §3.1, orienting equations, we obtain normal forms, which are generated for any argument  $X$  by the grammar

$$\begin{aligned} e &::= 0 \mid s(e) \mid b \\ b &::= x \mid b + e \mid b \times e. \end{aligned}$$

Of course, by the same reasoning as for mere addition, the initial  $S$ -algebra is initial in the category of models of the whole language (i.e.,  $S$ -algebras with addition and multiplication, satisfying all four equations), hence the monad morphism  $S \rightarrow S^{add,mul}$  is admissible.

Categorically, because the new equation uses addition, we cannot define a second, independent structural extension of  $S$ . However, the recursive definition of multiplication yields a natural transformation

$$\Theta(\Sigma(X), Y) \rightarrow ST(\Theta(X, Y) + Y), \quad (4)$$

where  $\Theta(X, Y) := X \times Y$ . Furthermore, inspecting the above grammar for normal forms, we find  $S^{add,mul} \cong S \circ (\Gamma_S + \Theta_S)^*$ : from the top level, we have a first layer of basic operations, until we meet a binary operation; the first argument of the latter may then only consist of binary operations until it reaches a variable  $x \in X$ , while the second argument is arbitrary. Equivalently, letting  $T' = \Theta_S^*$ , we have

$$(\Gamma_S + \Theta_S)^* \cong \Gamma_S^* \oplus \Theta_S^* \cong T \oplus T',$$

hence

$$S^{add,mul} \cong S \circ (T \oplus T').$$

By induction, i.e., applying (4) repeatedly, we define a distributive law  $(T \oplus T') \circ S \rightarrow S \circ (T \oplus T')$ .

Of course  $\Theta$  is cocontinuous in its first argument, which easily entails that  $T \oplus T'$  is constant-free, and so  $S \rightarrow S^{add,mul} \cong S \circ (T \oplus T')$  is admissible.

#### 4.2 The abstract case

*4.2.1 Main result.* Let us now abstract over the previous section.

*Definition 4.1.* An **incremental structural law** on a locally finitely presentable category  $\mathbf{C}$  consists of

- a distributive law  $\delta: TS \rightarrow ST$  with free basic monad  $S = \Sigma^*$  and constant-free auxiliary monad  $T$ , together with
- a functor  $\Gamma: \mathbf{C}^2 \rightarrow \mathbf{C}$  which is cocontinuous in its first argument and finitary in its second argument, equipped with
- a natural transformation

$$\Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{STY}(X) + X + Y).$$

**THEOREM 4.2.** *Any incremental structural law*

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{ST(Y)}(X) + X + Y)$$

over  $\delta: TS \rightarrow ST$ , with  $S := \Sigma^*$ , induces a distributive law

$$d_{j\delta}: (T \oplus T')S \rightarrow S(T \oplus T'),$$

where  $T' = \Gamma_S^*$ , making the following diagram commute.

$$\begin{array}{ccc}
 \Gamma_X \Sigma X & \xrightarrow{d_{X,X}} & ST(\Gamma_{STX} X + X + X) \\
 \Gamma_{\eta_X^S} \eta_{\Sigma, X} \downarrow & & \downarrow ST(\Gamma_{STX} \eta_X^T + [X, X]) \\
 \Gamma_{SX} SX & & ST(\Gamma_{STX} TX + X) \\
 \Gamma_{\eta_{SX}^S} SX \downarrow & & \downarrow ST(\eta_{\Gamma_S, TX+X}) \\
 \Gamma_{SSX} SX & & ST(T' TX + X) \\
 \eta_{\Gamma_S, SX} \downarrow & & \downarrow ST(in_2 in_1 X + X) \\
 T' SX & & ST((T \oplus T')(T \oplus T')X + X) \\
 in_2 SX \downarrow & & \downarrow Sin_1[\mu_X^{T \oplus T'}, \eta^{T \oplus T'} X] \\
 (T \oplus T') SX & \xrightarrow{d_{j\delta X}} & S(T \oplus T')(T \oplus T')X \\
 & & \downarrow S\mu_X^{T \oplus T'}
 \end{array} \tag{5}$$

Furthermore,  $T \oplus T'$  is constant-free, hence the monad morphism

$$S \rightarrow S(T \oplus T')$$

is admissible.

The next subsection is devoted to sketching the proof, and may be safely ignored.

**4.2.2 Proofsketch.** Our first step will consist in defining an intermediate notion called incremental lifting, fitting in the following process:

incremental structural law  $\rightarrow$  incremental lifting  $\rightarrow$  distributive law  $\rightarrow$  admissible morphism.

To explain the idea, let us start by recalling from Beck [1969] that distributive laws are equivalent to monad liftings.

*Definition 4.3.* A **lifting** of a monad  $S: \mathbf{C} \rightarrow \mathbf{C}$  along a functor  $U: \mathbf{E} \rightarrow \mathbf{C}$  is a monad  $S': \mathbf{E} \rightarrow \mathbf{E}$  such that the following square commutes

$$\begin{array}{ccc}
 \mathbf{E} & \xrightarrow{S'} & \mathbf{E} \\
 U \downarrow & & \downarrow U \\
 \mathbf{C} & \xrightarrow{S} & \mathbf{C}
 \end{array}$$

and, furthermore,  $U$  preserves multiplication and unit, i.e., for all  $E \in \mathbf{E}$ ,  $U(\mu_E^{S'}) = \mu_{U(E)}^S$  and  $U(\eta_E^{S'}) = \eta_{U(E)}^S$ .

Given any monads  $T$  and  $S$  on a category  $\mathbf{C}$ , a  **$T$ -lifting** of  $S$  is a lifting of  $S$  along the forgetful functor  $T\text{-Alg} \rightarrow \mathbf{C}$ .

**PROPOSITION 4.4.** *For all  $S$  and  $T$ ,  $T$ -liftings of  $S$  are in one-to-one correspondence with endofunctors  $S': T\text{-Alg} \rightarrow T\text{-Alg}$ , such that the following square commutes*

$$\begin{array}{ccc}
 T - \mathbf{Alg} & \xrightarrow{S'} & T - \mathbf{Alg} \\
 \downarrow & & \downarrow \\
 \mathbf{C} & \xrightarrow{S} & \mathbf{C}
 \end{array}$$

and, for all  $X \in T - \mathbf{Alg}$ ,  $\mu_X^S: SSX \rightarrow SX$  and  $\eta_X^S: X \rightarrow SX$  are  $T$ -algebra morphisms.

PROOF. Straightforward. □

LEMMA 4.5 (BECK [1969, §1]). For any monads  $S, T: \mathbf{C} \rightarrow \mathbf{C}$ , monad distributive laws  $TS \rightarrow ST$  are in one-to-one correspondence with  $T$ -liftings of  $S$ .

We now introduce incremental liftings.

**Definition 4.6.** Let  $S, T$ , and  $T'$  be monads on  $\mathbf{C}$ , and  $\delta: TS \rightarrow ST$  be a monad distributive law. An **incremental lifting** of  $S$  to  $T' - \mathbf{Alg}$  along  $\delta$  is a functorial assignment, to each pair of a  $T$ -algebra structure and a  $T'$ -algebra structure on  $X$ , of a  $T'$ -algebra structure on  $SX$ , such that for each such  $X$ , the multiplication  $SSX \rightarrow SX$  and unit  $X \rightarrow SX$  are  $T'$ -algebra morphisms.

This rather technical definition in fact unfolds rather simply, as we now explain. We first recall the following, well-known characterisation of  $(T \oplus T')$ -algebras.

LEMMA 4.7. Let  $T$  and  $T'$  be monads on a complete (locally small) category  $\mathbf{C}$  such that the monad coproduct  $T \oplus T'$  exists. Then, the category of  $(T \oplus T')$ -algebras consists of objects of  $\mathbf{C}$  equipped with algebra structures for both  $T$  and  $T'$ . Otherwise said, the following square is a pullback.

$$\begin{array}{ccc}
 (T \oplus T') - \mathbf{Alg} & \longrightarrow & T - \mathbf{Alg} \\
 \downarrow \lrcorner & & \downarrow \\
 T' - \mathbf{Alg} & \longrightarrow & \mathbf{C}
 \end{array}$$

PROOF. See §B. □

**Remark 11.** In passing, this result yields an easy proof that  $T \oplus T'$  is constant-free.

This directly entails the following equivalent presentation of incremental liftings.

COROLLARY 4.8. Incremental liftings of  $S$  to  $T' - \mathbf{Alg}$  along  $\delta$  on a complete category  $\mathbf{C}$  such that the coproduct  $T \oplus T'$  exists are in one-to-one correspondence with liftings along the projection  $(T \oplus T') - \mathbf{Alg} \rightarrow T - \mathbf{Alg}$  of the monad on  $T - \mathbf{Alg}$ , say  $S^\delta$ , itself obtained by lifting  $S$ , as in

$$\begin{array}{ccc}
 (T \oplus T') - \mathbf{Alg} & \xrightarrow{S^{\delta, T'}} & (T \oplus T') - \mathbf{Alg} \\
 \downarrow & & \downarrow \\
 T - \mathbf{Alg} & \xrightarrow{S^\delta} & T - \mathbf{Alg} \\
 \downarrow & & \downarrow \\
 \mathbf{C} & \xrightarrow{S} & \mathbf{C}
 \end{array}$$

In such a situation,  $S^{\delta, T'}$  is furthermore a  $(T \oplus T')$ -lifting of  $S$ .

It is now easy to see that incremental liftings give rise to distributive laws.

COROLLARY 4.9. Let  $S, T$ , and  $T'$  be monads on a complete category  $\mathbf{C}$  such that the monad coproduct  $T \oplus T'$  exists, and let  $\delta: TS \rightarrow ST$  be any monad distributive law. Then, any incremental lifting of  $S$  to  $T' - \mathbf{Alg}$  along  $\delta$  gives rise to a distributive law  $\delta': (T \oplus T')S \rightarrow S(T \oplus T')$ .

Furthermore, if  $T$  and  $T'$  are constant-free, so is  $T \oplus T'$ , hence the induced monad morphism  $S \rightarrow S(T \oplus T')$  is admissible.

PROOF. By Corollary 4.8 and Lemma 4.5. Constant-freeness of  $T \oplus T'$  follows by Lemma 4.7.  $\square$

**Remark 12.** *With the same hypotheses, it is straightforward to deduce that  $ST \rightarrow S(T \oplus T')$  is admissible, since  $S \rightarrow ST$  and  $S \rightarrow ST \rightarrow S(T \oplus T')$  both are.*

We thus mostly reduce Theorem 4.2 to the following.

THEOREM 4.10. *Any incremental structural law*

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{ST(Y)}(X) + X + Y)$$

over  $\delta: TS \rightarrow ST$ , with  $S := \Sigma^*$ , induces an incremental lifting of  $S$  to  $T' - \mathbf{Alg}$  along  $\delta$ , where  $T' = \Gamma_S^*$ .

**Remark 13.** *This result does not directly entail (5), which will follow from the construction of the incremental lifting.*

PROOF SKETCH, SEE §C FOR A COMPLETE PROOF. We first define from  $d$  a natural transformation

$$d_{X,Y}^{\bullet\omega}: \Gamma_{SY}(S(X)) \rightarrow ST(\Gamma_{STY}(X) + X + Y),$$

notably using the fact that  $\Gamma_{STY}$ , being a cocontinuous endofunctor on a locally finitely presentable category, admits a right adjoint.

We then want to construct a lifting of  $S$  to some functor  $(T \oplus T') - \mathbf{Alg} \rightarrow T' - \mathbf{Alg}$ . But we have  $T' - \mathbf{Alg} \cong \Gamma_S - \mathbf{alg}$ , so we reduce to constructing a functor  $(T \oplus T') - \mathbf{Alg} \rightarrow \Gamma_S - \mathbf{Alg}$  – still behaving like  $S$  on underlying objects.

For any  $T$ -algebra  $\mathbf{b}: TX \rightarrow X$  equipped with  $\Gamma_S \Delta$ -algebra structure  $\mathbf{c}: \Gamma_{SX}X \rightarrow X$ , we construct the desired  $\Gamma_S \Delta$ -algebra structure on  $S(X)$  as

$$\Gamma_{SSX}SX \xrightarrow{\Gamma_{\mu_X^S} SX} \Gamma_{SX}SX \xrightarrow{d_{X,X}^{\bullet\omega}} ST(\Gamma_{STX}(X) + X + X) \xrightarrow{ST(\mathbf{b} \triangleright \mathbf{c})} STX \xrightarrow{S\mathbf{b}} SX,$$

where  $(\mathbf{b} \triangleright \mathbf{c}): \Gamma_{STX}X + X + X \rightarrow X$  is defined as follows.

*Definition 4.11.* For any  $T$ -algebra  $\mathbf{b}: TX \rightarrow X$  equipped with  $\Gamma_S \Delta$ -algebra structure  $\mathbf{c}: \Gamma_{SX}X \rightarrow X$ , let  $\mathbf{b} \triangleright \mathbf{c}$  denote the following composite.

$$\Gamma_{STX}(X) + X + X \xrightarrow{\Gamma_{\mathbf{b} \triangleright \mathbf{c}} X + [X, X]} \Gamma_{SX}X + X \xrightarrow{[\mathbf{c}, X]} X.$$

We then show that  $\mu^S$  and  $\eta^S$  are algebra morphisms, as desired.  $\square$

### 4.3 Augmented algebras

Let us now present the announced characterisation of the category of algebras of the composite monad  $S(T \oplus T')$  induced by an incremental structural law, which we call augmented algebras.

*Definition 4.12.* Consider any incremental structural law

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{STY}(X) + X + Y)$$

over  $\delta: TS \rightarrow ST$ , with  $S = \Sigma^*$ .

- A  $(\delta, d)$ -**algebra** is an object  $X$  equipped with morphisms

$$\mathbf{a}: SX \rightarrow X \qquad \mathbf{b}: TX \rightarrow X \qquad \mathbf{c}: \Gamma(X, X) \rightarrow X,$$

the first two of which are monad algebra structures, making the following diagrams commute.

$$\begin{array}{ccc}
 TSX & \xrightarrow{\delta_X} & STX \\
 Ta \downarrow & (d1) & \downarrow Sb \\
 TX & & SX \\
 & \searrow b & \swarrow a \\
 & & X
 \end{array}$$
  

$$\begin{array}{ccc}
 \Gamma(\Sigma X, X) & \xrightarrow{d_{X,X}} & ST(\Gamma(X, STX) + X + X) \\
 \Gamma(\eta_{\Sigma, X}, X) \downarrow & (d2) & \downarrow ST(\Gamma(X, a \circ Sb) + [X, X]) \\
 \Gamma(SX, X) & & ST(\Gamma(X, X) + X) \\
 \Gamma(a, X) \downarrow & & \downarrow ST[c, X] \\
 \Gamma(X, X) & & ST(X) \\
 & \searrow c & \swarrow a \circ Sb \\
 & & X
 \end{array}$$

- A  $(\delta, d)$ -**algebra morphism** is a morphism between underlying objects which is an  $S$ -,  $T$ -, and  $\Gamma\Delta$ -algebra morphism.
- We let  $(\delta, d)$  - **Alg** denote the category of  $(\delta, d)$ -algebras and morphisms between them.

**THEOREM 4.13.** *Consider any incremental structural law*

$$d_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{STY}(X) + X + Y)$$

over  $\delta: TS \rightarrow ST$ , with  $S = \Sigma^*$ . Letting  $T' = \Gamma_S^*$  as before, there is an isomorphism  $S(T \oplus T') - \mathbf{Alg} \cong (\delta, d) - \mathbf{Alg}$  of categories over the base category  $\mathbf{C}$ .

More precisely, for any  $S \circ (T \oplus T')$ -algebra  $X$ , the canonical morphisms from  $SX, TX$ , and  $\Gamma(X, X)$  into  $S((T \oplus T')(X))$  equip  $X$  with  $(\delta, d)$ -algebra structure. This assignment extends to a functor  $S(T \oplus T') - \mathbf{Alg} \rightarrow (\delta, d) - \mathbf{Alg}$  over  $\mathbf{C}$ , which is an isomorphism of categories.

#### 4.4 Independent extensions

Let us end the theoretical part of this section by briefly showing how to combine independent incremental structural laws.

**PROPOSITION 4.14.** *For any set  $I$ , any  $I$ -indexed family*

$$d_{X,Y}^i: \Gamma_Y^i(\Sigma(X)) \rightarrow ST(\Gamma_{ST(Y)}^i(X) + X + Y)$$

of incremental structural laws over  $\delta: TS \rightarrow ST$ , with  $i \in I$  and  $S := \Sigma^*$ , induce a distributive law

$$(T \oplus \bigoplus_i T_i)S \rightarrow S(T \oplus \bigoplus_i T_i),$$

where  $T_i = (\Gamma_S^i)^*$  and  $T \oplus \bigoplus_i T_i$  is constant-free. We thus get an admissible morphism

$$S \rightarrow S(T \oplus \bigoplus_i T_i).$$

**PROOF.** Taking  $\Gamma_Y(X) = \sum_i \Gamma_Y^i(X)$  and observing that  $\bigoplus_i T_i = \Gamma_S^*$ , we obtain the desired result directly by applying Theorem 4.2 to the cotupling of all

$$\Gamma_Y^i(\Sigma(X)) \xrightarrow{d_{X,Y}^i} ST(\Gamma_{ST(Y)}^i(X) + X + Y) \xrightarrow{ST((in_i)_{X, ST(Y)} + X + Y)} ST(\Gamma_{ST(Y)}(X) + X + Y) \quad \square$$

#### 4.5 Application: differential $\lambda$ -calculus

In this section, we sketch an imperfect treatment of differential  $\lambda$ -calculus, to illustrate incremental structural laws.

*4.5.1 Standard definition.* Let us first introduce the calculus in the usual, informal way. The syntax is:

$$\begin{aligned} \text{Simple terms } \ni e, f & ::= x \mid e M \mid \lambda x. e \mid \mathbf{D}e \cdot f \\ \text{Multiterms } \ni M, N & ::= 0 \mid e + M, \end{aligned}$$

where simple and multiterms are considered equivalent modulo the following equations.

$$\begin{aligned} \mathbf{D}(\mathbf{D}e \cdot f) \cdot g & \equiv \mathbf{D}(\mathbf{D}e \cdot g) \cdot f \\ e + f + M & \equiv f + e + M \end{aligned} \quad (6)$$

Structural operations are then defined as follows:

1. First of all, operations are extended to multiterms by induction, as in Figure 1.A. The first two

A. Extended operations	B. Capture-avoiding substitution
$0 + N = N$	$x[x \mapsto M] = M$
$(e + M) + N = e + (M + N)$	$y[x \mapsto M] = y + 0 \text{ (when } x \neq y)$
$0 N = 0$	$(\lambda y. e)[x \mapsto M] = \lambda y. (e[x \mapsto M])$
$(e + M) N = (e N) + M N$	$(y \text{ fresh for } x \text{ and } M)$
$\lambda x. 0 = 0$	$(e N)[x \mapsto M] = e[x \mapsto M] N[x \mapsto M]$
$\lambda x. (e + M) = \lambda x. e + \lambda x. M$	$(\mathbf{D}e \cdot f)[x \mapsto M] = \mathbf{D}e[x \mapsto M] \cdot f[x \mapsto M]$
$\mathbf{D}(e) \cdot 0 = 0$	$0[x \mapsto M] = 0$
$\mathbf{D}(e) \cdot (f + N) = \mathbf{D}(e) \cdot f + \mathbf{D}(e) \cdot N$	$(e + N)[x \mapsto M] = e[x \mapsto M] + N[x \mapsto M]$
$\mathbf{D}(0) \cdot N = 0$	
$\mathbf{D}(e + M) \cdot N = \mathbf{D}(e) \cdot N + \mathbf{D}(M) \cdot N$	
<u>C. Partial differentiation</u>	
$\frac{\partial x}{\partial x} \cdot M = M$	
$\frac{\partial y}{\partial x} \cdot M = 0 \text{ (when } x \neq y)$	
$\frac{\partial(e N)}{\partial x} \cdot M = \left(\frac{\partial e}{\partial x} \cdot M\right) N + \left(\mathbf{D}e \cdot \left(\frac{\partial N}{\partial x} \cdot M\right)\right) N$	
$\frac{\partial \lambda y. e}{\partial x} \cdot M = \lambda y. \left(\frac{\partial e}{\partial x} \cdot M\right) \text{ (} y \text{ fresh for } x \text{ and } M)$	
$\frac{\partial \mathbf{D}e \cdot f}{\partial x} \cdot M = \mathbf{D}\left(\frac{\partial e}{\partial x} \cdot M\right) \cdot f + \mathbf{D}e \cdot \left(\frac{\partial f}{\partial x} \cdot M\right)$	
$\frac{\partial 0}{\partial x} \cdot M = 0$	
$\frac{\partial(e+N)}{\partial x} \cdot M = \frac{\partial e}{\partial x} \cdot M + \frac{\partial N}{\partial x} \cdot M$	

Fig. 1. Auxiliary functions for differential  $\lambda$ -calculus

lines extend  $e + M$  to take a multiterm as its first argument. The next two extend  $e M$  similarly. The next two extend  $\lambda$ -abstraction. Matters then get slightly subtle:  $\mathbf{D} \cdot \cdot$  is first extended to take a multiterm as its second argument, and then the obtained extension is further extended to take a multiterm as its first argument.

2. Then, relying on this, capture-avoiding substitution of a variable by a multiterm in a simple or multiterm is defined by induction in Figure 1.B, the result being a multiterm.

3. Finally, partial differentiation is also defined inductively, relying on extended operations, in Figure 1.C.

4.5.2 *Using incremental structural laws.* Let us now model this syntax and auxiliary operations using incremental structural laws, but ignoring equations for this paper.

We have two syntactic categories, simple terms and multiterms, but only one sort for variables, to be replaced with multiterms. As in §3.4, we model this by working with the category  $[\mathbf{Set}, \mathbf{Set}^2]_f$  of finitary functors  $F: \mathbf{Set} \rightarrow \mathbf{Set}^2$ , or equivalently functors  $\mathbb{F} \rightarrow \mathbf{Set}^2$ . For a change, this time, we emphasise the presentation as functors  $\mathbf{Set} \rightarrow \mathbf{Set}^2$ . Furthermore, we write  $\mathbf{s}$  and  $\mathbf{m}$  for the elements of  $2$ , and think of  $F(X)_\mathbf{s}$  as a set of simple terms with free variables in  $X$ , and of  $F(X)_\mathbf{m}$  as a set of multiterms with free variables in  $X$ . We sometimes write such functors as pairs  $(F_\mathbf{s}, F_\mathbf{m})$  of set-functors.

The basic endofunctor is then defined as follows.

$$\begin{aligned} \Sigma(F)(X)_\mathbf{s} &= X + F(X)_\mathbf{s} \times F(X)_\mathbf{m} + F(X+1)_\mathbf{s} + F(X)_\mathbf{s}^2 \\ e, f &::= x \mid e M \mid \lambda(e) \mid \text{De} \cdot f \\ \Sigma(F)(X)_\mathbf{m} &= 1 + F(X)_\mathbf{s} \times F(X)_\mathbf{m} \\ M &::= 0 \mid e + M \end{aligned}$$

**Notation 3.** Writing  $\mathbf{y}_\mathbf{s} = (1, \emptyset)$  and  $\mathbf{y}_\mathbf{m} = (\emptyset, 1)$  (viewing  $\mathbf{Set}^2$  as the category of presheaves over the discrete category  $\{\mathbf{s}, \mathbf{m}\}$ ), we equivalently have

$$\begin{aligned} \Sigma(F)(X) &= (X + F(X)_\mathbf{s} \times F(X)_\mathbf{m} + F(X+1)_\mathbf{s} + F(X)_\mathbf{s}^2) \cdot \mathbf{y}_\mathbf{s} \\ &\quad + (1 + F(X)_\mathbf{s} \times F(X)_\mathbf{m}) \cdot \mathbf{y}_\mathbf{m}. \end{aligned}$$

Let us now define extended operations. We start by defining the first four layers of Figure 1.A independently, using Proposition 4.14 (with  $T = \text{id}$ ). The relevant arities are

$$\Gamma^{\text{plus}}(F, G)(X) = (F(X)_\mathbf{m} \times G(X)_\mathbf{m}) \cdot \mathbf{y}_\mathbf{m} \quad \Gamma^{\text{app}}(F, G)(X) = (F(X)_\mathbf{m} \times G(X)_\mathbf{m}) \cdot \mathbf{y}_\mathbf{m}$$

$$\Gamma^{\text{abs}}(F, G)(X) = F(X+1)_\mathbf{m} \cdot \mathbf{y}_\mathbf{m} \quad \Gamma^{\text{lapp}_0}(F, G)(X) = (G(X)_\mathbf{s} \times F(X)_\mathbf{m}) \cdot \mathbf{y}_\mathbf{m},$$

and the equations in the table may be read as defining the desired simple structural laws

$$\Gamma^i(\Sigma(F), G) \rightarrow S(\Gamma^i(F, S(G)) + F + G),$$

for  $i \in \{\text{plus}, \text{app}, \text{abs}, \text{lapp}_0\}$ . We get a distributive law

$$\delta_4: T_4 S \rightarrow S T_4,$$

where  $T_4 = (\sum_i \Gamma^i_\mathbf{s})^*$ .

**Remark 14.** The operation in the fourth layer is defined by induction on the second argument, so the main (inductive) argument for  $\Gamma^{\text{lapp}_0}$  goes to the right of the product.

For the last layer in Figure 1.A, we define an incremental structural law over  $\delta_4$ , with arity

$$\Gamma^{\text{lapp}}(F, G)(X) = (F(X)_\mathbf{m} \times G(X)_\mathbf{m}) \cdot \mathbf{y}_\mathbf{m}.$$

Again the equations of the last layer may be read as defining this incremental law. By Theorem 4.2, we get a distributive law  $\delta_5: T_5 S \rightarrow S T_5$ , where  $T_5 = T_4 \oplus T_{\text{lapp}}$ , with  $T_{\text{lapp}} = (\Gamma^{\text{lapp}}_\mathbf{s})^*$ .

We now define capture-avoiding substitution. The arity is reminiscent of substitution monoidal structures, so we write it as a tensor product. Because the result of a substitution is a multiterm, we readily put  $(F \otimes G)(X)_\mathbf{s} = \emptyset$ . Then, we define

$$(F \otimes G)(X)_\mathbf{m} = F(G(X)_\mathbf{m})_\mathbf{s} + F(G(X)_\mathbf{m})_\mathbf{m},$$

reflecting the fact that we substitute multiterms for variables in both simple and multiterms.

The two layers of Figure 1.B may be read as defining the two components of the desired incremental structural law

$$\Sigma(F) \otimes G \rightarrow S T_5 (F \otimes S T_5 G + F + G),$$

which yields by Theorem 4.2 a distributive law  $\delta_6: T_6S \rightarrow ST_6$ , where  $T_6 = T_5 \oplus T_{subst}$ , with  $T_{subst} = (- \otimes S -)^*$ .

Finally, we define partial differentiation. The arity is again empty at  $s$ , with

$$\Gamma^{diff}(F, G)(X)_m = F(X+1)_s \times G(X)_m + F(X+1)_m \times G(X)_m.$$

The additional variable in the first argument models the distinguished variable  $x$  along which we differentiate. Again the equations may be read as defining an incremental structural law and we get a distributive law  $\delta_7: T_7S \rightarrow ST_7$ , where  $T_7 = T_6 \oplus T_{diff}$ , with  $T_{diff} = (\Gamma_S^{diff})^*$ .

One could be content with this, prove the desired commutation lemmas [Vaux 2007, §6.1.4], and then show that everything is compatible with Equations (6), all by hand. But of course, it would be better to derive both results automatically. In the next section, we will explain how to derive commutation lemmas (though on a simpler example), leaving compatibility with equations for further work.

#### 4.6 Application: capture-avoiding substitution, De Bruijn style

Most applications of the paper take place in the setting of so-called presheaf models, i.e., mild generalisations of the category of finitary functors  $\mathbf{Set} \rightarrow \mathbf{Set}$  [Fiore et al. 1999]. Our framework, however, applies just as well in other settings. To illustrate this, in this subsection, we transpose the example of capture-avoiding substitution to the De Bruijn-style setting of [Hirschowitz et al. 2022b].

To summarise the idea: just as the presheaf-based approach equips the nested datatypes representation [Bird and Paterson 1999] with initial algebra semantics, the setting of [Hirschowitz et al. 2022b] does the same for De Bruijn representation [De Bruijn 1972]. In the presheaf-based approach, terms are indexed by sets  $n$  of potential free variables, and by convention the bound variable is always the greatest one, typically  $x_{n+1}$  for  $\lambda_n: X(n+1) \rightarrow X(n)$ . Indexing is thus made necessary by the binding convention. In De Bruijn representation, the binding convention is the opposite: the bound variable is always 0. This makes indexing unnecessary, but some operations become less intuitive (to many, at least [Berghofer and Urban 2007]).

In order to specify capture-avoiding substitution in the De Bruijn setting, we will need several layers. Indeed, the presheaf-based approach features not only indexing, but also built-in renaming, which is not the case in the De Bruijn setting. We thus need to define a first layer for renaming, and then a second one for substitution.

We take  $\mathbf{Set}$  as ambient category, and the basic syntax is specified by the endofunctor

$$\Sigma(X) = \mathbb{N} + X^2 + X.$$

For the first layer, letting  $S = \Sigma^*$  again, the auxiliary bifunctor is

$$\Gamma(X, Y) = X \times \mathbb{N}^{\mathbb{N}},$$

and renaming is specified by the following simple structural law (omitting  $\eta^S$  for readability):

$$\begin{aligned} d_{X,Y}: \Sigma(X) \times \mathbb{N}^{\mathbb{N}} &\rightarrow S(X \times \mathbb{N}^{\mathbb{N}} + X + Y) \\ (x_i, \rho) &\mapsto x_{\rho(i)} \\ (\lambda(e), \rho) &\mapsto \lambda(in_1(e, 1 + \rho)) \\ (e f, \rho) &\mapsto in_1(e, \rho) in_1(f, \rho) \end{aligned}$$

where  $1 + \rho: \mathbb{N} \rightarrow \mathbb{N}$  is defined by

$$\begin{aligned} (1 + \rho)(0) &= 0 \\ (1 + \rho)(1 + n) &= 1 + \rho(n). \end{aligned}$$

Letting  $T = \Gamma_S^*$ , we get a distributive law  $\delta: TS \rightarrow ST$  by Theorem 3.5.

**Notation 4.** We denote by  $e[\rho]$  the renaming operation of  $T$ .

We then specify substitution by taking as auxiliary bifunctor  $\Theta(X, Y) = X \times Y^{\mathbb{N}}$ , with incremental structural law defined by

$$\begin{aligned} \Sigma(X) \times Y^{\mathbb{N}} &\rightarrow ST(X \times (STY)^{\mathbb{N}} + X + Y) \\ (x_i, \sigma) &\mapsto in_3(\sigma(i)) \\ (\lambda(e), \sigma) &\mapsto \lambda(in_1(e, \uparrow \sigma)) \\ (e f, \sigma) &\mapsto in_1(e, \sigma) in_1(f, \sigma), \end{aligned}$$

where  $\uparrow \sigma: \mathbb{N} \rightarrow STY$  is defined by

$$\begin{aligned} \uparrow \sigma: \mathbb{N} &\rightarrow STY \\ 0 &\mapsto x_0 \\ n + 1 &\mapsto \sigma(n)[\uparrow], \end{aligned}$$

letting  $\uparrow: \mathbb{N} \rightarrow \mathbb{N}$  denote the successor map.

## 5 BENIGN EQUATIONS

In this section, we examine what we call “benign equations”, i.e., equations that are automatically satisfied by the initial model. The initial model is thus initial in a potentially smaller category.

We start in §5.1 by presenting a simple example to motivate this investigation. In §5.2, along with recalling Fiore and Hur’s [2009] **equational systems** and some basic facts about them, we then introduce a mathematical definition of a system of benign equations, called a **benign** equational system. In §5.3, we present a notion of signature for benign equational systems, called **structural equational systems**, which we apply in §5.4 to prove associativity of capture-avoiding substitution.

### 5.1 On a simple example

In the case of addition, as defined by structural recursion in §3.1, an example of benign equation is associativity:

$$(x + y) + z = x + (y + z). \quad (7)$$

This is typically proven by induction on the first variable  $x$ . We explain how to derive this result from the results of §3.1 and §4, which leads to structural equational systems.

We start by briefly sketching the idea. By the construction of §4, we define a new, auxiliary ternary operation  $op(x, y, z)$  by the structurally recursive equations

$$op(0, y, z) = y + z \quad (8)$$

$$op(s(x), y, z) = s(op(x, y, z)). \quad (9)$$

By Theorem 4.2, the syntax,  $S\emptyset$ , admits a unique such operation. But on the other hand, one easily proves that taking  $op(x, y, z) = x + (y + z)$  or  $op(x, y, z) = (x + y) + z$  yields two such operations. Indeed, we have

$$s(x) + (y + z) = s(x + (y + z)) \quad 0 + (y + z) = y + z$$

and

$$(s(x) + y) + z = s(x + y) + z = s((x + y) + z) \quad (0 + y) + z = y + z$$

by the defining equations of addition. By uniqueness, both definitions of  $op$  must thus coincide, which proves (7).

More formally, we start from the distributive law  $\delta: TS \rightarrow ST$  of §3.1. We introduce a bifunctor  $\Psi: \mathbf{Set}^2 \rightarrow \mathbf{Set}$  mapping  $(X, Y)$  to  $X \times Y \times Y$ , for the arity of the ternary operation. Again,  $X$

corresponds to the decreasing argument in the recursive definition of  $op$ , while  $Y$  accounts for other arguments. As in §4, the recursive definition is modelled by an incremental structural law

$$\begin{aligned} d_{X,Y}: \Psi(\Sigma X, Y) &\rightarrow ST(\Psi(X, STY) + X + Y) \\ (0, y_1, y_2) &\mapsto in_3(y_1) + in_3(y_2) \\ (s(x), y_1, y_2) &\mapsto s(in_1(x, y_1, y_2)). \end{aligned}$$

Equation (7) induces a pair

$$L_X, R_X: \Psi(X, X) \rightarrow STX$$

of natural transformations, respectively mapping any triple  $(x, y, z) \in X^3$  to  $(x+y)+z$  and  $x+(y+z)$ , and we want to show that the algebra structure  $\mu_0^{ST}: STST\emptyset \rightarrow ST\emptyset (\cong S\emptyset)$  of the initial  $ST$ -algebra coequalises  $L_{ST\emptyset}$  and  $R_{ST\emptyset}$ , as in

$$\Psi(ST\emptyset, ST\emptyset) \begin{array}{c} \xrightarrow{L_{ST\emptyset}} \\ \xrightarrow{R_{ST\emptyset}} \end{array} STST\emptyset \xrightarrow{\mu_0^{ST}} ST\emptyset.$$

By Theorem 4.13, models of the incremental structural law  $d$  are  $ST$ -algebras equipped with an additional ternary operation making the diagram (d2) commute (with  $S := ST$  and  $\Gamma := \Psi$ ), and furthermore, by Theorem 4.10,  $ST\emptyset$  is an initial model. By initiality, we thus merely need to show that both induced composites  $\Psi(ST\emptyset, ST\emptyset) \rightarrow ST\emptyset$  make (d2) commute.

In the next subsections, we propose an abstract version of this idea, with a simple sufficient condition for it to work.

## 5.2 Equations

In this subsection, loosely following [Fiore and Hur 2009; Hirschowitz et al. 2022a], we introduce an abstract notion of equational system, and define what it means for such an equational system to be benign.

*Definition 5.1.* Given a finitary monad  $T$  on a locally finitely presentable category  $\mathbf{C}$ , an **equational system** consists of a finitary monad  $G$  on  $\mathbf{C}$ , together with two monad morphisms  $G \rightarrow T$ .

*Example 5.2.* Taking  $\mathbf{C} = \mathbf{Set}$ , we model associativity of a binary operation by taking  $T$  and  $G$  to be the free monads on the endofunctors  $\Sigma(X) = X^2$  and  $\Theta(X) = X^3$  on sets, respectively. The monad morphisms  $L, R: G \rightarrow T$  are induced by universal property of  $G = \Theta^*$  from the natural transformations  $L^0, R^0: \Theta \rightarrow T$  defined by

$$\begin{aligned} L_X^0(x_1, x_2, x_3) &= x_1 + (x_2 + x_3) \\ R_X^0(x_1, x_2, x_3) &= (x_1 + x_2) + x_3. \end{aligned}$$

*Definition 5.3.* The **quotient**  $E^*$  of a finitary monad  $T$  by an equational system  $E = (G, L, R)$  is the coequaliser

$$G \begin{array}{c} \xrightarrow{L} \\ \xrightarrow{R} \end{array} T \xrightarrow{q} E^*$$

of  $L, R$  in  $\mathbf{Mnd}_f(\mathbf{C})$ .

**Remark 15.** The coequaliser exists because the category of finitary monads on a locally finitely presentable category is itself locally finitely presentable [Lack 1997], hence in particular cocomplete.

**Remark 16.** The coequaliser is also a coequaliser in the category  $\mathbf{Mnd}(\mathbf{C})$  of (not necessarily finitary) monads on  $\mathbf{C}$ , since colimits are preserved by the embedding  $\mathbf{Mnd}_f(\mathbf{C}) \rightarrow \mathbf{Mnd}(\mathbf{C})$  by Blackwell [1976, Proposition 5.6].

*Definition 5.4.* An equational system is **benign** when the universal coequalising morphism  $T \rightarrow E^*$  is admissible.

Although coequalisers of finitary monads may not be intuitive to all readers, the monad  $E^*$  admits the following nice characterisation by its algebras.

*Definition 5.5.* Given a finitary monad  $T$  and an equational system  $E = (G, L, R)$  on it, a  $T$ -algebra  $a: TX \rightarrow X$  **satisfies**  $E$  iff  $a$  coequalises  $L_X$  and  $R_X$ , i.e.,  $a \circ L_X = a \circ R_X$ .

Such  $T$ -algebras are called  $E$ -**algebras**, and we denote by  $E$ -**alg** the full subcategory of  $T$ -**Alg** spanned by them.

**Remark 17.** *Equivalently,  $E$ -**alg** is the equaliser in  $\mathbf{CAT}$  of the induced functors  $T$ -**Alg**  $\rightarrow$   $G$ -**Alg**, which a priori might differ from the equaliser in  $\mathbf{Monadic}_f/\mathbf{C}$ , the full subcategory of  $\mathbf{Monadic}/\mathbf{C}$  spanned by finitary monadic functors. They in fact coincide, by the following Proposition.*

**PROPOSITION 5.6.** *For any finitary monad  $T$  on a locally finitely presentable category  $\mathbf{C}$  and equational system  $E = (G, L, R)$  on it, we have an isomorphism  $E$ -**alg**  $\cong$   $E^*$ -**Alg** over  $\mathbf{C}$ . Otherwise said, the forgetful functor  $E$ -**alg**  $\rightarrow$   $\mathbf{C}$  is finitary and monadic, and the associated monad is isomorphic to  $E^*$ .*

**PROOF.** By Remark 16,  $E^*$  is a coequaliser of  $L$  and  $R$ , as monad morphisms. By Kelly [1980, Proposition 26.3], its category of algebras is thus computed as the equaliser in  $\mathbf{CAT}/\mathbf{C}$  of the functors  $T$ -**Alg**  $\rightarrow$   $G$ -**Alg**  $\rightarrow$   $\Theta$ -**alg**, as claimed.  $\square$

We may derive from this the following useful characterisation of benign equational systems.

**PROPOSITION 5.7.** *An equational system is benign iff the initial  $T$ -algebra satisfies it, i.e.,  $\mu_0^T \circ L_{T\emptyset} = \mu_0^T \circ R_{T\emptyset}$ .*

**PROOF.** If the initial algebra satisfies an equational system  $E$ , then it is *a fortiori* initial in  $E$ -**alg**, hence in  $E^*$ -**Alg** by Proposition 5.6. The forgetful functor  $E^*$ -**Alg**  $\rightarrow$   $T$ -**Alg** thus preserves the initial object, and we conclude by Proposition 2.4.

Conversely, if an equational system  $E$  is benign, then, by Proposition 2.4, the initial  $T$ -algebra admits a unique  $E^*$ -algebra structure  $e: E^*T\emptyset \rightarrow T\emptyset$  making the following diagram commute

$$\begin{array}{ccc} TT\emptyset & \xrightarrow{q_{T\emptyset}} & E^*T\emptyset \\ & \searrow \mu_0^T & \swarrow e \\ & T\emptyset & \end{array}$$

and  $e$  makes  $T\emptyset$  into an initial  $E^*$ -algebra. But  $q_{T\emptyset}$  coequalises  $L_{T\emptyset}$  and  $R_{T\emptyset}$ , hence so does  $\mu_0^T$ , as desired.  $\square$

Let us conclude this subsection with the following observation on combining equations.

*Definition 5.8.* For any family  $(E_i)_{i \in I}$  of equational systems on a given finitary monad  $T$  on a locally finitely presentable category  $\mathbf{C}$ , with  $E_i = (G_i, L_i, R_i)$  for all  $i \in I$ , let  $\sum_i E_i$  denote the equational system defined by the cotuplings

$$[L_i]_{i \in I}: \bigoplus_{i \in I} G_i \rightarrow T \quad \text{and} \quad [R_i]_{i \in I}: \bigoplus_{i \in I} G_i \rightarrow T.$$

**PROPOSITION 5.9.** *For any family  $(E_i)_{i \in I}$  of equational systems on a given finitary monad  $T$  on a locally finitely presentable category  $\mathbf{C}$ ,  $\sum_i E_i$  is benign iff each  $E_i$  is.*

**PROOF.** By Proposition 5.7 and universal property of coproduct.  $\square$

### 5.3 Structural equational systems

In this subsection, we introduce a notion of signature for benign equational systems.

We fix a free monad  $S = \Sigma^*$  on a locally finitely presentable category  $\mathbf{C}$ , with  $\Sigma$  (hence  $S$ ) finitary.

*Definition 5.10.*

- A **structural interpretation** of an incremental structural law

$$d_{X,Y} : \Theta(\Sigma X, Y) \rightarrow ST(\Theta(X, STY) + X + Y)$$

over some given distributive law  $\delta : TS \rightarrow ST$  is a natural transformation  $K_X : \Theta(X, X) \rightarrow STX$  making the coherence diagram of Figure 2 commute.

$$\begin{array}{ccc}
 \Theta(\Sigma X, X) & \xrightarrow{d_{X,X}} & ST(\Theta(X, STX) + X + X) \\
 \Theta(\eta_{\Sigma, X}, \eta_X^S) \downarrow & & \downarrow ST(\Theta(\eta_X^{ST}, STX) + [X, X]) \\
 \Theta(SX, SX) & & ST(\Theta(STX, STX) + X) \\
 K_{SX} \downarrow & & \downarrow ST(K_{STX} + X) \\
 STSX & & ST(STSTX + X) \\
 S\delta_X \downarrow & & \downarrow ST[\mu_X^{ST}, \eta_X^{ST}] \\
 SSTX & & ST(STX) \\
 \mu_{TX}^S \swarrow & & \swarrow \mu_X^{ST} \\
 & STX & 
 \end{array}$$

Fig. 2. Coherence diagram for structural interpretations

- A **structural equational system** over a distributive law  $\delta : TS \rightarrow ST$  consists of
  - an incremental structural law  $(\Theta, d)$  over  $\delta$ , together with
  - a pair of structural interpretations  $L, R : \Theta\Delta \rightarrow ST$  of  $d$ .

We now associate an equational system to any structural equational system, and prove that it is benign.

*Definition 5.11.* For any structural interpretation  $K : \Theta\Delta \rightarrow ST$  of

$$d_{X,Y} : \Theta(\Sigma X, Y) \rightarrow ST(\Theta(X, STY) + X + Y)$$

over  $\delta : TS \rightarrow ST$ , let  $\tilde{K} : \Theta_S^* \rightarrow ST$  denote the monad morphism induced by universal property of  $\Theta_S^*$  from the composite

$$\Theta_{SX} X \xrightarrow{\Theta_{SX} \eta_X^S} \Theta_{SX} SX \xrightarrow{K_{SX}} STSX \xrightarrow{S\delta_X} SSTX \xrightarrow{\mu_{TX}^S} STX.$$

For any structural equational system  $E = (d, L, R)$  over  $\delta$ , the equational system  $\tilde{E}$  induced by  $E$  is the pair  $\tilde{L}, \tilde{R} : \Theta_S^* \rightarrow ST$ .

**Notation 5.** We often conflate  $E$  and the associated equational system  $\tilde{E}$ . In particular, recalling Definition 5.5 and Proposition 5.6, we speak of  $E$ -algebras, which form a category  $E\text{-alg} \cong E^*\text{-Alg}$ .

**THEOREM 5.12.** Consider any monad distributive law  $\delta : TS \rightarrow ST$  in a locally finitely presentable category  $\mathbf{C}$  such that  $T$  is constant-free. Then for any structural equational system  $E$  over  $\delta$ , the quotient morphism  $ST \rightarrow \tilde{E}^*$  is admissible.

PROOF SKETCH (SEE §E FOR MORE DETAIL). Let

$$d_{X,Y} : \Theta(\Sigma X, Y) \rightarrow ST(\Theta(X, STY) + X + Y)$$

denote the given incremental structural law, and  $L$  and  $R$  denote the two structural interpretations.

We first prove that each any structural interpretation  $K$  induces  $\Theta\Delta$ -algebra structure on any  $ST$ -algebra  $X$ , given by

$$\Theta(X, X) \xrightarrow{K_X} STX \rightarrow X,$$

and furthermore that this  $\Theta\Delta$ -algebra structure satisfies (d2), hence by Theorem 4.13 makes  $X$  into an  $S(T \oplus T')$ -algebra.

The given structural interpretations  $L$  and  $R$  thus induce two extensions of the  $ST$ -algebra structure of  $S\emptyset \cong ST\emptyset$  to  $S(T \oplus T')$ -algebra structure. But by Theorem 4.2 and Proposition 2.4,  $S\emptyset$  has a unique compatible  $\Theta\Delta$ -algebra structure making it into a  $S(T \oplus T')$ -algebra structure, so both algebra structures derived from  $L$  and  $R$  must agree with the canonical one. In particular the diagram

$$\Theta(S\emptyset, S\emptyset) \begin{array}{c} \xrightarrow{L_{S\emptyset}} \\ \xrightarrow{R_{S\emptyset}} \end{array} STS\emptyset \longrightarrow S\emptyset$$

commutes, hence  $S\emptyset$  satisfies the induced equational system, and is thus initial in  $S(T \oplus T')$  - **Alg**.  $\square$

Let us conclude this subsection with the following direct consequence of Theorem 5.12 and Proposition 5.9.

**COROLLARY 5.13.** *Consider any monad distributive law  $\delta: TS \rightarrow ST$  in a locally finitely presentable category  $\mathcal{C}$  such that  $T$  is constant-free. Then for any family  $(E_i)_{i \in I}$  of structural equational systems over  $\delta$ , the quotient morphism  $ST \rightarrow (\sum_i \tilde{E}_i)^*$  is admissible.*

#### 5.4 Application: associativity of substitution

In this section, we continue the development of §3.5. There, we specified the syntax of pure  $\lambda$ -calculus by an endofunctor  $\Sigma$  on  $[\mathbf{Set}, \mathbf{Set}]_f$ , and substitution by a simple structural law

$$d_{X,Y} : \Gamma(\Sigma X, Y) \rightarrow S(\Gamma_{SY}(X) + X + Y),$$

where  $\Gamma(X, Y) = X \otimes Y$ , thus generating an admissible morphism  $S \rightarrow ST$ , where  $S = \Sigma^*$  and  $T = \Gamma_S^*$ .

We now want to show that the  $\Gamma\Delta$ -algebra structure of the syntactic model  $S\emptyset$ , a.k.a. capture-avoiding substitution, is associative. Again, this will be subsumed by §5.5.

We define a structural equational system with the following components:

- the arity functor is  $\Theta(X, Y) = (X \otimes Y) \otimes Y$ ;
- the incremental structural law is defined by

$$\begin{aligned} d'_{X,Y}(x, \sigma, \theta) &= \sigma(x)[\theta] \\ d'_{X,Y}(e f, \sigma, \theta) &= (e, \sigma, \theta)(f, \sigma, \theta) \\ d'_{X,Y}(\lambda(e), \sigma, \theta) &= \lambda(e, \sigma^\dagger, \theta^\dagger), \end{aligned}$$

where

- $\sigma : p \rightarrow ST(Y)(q)$ ,
- $\theta : q \rightarrow ST(Y)(n)$ ,
- $-_1[-_2]$  denotes the formal (= explicit) substitution operation of  $T$ , and
- we omit coproduct injections and  $\eta^{ST}$  for readability;

- the structural interpretations are defined by

$$L_X(e, \sigma, \theta) = e[\sigma][\theta] \text{ and } R_X(e, \sigma, \theta) = e[\sigma[\theta]],$$

where by definition  $\sigma[\theta](x_i) := \sigma(x_i)[\theta]$  (writing  $x_i$  for the  $i$ th element of  $p$  to emphasise that it is thought of as a variable).

Checking the coherence condition of Figure 2 essentially amounts to checking each case of the usual induction, separately. The most interesting case is that of abstraction, with the right-hand side  $R$ :

- $d$  maps any triple  $(\lambda(e), \sigma, \theta)$  to  $\lambda(e, \sigma^\uparrow, \theta^\uparrow)$ , which the right-hand composite then maps to  $\lambda(e[\sigma^\uparrow[\theta^\uparrow]])$ , while
- the left-hand composite maps the triple to  $\lambda(e[\sigma[\theta]^\uparrow])$ .

We thus need to prove  $\sigma^\uparrow[\theta^\uparrow] = \sigma[\theta]^\uparrow$ :

- on  $x_{p+1}$ , we directly have  $\sigma[\theta]^\uparrow(x_{p+1}) = x_{n+1}$ , and, slightly less directly,

$$\sigma^\uparrow[\theta^\uparrow](x_{p+1}) = \sigma^\uparrow(x_{p+1})[\theta^\uparrow] = x_{q+1}[\theta^\uparrow] = x_{n+1};$$

- on  $x_i$  for  $i \in p$ , we have  $\sigma[\theta]^\uparrow(x_i) = w_n \cdot (\sigma(x_i)[\theta]) = \sigma(x_i)[STX(w_n) \circ \theta]$ , where  $w_n: n \hookrightarrow n+1$  denotes the inclusion, while

$$\begin{aligned} \sigma^\uparrow[\theta^\uparrow](x_i) &= \sigma^\uparrow(x_i)[\theta^\uparrow] \\ &= (w_q \cdot \sigma(x_i))[\theta^\uparrow] \\ &= \sigma(x_i)[\theta^\uparrow \circ w_q]. \end{aligned}$$

But the following square commutes by definition of  $\theta^\uparrow$ , hence the result.

$$\begin{array}{ccc} q & \xrightarrow{w_q} & q+1 \\ \theta \downarrow & & \downarrow \theta^\uparrow \\ STX(n) & \xrightarrow{STX(w_n)} & STX(n+1) \end{array}$$

Theorem 5.12 then tells us that the usual substitution lemma is satisfied in the syntax  $S(\emptyset)$ .

## 5.5 Embedding presheaf-based models

In this section, we show how the general framework of pointed strong endofunctors [Fiore et al. 1999; Fiore 2008] embeds into ours. More precisely, for any pointed strong endofunctor  $\Sigma$  on a monoidal, locally finitely presentable category  $(\mathbb{C}, \otimes, I, \alpha, \lambda, \rho)$  satisfying standard additional axioms (see Definition 5.16 below):

- We define a simple structural law, whose initial algebra  $S\emptyset = (I + \Sigma)^*\emptyset$  is the desired syntax, and whose category of algebras is a relaxed variant of Fiore et al.'s;
- We then define two structural equational systems  $E_1$  and  $E_2$ , whose joint algebras in the sense of Corollary 5.13 are precisely those of Fiore et al.'s.

We thus recover the admissible morphism

$$S \rightarrow ST \rightarrow (\tilde{E}_1 + \tilde{E}_2)^\star \cong \Sigma^\otimes$$

of Example 2.5.

**Remark 18.** *The motivation for this subsection is one of connecting to other people's work. In applications, it will probably be easier to directly define the desired structural laws and equational systems.*

We start by recalling the notion of pointed strong endofunctor, and its associated category of models.

*Definition 5.14.* A **pointed strength** on an endofunctor  $\Sigma: \mathbf{C} \rightarrow \mathbf{C}$  on a monoidal category  $(\mathbf{C}, \otimes, I, \alpha, \lambda, \rho)$  is a family of morphisms  $st_{C,(D,v)}: \Sigma(C) \otimes D \rightarrow \Sigma(C \otimes D)$ , natural in  $C \in \mathbf{C}$  and  $(D, v: I \rightarrow D) \in I/\mathbf{C}$ , the coslice category below  $I$ , making the following diagrams commute,

$$\begin{array}{ccc}
& \Sigma(A) & \\
\rho_{\Sigma(A)} \swarrow & & \searrow \Sigma(\rho_A) \\
\Sigma(A) \otimes I & \xrightarrow{st_{A,(I,\text{id})}} & \Sigma(A \otimes I) \\
(\Sigma(A) \otimes X) \otimes Y & \xrightarrow{st_{A,(X,v_X)} \otimes Y} \Sigma(A \otimes X) \otimes Y & \xrightarrow{st_{A \otimes X,(Y,v_Y)}} \Sigma((A \otimes X) \otimes Y) \\
\alpha_{\Sigma(A),X,Y} \downarrow & & \downarrow \Sigma(\alpha_{A,X,Y}) \\
\Sigma(A) \otimes (X \otimes Y) & \xrightarrow{st_{A,(X \otimes Y, v_{X \otimes Y})}} & \Sigma(A \otimes (X \otimes Y))
\end{array}$$

where  $v_X: I \rightarrow X$  and  $v_Y: I \rightarrow Y$  are the given points, and  $v_{X \otimes Y}$  denotes the composite

$$I \xrightarrow{\rho_I^{-1}} I \otimes I \xrightarrow{v_X \otimes v_Y} X \otimes Y.$$

*Definition 5.15.* For any pointed strong endofunctor  $\Sigma$  on  $\mathbf{C}$ , a  $\Sigma$ -**monoid** is an object  $X$  equipped with  $\Sigma$ -algebra and monoid structure, say  $a: \Sigma(X) \rightarrow X$ ,  $s: X \otimes X \rightarrow X$ , and  $v: I \rightarrow X$ , such that the following pentagon commutes.

$$\begin{array}{ccc}
\Sigma(X) \otimes X & \xrightarrow{st_{X,(X,v)}} \Sigma(X \otimes X) & \xrightarrow{\Sigma(s)} \Sigma(X) \\
a \otimes X \downarrow & & \downarrow a \\
X \otimes X & \xrightarrow{s} & X
\end{array} \tag{10}$$

A morphism of  $\Sigma$ -monoids is a morphism in  $\mathbf{C}$  which is a morphism both of  $\Sigma$ -algebras and of monoids. We let  $\Sigma$ -**Mon** denote the category of  $\Sigma$ -monoids and morphisms between them.

Let us now show how any pointed strong endofunctor  $\Sigma$  gives rise to a simple structural law, and how the coherence laws of  $\Sigma$ -monoids may be enforced by benign equations.

Following [Fiore et al. 1999; Fiore 2008], we assume that our category  $\mathbf{C}$  is leftist, in the following sense.

*Definition 5.16.* A monoidal category  $(\mathbf{C}, \otimes, I, \alpha, \lambda, \rho)$  is **leftist** iff  $\otimes$  is cocontinuous in its first argument, and finitary in its second argument.

*Definition 5.17.* The simple structural law associated to any pointed strong endofunctor  $(\Sigma, st)$  is defined as follows.

- We take as basic functor  $\Sigma^+(X) = I + \Sigma(X)$ , and
- as auxiliary bifunctor  $\Gamma(X, Y) = X \otimes Y$ .
- We then take as simple structural law  $d_{X,Y}$  the composite

$$\begin{array}{ccc}
\Sigma^+(X) \otimes SY & \xrightarrow{\lambda_{SY} + st_{X,(SY,v_Y)}} & SY + \Sigma(X \otimes SY) \\
\Sigma^+(X) \otimes \eta_Y^S \uparrow & & \downarrow [Sin_3, S(in_1) \circ \eta_{\Sigma^+, X \otimes SY} \circ in_2] \\
\Sigma^+(X) \otimes Y & & S(X \otimes SY + X + Y);
\end{array}$$

where  $S = (\Sigma^+)^*$  and  $v$  denotes the composite  $I \rightarrow \Sigma^+ \rightarrow S$ .

By Theorem 3.5, the initial  $\Sigma^+$ -algebra has a unique model structure, which makes it initial, and furthermore, by Theorem 4.13, models are  $\Sigma$ -algebras  $a: \Sigma X \rightarrow X$ , equipped with morphisms  $v: I \rightarrow X$  and  $s: X \otimes X \rightarrow X$  making the pentagon

$$\begin{array}{ccc}
 \Sigma^+(X) \otimes X & \xrightarrow{d_{X,X}} & S(X \otimes SX + X + X) \\
 \downarrow [v,a] \otimes X & & \downarrow S(X \otimes \overline{[v,a]} + X + X) \\
 X \otimes X & & S(X \otimes X + X + X) \\
 & & \downarrow S[s, X, X] \\
 & & SX \\
 & \swarrow s & \nwarrow \overline{[v,a]} \\
 & X &
 \end{array} \tag{11}$$

commute, where  $\overline{[v,a]}$  is induced by universal property of  $S(X)$ .

PROPOSITION 5.18. *Commutation of (11) is equivalent to joint commutation of the pentagon (10) and the diagram below.*

$$\begin{array}{ccc}
 I \otimes X & \xrightarrow{v \otimes X} & X \otimes X \\
 \searrow \lambda_X & & \swarrow s \\
 & X &
 \end{array} \tag{12}$$

PROOF. The domain of (11) is a coproduct, and we claim that the restriction to each term yields one of the given diagrams. For the first term, we get the following diagram.

$$\begin{array}{ccccccc}
 I \otimes X & \xrightarrow{I \otimes \eta_X^S} & I \otimes SX & \xrightarrow{\lambda_{SX}} & SX & \xrightarrow{Sin_3} & S(X \otimes SX + X + X) \\
 \downarrow v \otimes X & \searrow \lambda_X & \downarrow \eta_X^S & \nearrow \eta_X^S & \downarrow \eta_X^S & & \downarrow S(X \otimes \overline{[v,a]} + X + X) \\
 X \otimes X & & X & & X & & S(X \otimes X + X + X) \\
 & & & & & & \downarrow S[s, X, X] \\
 & & & & & & SX \\
 & & & & & & \nwarrow \overline{[v,a]} \\
 & & & & & & X
 \end{array}$$

For the second term, we verify that both pentagons are equivalent by chasing the following diagram.

$$\begin{array}{ccccccc}
 \Sigma(X) \otimes X & \xrightarrow{\Sigma(X) \otimes \eta_X^S} & \Sigma(X) \otimes S(X) & \xrightarrow{st_{X,(X,v_X)}} & \Sigma(X \otimes SX) & \xrightarrow{in_2} & \Sigma^+(X \otimes SX) & \xrightarrow{\eta_{\Sigma^+, in_1}} & S(X \otimes SX + X + X) \\
 \downarrow a \otimes X & \searrow \Sigma(X) \otimes \overline{[v,a]} & \downarrow \Sigma(X) \otimes \overline{[v,a]} & \searrow \Sigma(X) \otimes \overline{[v,a]} & \downarrow \Sigma(s) & \searrow \Sigma^+(s) & \downarrow \Sigma^+(s) & \searrow S[s, X, X] & \downarrow S(X \otimes \overline{[v,a]} + X + X) \\
 X \otimes X & & \Sigma(X) \otimes X & \xrightarrow{st_{X,(X,v)}} & \Sigma(X \otimes X) & \xrightarrow{in_2} & \Sigma^+(X \otimes X) & \xrightarrow{\eta_{\Sigma^+, in_1}} & S(X \otimes X + X + X) \\
 & & & & & & & & \downarrow S[s, X, X] \\
 & & & & & & & & SX \\
 & & & & & & & & \nwarrow \overline{[v,a]} \\
 & & & & & & & & X
 \end{array}$$

□

However,  $s$  and  $v$  are not yet required to satisfy the remaining monoid equations. In order to enforce them, relying on Corollary 5.13, we introduce two structural equational systems, one for each equation. First, we introduce some notation.

*Definition 5.19.* For any object  $Z$ , we define  $j_Z: Z \otimes Z \rightarrow TZ$  to be the composite

$$Z \otimes Z \xrightarrow{Z \otimes \eta_Z^S} Z \otimes SZ = \Gamma_{SZ} Z \xrightarrow{\eta_{\Gamma_S \Delta, Z}} TZ.$$

*Definition 5.20.* For our first structural equational system  $E_1$ , we take

- as auxiliary bifunctor  $\Theta_1(X, Y) = (X \otimes Y) \otimes Y$ ;
- as incremental structural law, say  $d_1$ , we take the left-hand composite of Figure 3;

$$\begin{array}{ccc}
\Sigma^+ X \otimes Y \otimes Y & & \Sigma^+ X \otimes I \\
\parallel & & \parallel \\
I \otimes Y \otimes Y + \Sigma X \otimes Y \otimes Y & & I \otimes I + \Sigma X \otimes I \\
\downarrow \lambda_Y \otimes Y + X \otimes \eta_Y^S \otimes \eta_Y^S & & \downarrow \lambda_I + st_{X, (I, id_I)} \\
Y \otimes Y + \Sigma X \otimes SY \otimes SY & & I + \Sigma(X \otimes I) \\
\downarrow j_Y + st_{X, (SY, v_Y)} \otimes SY & & \parallel \\
TY + \Sigma(X \otimes SY) \otimes SY & & \Sigma^+(X \otimes I) \\
\downarrow TY + st_{X \otimes SY, (SY, v_Y)} & & \downarrow \eta_{\Sigma^+} \\
TY + \Sigma(X \otimes SY \otimes SY) & & S(X \otimes I) \\
\downarrow TY + in_2 & & \downarrow S\eta_{X \otimes I}^T \\
TY + \Sigma^+(X \otimes SY \otimes SY) & & ST(X \otimes I) \\
\downarrow \eta_{TY}^S + \eta_{\Sigma^+} \eta^T (X \otimes S\eta_Y^T \otimes S\eta_Y^T) & & \downarrow STin_1 \\
STY + ST(X \otimes STY \otimes STY) & & ST(X \otimes I + X + Y) \\
\downarrow [STin_3, STin_1] & & \\
ST(X \otimes STY \otimes STY + X + Y) & & 
\end{array}$$

Fig. 3. Two incremental structural laws

- as first structural interpretation, say  $L_1$ , at any  $X$ , we take

$$X \otimes X \otimes X \xrightarrow{j_X \otimes \eta_X^T} TX \otimes TX \xrightarrow{j_{TX}} TTX \xrightarrow{\eta^S \mu_X^T} STX;$$

- as second structural interpretation, say  $R_1$ , at any  $X$ , we take

$$X \otimes X \otimes X \xrightarrow{\alpha_{X, X, X}} X \otimes (X \otimes X) \xrightarrow{\eta_X^T \otimes j_X} TX \otimes TX \xrightarrow{j_{TX}} TTX \xrightarrow{\eta^S \mu_X^T} STX.$$

*Definition 5.21.* For our second structural equational system  $E_2$ , we take

- as auxiliary bifunctor  $\Theta_2(X, Y) = X \otimes I$ ;
- as simple structural law, say  $d_2$ , we take the right-hand composite of Figure 3;
- as first structural interpretation, say  $L_2$ , at any  $X$ , we take

$$X \otimes I \xrightarrow{\eta_X^S \otimes v_X} SX \otimes SX \xrightarrow{j_{SX}} TSX \xrightarrow{\delta_X} STX;$$

- as second structural interpretation, say  $R_2$ , at any  $X$ , we simply take

$$X \otimes I \xrightarrow{\rho_X} X \xrightarrow{\eta_X^{ST}} STX.$$

**THEOREM 5.22.** *For any locally finitely presentable, leftist monoidal category  $(\mathbf{C}, \otimes, I, \alpha, \lambda, \rho)$  and pointed strong endofunctor  $\Sigma$ , the above data  $E_1$  and  $E_2$  indeed form structural equational systems, and  $(E_1 + E_2)$ -algebras form a category isomorphic to  $\Sigma$ -**Mon** over  $\mathbf{C}$ .*

*Furthermore, there are unique morphisms*

$$s: S_0 \otimes S_0 \rightarrow S_0 \qquad v: I \rightarrow S_0$$

*rendering the pentagon (10) and triangle (12) commutative.*

*Finally, these morphisms, together with  $\mu_0^S$ , satisfy the remaining  $\Sigma$ -monoid axioms, and make the initial  $\Sigma^+$ -algebra  $S_0$  into an initial  $\Sigma$ -monoid.*

**PROOF.** We start by reducing to proving the first claim, by observing that

- the second claim has already been stated, right before Proposition 5.18, and,
- assuming the first claim, the last one follows directly by Corollary 5.13.

The first claim is proved in §F. □

## 6 CONCLUSION AND PERSPECTIVES

We have introduced admissible monad morphisms as a foundation for syntax with auxiliary functions. We have then shown how to generate admissible morphisms from monad distributive laws, and defined simple structural laws and incremental structural laws as basic notions of signatures to generate such monad distributive laws, hence admissible morphisms. We have also defined structural equational systems as a basic format for ensuring that the generated auxiliary functions satisfy some (hopefully useful) properties. We have used these tools to cover significant examples of auxiliary functions, from addition and multiplication of natural numbers to binding evaluation contexts, capture-avoiding substitution, and partial differentiation. We have finally shown that the standard framework of Fiore et al. [1999] is subsumed by ours.

An important question, already raised in the introduction, remains open: can we devise some further tools to ensure that the auxiliary functions generated by our signatures are compatible with severe equations, i.e., equations on the basic syntax?

Finally, our framework is designed to account for auxiliary functions defined on top of an existing syntax. It would be useful to extend it to settings where the syntax and functions are mutually dependent, as in induction-recursion [Dybjer and Setzer 2001].

## REFERENCES

- Beniamino Accattoli. 2019. A Fresh Look at the lambda-Calculus (Invited Talk). In *Proc. 4th International Conference on Formal Structures for Computation and Deduction (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 131)*, Herman Geuvers (Ed.), 1:1–1:20. <https://doi.org/10.4230/LIPIcs.FSCD.2019.1>
- J. Adámek and J. Rosický. 1994. *Locally Presentable and Accessible Categories*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511600579>
- Guillaume Allais, Robert Atkey, James Chapman, Conor McBride, and James McKinna. 2018. A type and scope safe universe of syntaxes with binding: their semantics and proofs. *Proceedings of the ACM on Programming Languages 2*, ICFP (2018), 90:1–90:30. <https://doi.org/10.1145/3236785>
- Michael Barr. 1970. Coequalizers and free triples. *Mathematische Zeitschrift* 116 (1970), 307–322.
- Jon M. Beck. 1969. Distributive laws. In *Seminar on Triples and Categorical Homology Theory (Lecture Notes in Mathematics, Vol. 80)*, Beno Eckmann and Myles Tierney (Eds.). Springer.
- Stefan Berghofer and Christian Urban. 2007. A Head-to-Head Comparison of de Bruijn Indices and Names. *Electronic Notes in Theoretical Computer Science* 174, 5 (2007), 53–67. <https://doi.org/10.1016/j.entcs.2007.01.018>

- Richard S. Bird and Ross Paterson. 1999. De Bruijn notation as a nested datatype. *Journal of Functional Programming* 9, 1 (1999), 77–91. <https://doi.org/10.1017/S0956796889003366>
- Robert Oswald Blackwell. 1976. *Some existence theorems in the theory of doctrines*. Ph.D. Dissertation. University of New South Wales.
- Greta Coraglia and Ivan Di Liberti. 2021. Context, judgement, deduction. (2021). arXiv:2111.09438 [math.LO] Preprint.
- N. G. De Bruijn. 1972. Lambda-Calculus Notation with Nameless Dummies, a Tool for Automatic Formula Manipulation, with Application to the Church-Rosser Theorem. *Indagationes Mathematicae* 34 (1972), 381–392.
- Peter Dybjer and Anton Setzer. 2001. Indexed Induction-Recursion. In *Proof Theory in Computer Science, International Seminar (LNCS, Vol. 2183)*, Reinhard Kahle, Peter Schroeder-Heister, and Robert F. Stärk (Eds.). Springer, 93–113. [https://doi.org/10.1007/3-540-45504-3\\_7](https://doi.org/10.1007/3-540-45504-3_7)
- Thomas Ehrhard and Laurent Regnier. 2003. The differential lambda-calculus. *Theoretical Computer Science* 309, 1–3 (2003), 1–41. [https://doi.org/10.1016/S0304-3975\(03\)00392-X](https://doi.org/10.1016/S0304-3975(03)00392-X)
- Marcelo Fiore and Chung-Kil Hur. 2009. On the construction of free algebras for equational systems. *Theoretical Computer Science* 410 (2009), 1704–1729.
- Marcelo Fiore, Gordon Plotkin, and Daniele Turi. 1999. Abstract Syntax and Variable Binding. In *Proc. 14th Symposium on Logic in Computer Science* IEEE.
- Marcelo P. Fiore. 2008. Second-Order and Dependently-Sorted Abstract Syntax. In *LICS*. IEEE, 57–68. <https://doi.org/10.1109/LICS.2008.38>
- Murdoch J. Gabbay and Andrew M. Pitts. 1999. A New Approach to Abstract Syntax Involving Binders. In *Proc. 14th Symposium on Logic in Computer Science* IEEE.
- Daniel Gratzer and Jonathan Sterling. 2021. Syntactic categories for dependent type theories: sketching and adequacy. (2021). Preprint.
- André Hirschowitz, Tom Hirschowitz, and Ambroise Lafont. 2020. Modules over Monads and Operational Semantics. In *Proc. 5th International Conference on Formal Structures for Computation and Deduction (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 167)*, Zena M. Ariola (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 12:1–12:23. <https://doi.org/10.4230/LIPIcs.FSCD.2020.12>
- André Hirschowitz, Tom Hirschowitz, and Ambroise Lafont. 2022a. Modules over Monads and Operational Semantics. (2022). Submitted expanded version of [Hirschowitz et al. 2020].
- André Hirschowitz, Tom Hirschowitz, Ambroise Lafont, and Marco Maggesi. 2022b. Variable binding and substitution for (nameless) dummies. In *Proc. 25th Foundations of Software Science and Computational Structures (LNCS)*, Lutz Schröder and Patricia Bouyer (Eds.). Springer.
- Tom Hirschowitz, Xavier Leroy, and J. B. Wells. 2009. Compilation of extended recursion in call-by-value functional languages. *Higher Order and Symbolic Computation* 22, 1 (2009), 3–66. <https://doi.org/10.1007/s10990-009-9042-z>
- Martin Hofmann. 1999. Semantical Analysis of Higher-Order Abstract Syntax. In *Proc. 14th Symposium on Logic in Computer Science* IEEE.
- G. M. Kelly. 1980. A unified treatment of transfinite constructions for free algebras, free monoids, colimits, associated sheaves, and so on. *Bulletin of the Australian Mathematical Society* 22 (1980), 1–83.
- G. M. Kelly. 1982. *Basic concepts of enriched category theory*. London Mathematical Society Lecture Note Series, Vol. 64. Cambridge University Press. Republished as: *Reprints in Theory and Applications of Categories* 10 (2005).
- Stephen Lack. 1997. On the monadicity of finitary monads. *Journal of Pure and Applied Algebra* 140 (1997), 65–73.
- Saunders Mac Lane. 1998. *Categories for the Working Mathematician* (2nd ed.). Number 5 in Graduate Texts in Mathematics. Springer.
- Michel Parigot. 1992.  $\lambda\mu$ -calculus: an algorithmic interpretation of classical natural deduction. In *Proc. 19th International Colloquium on Automata, Languages and Programming (LNCS, Vol. 624)*. Springer.
- Benjamin C. Pierce (Ed.). 2004. *Advanced Topics in Types and Programming Languages*. MIT Press.
- Jan Reiterman. 1977. A left adjoint construction related to free triples. *Journal of Pure and Applied Algebra* 10 (1977), 57–71.
- Peter Sewell, Gareth Paul Stoye, Michael Hicks, Gavin M. Bierman, and Keith Wansbrough. 2008. Dynamic rebinding for marshalling and update, via redex-time and destruct-time reduction. *Journal of Functional Programming* 18, 4 (2008), 437–502. <https://doi.org/10.1017/S0956796807006600>
- Lionel Vaux. 2007.  *$\lambda$ -calcul différentiel et logique classique : interactions calculatoires*. Ph.D. Dissertation. Université Aix-Marseille 2.

## A PROOF OF PROPOSITION 2.4

First,  $\alpha_X : SX \rightarrow TX$  is an  $S$ -algebra morphism between  $SX$  and  $\mathbf{sem}(\alpha)(TX)$ . Since  $S\emptyset$  is an initial  $S$ -algebra,  $\mathbf{sem}(\alpha)(T\emptyset)$  is initial if and only if  $\alpha_\emptyset$  is an isomorphism. This argument shows (i)  $\Leftrightarrow$  (ii). Moreover, since  $S\emptyset$  is an initial  $S$ -algebra, creation implies preservation, so (iii)  $\Rightarrow$  (ii).

Finally, let us show that preservation implies creation, (ii)  $\Rightarrow$  (iii). Let  $SX \xrightarrow{x} X$  be an initial algebra. We show that there is a unique  $T$ -algebra structure on  $X$  that is mapped to  $x$  by  $\mathbf{sem}(\alpha)$ , and moreover that it is initial as a  $T$ -algebra. Since, by hypothesis,  $\mathbf{sem}(\alpha)$  preserves the initial object,  $\mathbf{sem}(\alpha)(T\emptyset)$  is also initial, and thus  $X$  is isomorphic to  $T\emptyset$ , as  $S$ -algebras. Now,  $\mathbf{sem}(\alpha)$  is an isofibration: through this isomorphism,  $X$  inherits a  $T$ -algebra structure which is mapped to  $SX \rightarrow X$  by  $\mathbf{sem}(\alpha)$ , and this isomorphism lifts to the category of  $T$ -algebras. Since this  $T$ -algebra  $TX \rightarrow X$  is isomorphic to  $T\emptyset$ , it is initial. It remains to show uniqueness. Consider an alternative  $T$ -algebra structure on  $X$ , that is also mapped to  $SX \rightarrow X$  by  $\mathbf{sem}(\alpha)$ . By initiality of  $X$  as a  $T$ -algebra, there is a morphism  $X \rightarrow X$  between these two  $T$ -algebras. It is enough to show that its image by  $\mathbf{sem}(\alpha)$  is the identity morphism, which follows from initiality of  $X$  as a  $S$ -algebra.

## B COLIMITS OF MONADS AND THEIR ALGEBRAS

In this section, we state a few useful results on colimits of monads, and on limits of the corresponding monadic functors, which notably entail Lemma 4.7.

*Definition B.1.* For any functor  $F: \mathbf{A} \rightarrow \mathbf{Mnd}(\mathbf{C})$ , let  $F\text{-Alg}$  denote a (choice of) limit of the induced diagram

$$\mathbf{A}^{op} \rightarrow \mathbf{Mnd}(\mathbf{C})^{op} \simeq \mathbf{Monadic}/\mathbf{C} \hookrightarrow \mathbf{CAT}/\mathbf{C}.$$

Thus, an  $F$ -algebra is an object  $X \in \mathbf{C}$ , equipped with compatible  $F_A$ -algebra structures, for all  $A \in \mathbf{A}$ .

The next proposition, due to Kelly, relates this with the colimit of  $F$ , when the latter exists.

For any functor  $F: \mathbf{A} \rightarrow \mathbf{Mnd}(\mathbf{C})$ , if  $T \in \mathbf{Mnd}(\mathbf{C})$  is a colimit of  $F$ , then the colimiting cocone induces a cone over the diagram

$$\mathbf{A}^{op} \rightarrow \mathbf{Mnd}(\mathbf{C})^{op} \simeq \mathbf{Monadic}/\mathbf{C} \hookrightarrow \mathbf{CAT}/\mathbf{C},$$

hence a functor  $m: T\text{-Alg} \rightarrow F\text{-Alg}$  over  $\mathbf{C}$ .

**PROPOSITION B.2** ([KELLY 1980, PROPOSITION 26.3]). *For any functor  $F: \mathbf{A} \rightarrow \mathbf{Mnd}(\mathbf{C})$  with colimit  $T$ , if  $\mathbf{C}$  is complete (and locally small), then the functor  $m: T\text{-Alg} \rightarrow F\text{-Alg}$  is an isomorphism.*

**Remark 19.** *Kelly explicitly requires  $\mathbf{C}$  to be locally small, which is implicit here by the convention fixed in §1.*

This immediately entails Lemma 4.7.

**COROLLARY B.3.** *Consider two monads  $S$  and  $T$  on a locally finitely presentable category  $\mathbf{C}$ , and a monad distributive law  $\delta: TS \rightarrow ST$ . For any bifunctor  $\Gamma: \mathbf{C} \times \mathbf{C} \rightarrow \mathbf{C}$ , letting  $T' := \Gamma_S^*$ , the following square is a pullback.*

$$\begin{array}{ccc} (T \oplus T')\text{-Alg} & \longrightarrow & T\text{-Alg} \\ \downarrow \lrcorner & & \downarrow \\ \Gamma_S\text{-alg} & \longrightarrow & \mathbf{C} \end{array}$$

**PROOF.** By Lemma 4.7 and the isomorphism  $T'\text{-Alg} \cong \Gamma_S\text{-alg}$  over  $\mathbf{C}$ . □

**Remark 20.** *Otherwise said, giving a  $(T \oplus T')$ -algebra structure on  $X$  is equivalent to giving a  $T$ -algebra structure and a  $\Gamma_S$ -algebra structure.*

### C PROOF OF THEOREM 4.10

*Definition C.1.* A bifunctor  $\Gamma: \mathbf{C}^2 \rightarrow \mathbf{C}$  on a category  $\mathbf{C}$  is **bipointed** if it comes equipped with natural transformations

$$X \xrightarrow{\alpha_{X,Y}^\Gamma} \Gamma(X, Y) \xleftarrow{\beta_{X,Y}^\Gamma} Y.$$

*Definition C.2.* For any bifunctor  $\Gamma$ , let  $\Gamma^\bullet(X, Y) := \Gamma(X, Y) + X + Y$ .

**PROPOSITION C.3.** *For any bifunctor  $\Gamma$ , the bifunctor  $\Gamma^\bullet$  is bipointed. (It is in fact the free bipointed bifunctor over  $\Gamma$ .)*

**PROOF.** Straightforward. □

**Notation 6.** *When applied to, say,  $\Gamma_F$  for some endofunctor  $F$ , the notation  $\Gamma_F^\bullet$  is ambiguous, as it could denote  $(\Gamma^\bullet)_F$  or  $(\Gamma_F)^\bullet$ . The issue is even worse for, e.g.,  $\Gamma_{GF}^\bullet$ . In order to resolve the ambiguity, we write  $\Gamma_{G|F}^\bullet$  for  $(\Gamma_G)_{F^\bullet}^\bullet$ , i.e.,*

$$\Gamma_{G|FY}^\bullet X := \Gamma(X, GFY) + X + FY.$$

We now want to show that any incremental structural law

$$h_{X,Y}: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{SY}(X) + X + Y),$$

or using Notation 6

$$h_{X,Y}: \Gamma_Y \Sigma X \rightarrow ST\Gamma_{ST|Y}^\bullet X,$$

induces an incremental lifting of  $S$  to  $T'$  - **Alg** along  $\delta$ . For this, we extend  $h$  to a natural transformation

$$h_{X,Y}^{\bullet\omega}: \Gamma_{SY} SX \rightarrow ST(\Gamma_{ST|Y}^\bullet(X))$$

making the following triangle commute

$$\begin{array}{ccc} \Gamma_Y \Sigma X & \xrightarrow{\Gamma_Y \eta_{\Sigma, X}} & \Gamma_{SY} SX \\ & \searrow h_{X,Y} & \swarrow h_{X,Y}^{\bullet\omega} \\ & ST\Gamma_{ST|Y}^\bullet X, & \end{array}$$

and then show how it induces an incremental lifting. This route being slightly long-winded, let us point to the definition of  $h^{\bullet\omega}$  (Definition C.35) and the proof that it induces an incremental lifting (Proposition C.38).

The plan is to successively define families

- $h_{X,Y}^\bullet: \Gamma_{S|Y}^\bullet(\Sigma(X)) \rightarrow ST(\Gamma_{ST|Y}^\bullet(X))$ ,
- $h_{X,Y}^\circ: \Gamma_{STY}^\circ \Sigma X \rightarrow ST\Gamma_{ST|Y}^\bullet X$ ,
- $\tilde{h}_{X,Y}^\circ: \Sigma X \rightarrow T_{STY}^\times STO_Y \Gamma_{STY}^\circ X$ ,
- $\tilde{h}_{X,Y}^{\circ\omega}: SX \rightarrow T_{STY}^\times STO_Y \Gamma_{STY}^\circ X$ ,
- $\tilde{h}_{X,Y}^{\circ\omega}: \Gamma_{STY}^\circ SX \rightarrow STO_Y \Gamma_{STY}^\circ X$

(involving new functors defined along the way), from which we then define  $h^{\bullet\omega}$  and prove that it satisfies the properties stated in Lemma C.37 below. We then use this to construct the desired incremental lifting.

### C.1 The family $h^\bullet$

Let us start with  $h^\bullet$ , which requires the following preliminary definition.

*Definition C.4.* Consider any pointed endofunctor  $F$ . We let  $\uparrow^F$  denote the natural transformation defined at any  $X, Y \in \mathbf{C}$  by the composite

$$X + FY \xrightarrow{\eta_X^F + FY} FX + FY \xrightarrow{[Fin_1, Fin_2]} F(X + Y).$$

**Remark 21.** In particular, for any  $F$  and  $G$  with  $F$  pointed, we have

$$\uparrow_{\Gamma_{GFY}X+X,Y}^F: \Gamma_G^\bullet|_{FY}X \rightarrow F\Gamma_{GF}^\bullet|_YX.$$

**LEMMA C.5.** For any bifunctor  $G: \mathbf{C}^2 \rightarrow \mathbf{C}$ , distributive law  $\delta: TS \rightarrow ST$ , and objects  $X, Y \in \mathbf{C}$ , the following diagram commutes.

$$\begin{array}{ccccc} G_{|TSY}^\bullet X & \xrightarrow{G_{|\delta_Y}^\bullet X} & G_{|STY}^\bullet X & \xrightarrow{\uparrow^S} & S_{|TY}^\bullet X \\ \uparrow^T \downarrow & & & & \downarrow S\uparrow^T \\ TG_{|SY}^\bullet X & \xrightarrow{T\uparrow^S} & TSG_{|SY}^\bullet X & \xrightarrow{\delta_{G_{|\delta_Y}^\bullet X}} & STG_{|SY}^\bullet X \end{array}$$

**PROOF.** Unfolding the definition, we refine the claim as follows,

$$\begin{array}{ccccccc} G_{TSY}X + X + TSY & \xrightarrow{G_{\delta_Y}X+X+\delta_Y} & G_{STY}X + X + STY & \xrightarrow{\eta^S+STY} & S(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(G_{STY}X + X + TY) \\ \eta^T+TSY \downarrow & & \downarrow \eta^T+STY & & \downarrow S\eta^T+STY & & \downarrow S(\eta^T+TY) \\ T(G_{TSY}X + X) + TSY & \xrightarrow{T(G_{\delta_Y}X+X+\delta_Y)} & T(G_{STY}X + X) + STY & \xrightarrow{\eta^S+STY} & ST(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(T(G_{STY}X + X) + TY) \\ \downarrow [Tim_1, Tim_2] & \searrow \eta^S+TSY & \nearrow ST(G_{\delta_Y}X+X+\delta_Y) & & \searrow [STin_1, STin_2] & & \downarrow S[Tim_1, Tim_2] \\ T(G_{TSY}X + X + SY) & \xrightarrow{T(\eta^S+SY)} & T(S(G_{STY}X + X) + SY) & \xrightarrow{T[Sin_1, Sin_2]} & TS(G_{TSY}X + X + Y) & \xrightarrow{\delta_{(G_{\delta_Y}X+X+Y)}} & ST(G_{STY}X + X + Y) \end{array}$$

where all subdiagrams easily commute except the top right square and the bottom left polygon. Both have coproducts as their domains, so we chase them termwise. The first term of the top right square is chased as follows.

$$\begin{array}{ccccc} S(G_{STY}X + X) & \xrightarrow{Sin_1} & S(G_{STY}X + X + TY) & & \\ \downarrow in_1 & \searrow S\eta^T & \downarrow \parallel & \searrow S(\eta^T+TY) & \\ S(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(G_{STY}X + X + TY) & \xrightarrow{S(\eta^T+TY)} & S(T(G_{STY}X + X) + TY) \\ \downarrow S\eta^T+STY & \downarrow in_1 & \downarrow \parallel & \downarrow \parallel & \\ ST(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(T(G_{STY}X + X) + TY) & & \end{array}$$

The second term of the top right square is chased as follows.

A unified treatment of structural definitions on syntax

$$\begin{array}{ccccc}
 STY & \xlongequal{\quad} & STY & \xlongequal{\quad} & STY \\
 \downarrow in_2 & & \downarrow Sin_2 & & \downarrow Sin_2 \\
 S(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(G_{STY}X + X + TY) & \xrightarrow{S(\eta^T + TY)} & S(T(G_{STY}X + X) + TY) \\
 \downarrow S\eta^T + STY & & \downarrow in_2 & & \downarrow Sin_2 \\
 ST(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & ST(G_{STY}X + X) + STY & \xrightarrow{[Sin_1, Sin_2]} & S(T(G_{STY}X + X) + TY)
 \end{array}$$

The first term of the bottom left polygon is chased as follows.

$$\begin{array}{ccccccc}
 T(G_{TSY}X + X) & \xrightarrow{\eta^s} & ST(G_{TSY}X + X) & \xrightarrow{ST(G_{G_y, X+X})} & ST(G_{STY}X + X) & \xrightarrow{\delta} & ST(G_{STY}X + X) \\
 \downarrow in_1 & \searrow T(G_{TSY}X + X) & \downarrow in_1 & \searrow \delta & \downarrow in_1 & \searrow ST(G_{G_y, X+X}) & \downarrow ST in_1 \\
 T(G_{TSY}X + X) + TSY & \xrightarrow{T\eta^s} & TS(G_{TSY}X + X) & \xrightarrow{ST(G_{G_y, X+X}) + \delta y} & ST(G_{STY}X + X) + STY & \xrightarrow{[ST in_1, ST in_2]} & ST(G_{STY}X + X + Y) \\
 \downarrow [T in_1, T in_2] & \searrow T in_1 & \downarrow T in_1 & \searrow T in_1 & \downarrow [ST in_1, ST in_2] & \searrow ST in_1 & \downarrow ST in_1 \\
 T(G_{TSY}X + X + SY) & \xrightarrow{T(\eta^s + TSY)} & T(S(G_{TSY}X + X) + SY) & \xrightarrow{T[Sin_1, Sin_2]} & TS(G_{TSY}X + X + Y) & \xrightarrow{\delta(G_{G_y, X+X+Y})} & ST(G_{STY}X + X + Y)
 \end{array}$$

Finally, the second term of the bottom left polygon is chased as follows.

$$\begin{array}{ccccccc}
 TSY & \xlongequal{\quad} & TSY & \xlongequal{\quad} & TSY & \xlongequal{\quad} & STY \\
 \downarrow in_2 & & \downarrow in_2 & & \downarrow in_2 & & \downarrow in_2 \\
 T(G_{TSY}X + X) + TSY & \xrightarrow{[T in_1, T in_2]} & T(G_{TSY}X + X + SY) & \xrightarrow{T(\eta^s + TSY)} & T(S(G_{TSY}X + X) + SY) & \xrightarrow{T[Sin_1, Sin_2]} & TS(G_{TSY}X + X + Y) \\
 \downarrow [T in_1, T in_2] & \searrow T in_1 & \downarrow T in_1 & \searrow T in_1 & \downarrow [ST in_1, ST in_2] & \searrow ST in_1 & \downarrow ST in_2 \\
 T(G_{TSY}X + X) + TSY & \xrightarrow{T in_1} & ST(G_{TSY}X + X) + STY & \xrightarrow{ST(G_{G_y, X+X}) + \delta y} & ST(G_{STY}X + X) + STY & \xrightarrow{\delta(G_{G_y, X+X+Y})} & ST(G_{STY}X + X + Y)
 \end{array} \quad \square$$

*Definition C.6.* Let

$$h_{X,Y}^\bullet : \Gamma_{S|Y}^\bullet(\Sigma(X)) \rightarrow ST(\Gamma_{ST|Y}^\bullet(X)),$$

be defined by cotupling the following three composites.

$$\begin{array}{l}
 \Gamma_{SY} \Sigma X \xrightarrow{h_{X,SY}} ST\Gamma_{ST|SY}^\bullet X \xrightarrow{ST\uparrow^S} STS\Gamma_{STS|Y}^\bullet X \xrightarrow{S\delta; \mu^S T} ST\Gamma_{STS|Y}^\bullet X \xrightarrow{(S\delta; \mu^S T)|Y^X} ST\Gamma_{ST|Y}^\bullet X \\
 \Sigma X \xrightarrow{\eta_{\Sigma, \eta^T} X} STX \xrightarrow{ST\alpha_{X,Y}^\bullet} ST(\Gamma_{ST|Y}^\bullet(X)) \quad Y \xrightarrow{\eta_Y^{ST}} STY \xrightarrow{ST\beta_{X,Y}^\bullet} ST(\Gamma_{ST|Y}^\bullet(X))
 \end{array}$$

LEMMA C.7. For all  $h, h^\bullet$  satisfies the following coherence laws.





By the last lemma,  $h_{X,Y}^\bullet$  equivalently has type

$$\Gamma_{S|Y}^\bullet \Sigma X \rightarrow STO_Y \Gamma_{STY}^\circ X.$$

*Definition C.12.* Let  $h_{X,Y}^\circ$  denote the natural transformation with components

$$\Gamma_{STY}^\circ \Sigma X \xrightarrow{[in_1, in_2]} \Gamma_{S|TY}^\bullet \Sigma X \xrightarrow{h_{X,TY}^\bullet} ST\Gamma_{ST|TY}^\bullet X \xrightarrow{ST\uparrow^T} STT\Gamma_{ST|Y}^\bullet X \xrightarrow{S\mu^T \Gamma_{S\mu^T|Y}^\bullet X} ST\Gamma_{ST|Y}^\bullet X.$$

LEMMA C.13. For all  $h$ ,  $h^\circ$  satisfies the following coherence laws.

$$\begin{array}{ccc} \Sigma(X) & \xrightarrow{\eta_\Sigma, \eta_X^T} & ST(X) \\ \alpha_{\Sigma X, STY}^\circ \downarrow & & \downarrow ST\alpha_{X,Y}^\bullet \\ \Gamma_{STY}^\circ(\Sigma(X)) & \xrightarrow{h_{X,Y}^\circ} & ST(\Gamma_{ST|Y}^\bullet(X)) \end{array}$$

$$\begin{array}{ccc} \Gamma_{STSY}^\circ \Sigma X & \xrightarrow{\Gamma_{(S\delta, \mu^{ST})_Y}^\circ \Sigma X} & \Gamma_{STY}^\circ \Sigma X \\ h_{X,SY}^\circ \downarrow & & \downarrow h_{X,Y}^\circ \\ ST\Gamma_{ST|SY}^\bullet X & \xrightarrow{ST\uparrow_{X,Y}^S} & STST\Gamma_{ST|Y}^\bullet X \xrightarrow{(S\delta, \mu^{ST})_Y \Gamma_{S\delta, \mu^{ST}|Y}^\bullet X} & ST\Gamma_{ST|Y}^\bullet X \end{array}$$

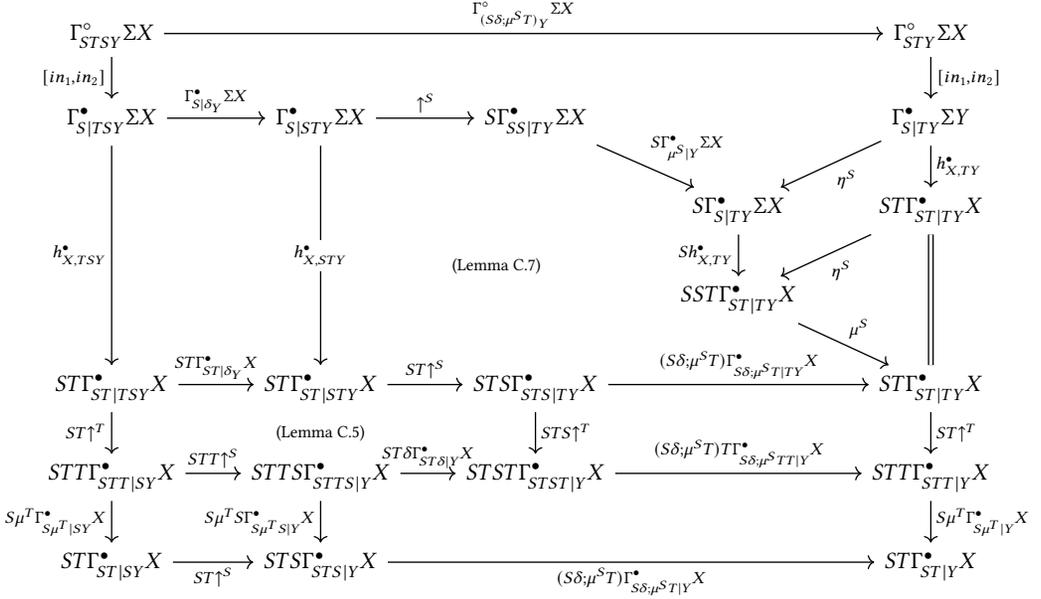
$$\begin{array}{ccc} \Gamma_Y \Sigma X & \xrightarrow{\Gamma_{\eta_Y^{ST}} \Sigma X} & \Gamma_{STY} \Sigma X \xrightarrow{in_1} & \Gamma_{STY}^\circ \Sigma X \\ & \searrow h_{X,Y} & \downarrow & \swarrow h_{X,Y} \\ & & ST\Gamma_{ST|Y}^\bullet X & \end{array}$$

$$\begin{array}{ccc} \Gamma_{STTY}^\circ \Sigma X & \xrightarrow{\Gamma_{S\mu_Y^T}^\circ \Sigma X} & \Gamma_{STY}^\circ \Sigma X \\ h_{X,TY}^\circ \downarrow & & \downarrow h_{X,Y}^\circ \\ STO_{TY} \Gamma_{STTY}^\circ X & \xrightarrow{ST\uparrow^T} & STTO_Y \Gamma_{STTY}^\circ X \xrightarrow{S\mu^T O_Y \Gamma_{STTY}^\circ X} & STO_Y \Gamma_{STTY}^\circ X \xrightarrow{STO_Y \Gamma_{S\mu_Y^T}^\circ} & STO_Y \Gamma_{STY}^\circ X \end{array}$$

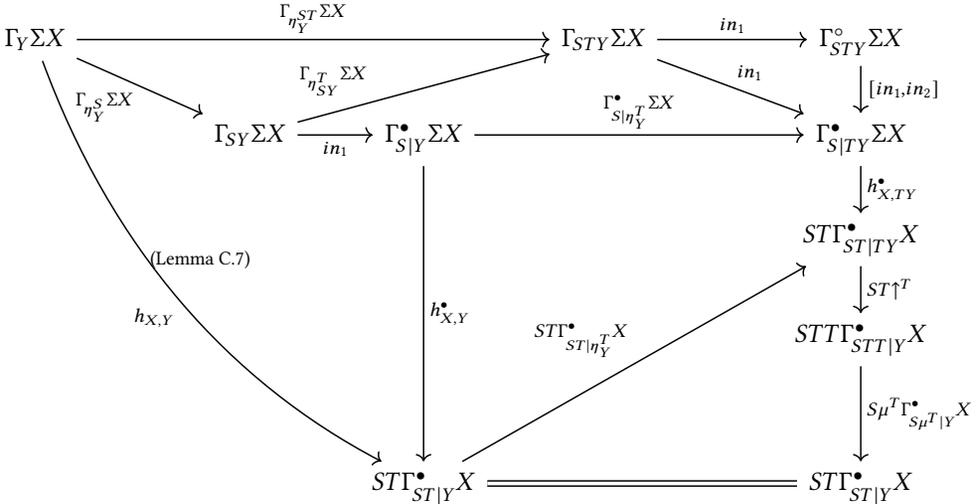
PROOF. The first law holds by chasing the following diagram.

$$\begin{array}{ccccccc} \Sigma X & \xrightarrow{\eta_\Sigma \eta_X^T} & STX & \xrightarrow{\quad\quad\quad} & STX & & \\ \downarrow in_2 & \searrow in_2 & \downarrow STin_2 & \searrow ST\eta^T & \downarrow S\mu_X^T & \downarrow STin_2 & \\ \Gamma_{STY}^\circ \Sigma X & \xrightarrow{[in_1, in_2]} & \Gamma_{S|TY}^\bullet \Sigma X \xrightarrow{h_{X,TY}^\bullet} & ST\Gamma_{ST|TY}^\bullet X \xrightarrow{ST\uparrow^T} & STT\Gamma_{ST|Y}^\bullet X \xrightarrow{S\mu^T \Gamma_{S\mu^T|Y}^\bullet X} & ST\Gamma_{ST|Y}^\bullet X & \end{array}$$

For the second one, we proceed by diagram chasing, as follows.



Let us now consider the third claim. It follows by chasing the following diagram,



whose bottom right triangle commutes by chasing the following diagram.

$$\begin{array}{ccc}
 T(\Gamma_{STY}X + X + Y) & \xrightarrow{T\Gamma_{ST|Y}^{\bullet} X} & T(\Gamma_{STTY}X + X + TY) \\
 \searrow & \searrow & \downarrow T(\eta^T + TY) \\
 & T(\Gamma_{ST\eta_Y}^T X + X + Y) & \xrightarrow{T(\eta^T + \eta_Y^T)} & T(T(\Gamma_{STTY}X + X) + TY) \\
 & \searrow & \searrow & \downarrow T[in_1, in_2] \\
 & & T\eta^T & TT\Gamma_{STT|Y}^{\bullet} X \\
 & & & \downarrow \mu^T \Gamma_{STT|Y}^{\bullet} X \\
 & & & T\Gamma_{STT|Y}^{\bullet} X \\
 & & & \downarrow T\Gamma_{S\mu^T|Y}^{\bullet} X \\
 & & & T\Gamma_{ST|Y}^{\bullet} X
 \end{array}$$

Finally, we prove the last claim by diagram chasing as follows,

$$\begin{array}{ccc}
 \Gamma_{STTY}^{\circ} \Sigma X & \xrightarrow{\Gamma_{S\mu_Y^T}^{\circ} \Sigma X} & \Gamma_{STY}^{\circ} \Sigma X \\
 [in_1, in_2] \downarrow & & \downarrow [in_1, in_2] \\
 \Gamma_{STTY}^{\bullet} \Sigma X & \xrightarrow{\Gamma_{S\mu_Y^T}^{\bullet} \Sigma X} & \Gamma_{STY}^{\bullet} \Sigma X \\
 h_{x, TTY}^* \downarrow & & \downarrow h_{x, TY}^* \\
 STOTTY \Gamma_{STTY}^{\circ} X & \xrightarrow{STO_{\mu_Y^T} \Gamma_{ST\mu_Y^T}^{\circ} X} & STOTY \Gamma_{STTY}^{\circ} X \\
 ST\uparrow^T \downarrow & & \downarrow ST\uparrow^T \\
 STTOTY \Gamma_{STTY}^{\circ} X & \xrightarrow{STT\uparrow^T} STTTOY \Gamma_{STTY}^{\circ} X & \xrightarrow{ST\mu^T O_Y \Gamma_{ST\mu_Y^T}^{\circ} X} STTOY \Gamma_{STTY}^{\circ} X \\
 S\mu^T O_Y \Gamma_{S\mu_Y^T}^{\circ} X \downarrow & & \downarrow S\mu^T T O_Y \Gamma_{S\mu_Y^T}^{\circ} X \\
 STOTY \Gamma_{STTY}^{\circ} X & \xrightarrow{ST\uparrow^T} STTOY \Gamma_{STTY}^{\circ} X & \xrightarrow{S\mu^T O_Y \Gamma_{STTY}^{\circ} X} STOY \Gamma_{STTY}^{\circ} X & \xrightarrow{STO_Y \Gamma_{S\mu_Y^T}^{\circ} X} STOY \Gamma_{STY}^{\circ} X
 \end{array}$$

where

- both top rectangles, as well as the bottom left and bottom right rectangles, commute by naturality, and
- the third rectangle commutes by naturality and Lemma C.14 below.  $\square$

LEMMA C.14. *For any monad  $T$  on a category with binary coproducts, for all objects  $X$  and  $Y$ , the following diagram commutes.*

$$\begin{array}{ccc}
 OTTYX & \xrightarrow{O_{\mu_Y^T} X} & OTYX \\
 \uparrow^T \downarrow & & \downarrow \uparrow^T \\
 TOTYX & \xrightarrow{T\uparrow^T} TTOYX & \xrightarrow{\mu^T O_Y X} TOYX
 \end{array}$$

PROOF. This hold by diagram chasing, as follows.

$$\begin{array}{ccc}
 X + TTY & \xrightarrow{X + \mu_Y^T} & X + TY \\
 \eta_X^T + TTY \downarrow & & \downarrow \eta_X^T + TY \\
 TX + TTY & \xrightarrow{TX + \mu_Y^T} & TX + TY \\
 [Tin_1, Tin_2] \downarrow & \searrow^{T\eta_X^T + TTY} & \downarrow [Tin_1, Tin_2] \\
 T(X + TY) & \xrightarrow{T(\eta_X^T + TY)} T(TX + TY) & \xrightarrow{T[Tin_1, Tin_2]} TT(X + Y) \xrightarrow{\mu^T} T(X + Y) \\
 & \swarrow [Tin_1, Tin_2] & \searrow [TTin_1, TTin_2] \\
 & TTX + TTY & \\
 & \xrightarrow{\mu_X^T + \mu_Y^T} & 
 \end{array}$$

### C.3 The family $\tilde{h}^\circ$

We now turn to defining our next family of morphisms,  $\tilde{h}^\circ$ . For this, we need to exploit the cocontinuity hypothesis in the first argument of  $\Gamma$ .

Because we have assumed  $\mathbf{C}$  to be locally finitely presentable, each  $\Gamma_Y$  has a right adjoint, which we denote by  $\mathbb{T}_Y$ . Furthermore, we have:

LEMMA C.15. *Each  $\Gamma_Y^\circ$  has as right adjoint the functor  $\mathbb{T}_Y^\times$  defined by  $\mathbb{T}_Y^\times(X) = \mathbb{T}_Y(X) \times X$ .*

PROOF. We have

$$\begin{aligned}
 \mathbf{C}(\Gamma_Y^\circ(X), Z) &= \mathbf{C}(\Gamma_Y(X) + X, Z) \\
 &\cong \mathbf{C}(\Gamma_Y(X), Z) \times \mathbf{C}(X, Z) \\
 &\cong \mathbf{C}(X, \mathbb{T}_Y Z) \times \mathbf{C}(X, Z) \\
 &\cong \mathbf{C}(X, \mathbb{T}_Y(Z) \times Z) \\
 &\cong \mathbf{C}(X, \mathbb{T}_Y^\times(Z)).
 \end{aligned}$$

Furthermore, by [Mac Lane 1998, Theorem IV.7.3], we readily get:

PROPOSITION C.16. *For all bifunctors  $L: \mathbf{C}^2 \rightarrow \mathbf{C}$  such that each  $L_Y$  has a right adjoint  $R_Y$ , the functors  $R_Y$  assemble into a bifunctor  $\mathbf{C} \times \mathbf{C}^{op} \rightarrow \mathbf{C}$  making the bijection*

$$\mathbf{C}(L_Y(X), Z) \cong \mathbf{C}(X, R_Y(Z))$$

natural in all variables.

Here, naturality in  $Y$  means that, e.g., for all morphisms  $f: Y \rightarrow Y'$  and  $u: L_{Y'}X \rightarrow Z$ , the transpose of

$$L_Y X \xrightarrow{L_f X} L_{Y'} X \xrightarrow{u} Z \quad \text{is} \quad X \xrightarrow{\tilde{u}} R_{Y'} Z \xrightarrow{R_f Z} R_Y Z,$$

where  $\tilde{u}$  denotes the transpose of  $u$ .

COROLLARY C.17. *For all  $f: Y \rightarrow Z$ ,  $g: B \rightarrow C$ ,  $u: A \rightarrow R_Y B$ , and  $v: A \rightarrow R_Z C$ , letting  $\tilde{u}$  and  $\tilde{v}$  denote the transposed morphisms, the square below left commutes iff the one below right does.*

$$\begin{array}{ccc}
 A \xrightarrow{u} R_Y B & & L_Y A \xrightarrow{\tilde{u}} B \\
 v \downarrow & \searrow^{R_Y g} & \downarrow L_f A \\
 R_Z C \xrightarrow{R_f C} R_Y C & & L_Z A \xrightarrow{\tilde{v}} C
 \end{array}$$

In particular, for all  $A$  and  $f: B \rightarrow C$ , the following diagrams commute.

$$\begin{array}{ccc}
 L_B R_C A \xrightarrow{L_f R_C A} L_C R_C A & & A \xrightarrow{\eta_C A} R_C L_C A \\
 L_B R_f A \downarrow & \searrow^{\epsilon_C A} & \downarrow R_f L_C A \\
 L_B R_B A \xrightarrow{\epsilon_B A} A & & R_B L_B A \xrightarrow{R_B L_f A} R_B L_C A
 \end{array}$$

Let us return to the construction of the next family.

*Definition C.18.* Let  $\tilde{h}^\circ_{X,Y}$  denote the transpose

$$\tilde{h}^\circ_{X,Y} : \Sigma X \rightarrow \mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ X$$

of  $h^\circ_{X,Y}$ .

*LEMMA C.19.* *The family  $\tilde{h}^\circ$  is natural in  $X$ , i.e., for all  $Y$  and  $f : X \rightarrow Z$ , the following square commutes.*

$$\begin{array}{ccc} \Sigma X & \xrightarrow{\Sigma f} & \Sigma Z \\ \tilde{h}^\circ_{X,Y} \downarrow & & \downarrow \tilde{h}^\circ_{Z,Y} \\ \mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ X & \xrightarrow{\mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ f} & \mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ Z \end{array}$$

*PROOF.* By naturality and bijectivity of transposition.  $\square$

*LEMMA C.20.* *The family  $\tilde{h}^\circ$  is extranatural [Kelly 1982] in  $Y$ , i.e., for all  $X$  and  $f : Y \rightarrow Z$ , the following square commutes.*

$$\begin{array}{ccc} \Sigma X & \xrightarrow{\tilde{h}^\circ_{X,Y}} & \mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ X \\ \tilde{h}^\circ_{X,Z} \downarrow & & \downarrow \mathbb{T}_{STY}^\times \text{STO}_f \Gamma_{STY}^\circ X \\ \mathbb{T}_{STZ}^\times \text{STO}_Z \Gamma_{STZ}^\circ X & \xrightarrow{\mathbb{T}_{STZ}^\times \text{STO}_Z \Gamma_{STZ}^\circ X} & \mathbb{T}_{STY}^\times \text{STO}_Z \Gamma_{STZ}^\circ X \end{array}$$

*PROOF.* By naturality of  $h^\circ$  and Corollary C.17.  $\square$

*LEMMA C.21.* *The family  $\tilde{h}^\circ$  satisfies the following laws.*

$$\begin{array}{ccc} \Sigma X & \xrightarrow{\eta_\Sigma, \eta_X^T} & STX \\ \tilde{h}^\circ_{X,Y} \downarrow & & \downarrow \text{ST}\alpha_{X,Y}^{\Gamma_{ST}} \\ \mathbb{T}_{STY}^\times \text{ST}\Gamma_{ST|Y}^\bullet X & \xrightarrow{\pi_2} & \text{ST}\Gamma_{ST|Y}^\bullet X \end{array}$$
  

$$\begin{array}{ccc} \Sigma X & \xrightarrow{\tilde{h}^\circ_{X,Y}} & \mathbb{T}_{STY}^\times \text{ST}\Gamma_{ST|Y}^\bullet X \\ \tilde{h}^\circ_{X,SY} \downarrow & & \downarrow \mathbb{T}_{STY}^\times \mu_{SY}^{\text{ST}\Gamma_{ST|Y}^\bullet X} \\ \mathbb{T}_{STSY}^\times \text{ST}\Gamma_{ST|SY}^\bullet X & \xrightarrow{\mathbb{T}_{STSY}^\times \text{ST}\uparrow_{X,Y}^S} & \mathbb{T}_{STSY}^\times \text{ST}\text{ST}\Gamma_{ST|SY}^\bullet X \xrightarrow{\mathbb{T}_{STSY}^\times (S\delta; \mu^S T)\Gamma_{S\delta; \mu^S T|Y}^\bullet X} & \mathbb{T}_{STSY}^\times \text{ST}\Gamma_{ST|Y}^\bullet X \\ & & & \downarrow \mathbb{T}_{STSY}^\times \mu_{SY}^{\text{ST}\Gamma_{ST|Y}^\bullet X} \\ & & & \mathbb{T}_{SSY}^\times \text{ST}\Gamma_{STY}^\bullet X \\ & & & \downarrow \mathbb{T}_{SSY}^\times \mu_{SY}^{\text{ST}\Gamma_{STY}^\bullet X} \\ & & & \mathbb{T}_{SY}^\times \text{ST}\Gamma_{ST|Y}^\bullet X \end{array}$$
  

$$\begin{array}{ccc} \Sigma X & \xrightarrow{\tilde{h}^\circ_{X,Y}} & \mathbb{T}_{STY}^\times \text{ST}\Gamma_{ST|Y}^\bullet X \\ & \searrow \tilde{h}_{X,Y} & \downarrow \pi_1 \\ & & \mathbb{T}_{STY} \text{ST}\Gamma_{ST|Y}^\bullet X \\ & & \downarrow \mathbb{T}_{\eta_Y^{\text{ST}\Gamma_{ST|Y}^\bullet X}} \\ & & \mathbb{T}_Y \text{ST}\Gamma_{ST|Y}^\bullet X \end{array}$$

$$\begin{array}{ccc}
 \Sigma A & \xrightarrow{\tilde{h}^\circ_{\mathbb{T}_{STY}^\times \mathbb{ST}\Gamma_{ST|Y}^\bullet \mathbb{X}, Y}} & \mathbb{T}_{STY}^\times \mathbb{STO}_Y \Gamma_{STY}^\circ A \\
 \tilde{h}^\circ_{\mathbb{T}_{STY}^\times \mathbb{ST}\Gamma_{ST|Y}^\bullet \mathbb{X}, TY} \downarrow & & \downarrow \Gamma_{S\mu_Y}^\times \mathbb{STO}_Y \Gamma_{STY}^\circ A \\
 \mathbb{T}_{STY}^\times \mathbb{STO}_{TY} \Gamma_{STY}^\circ A & \xrightarrow{\mathbb{T}_{STY}^\times \mathbb{ST}\uparrow} \mathbb{T}_{STY}^\times \mathbb{STTO}_Y \Gamma_{STY}^\circ A & \xrightarrow{\mathbb{T}_{STY}^\times S\mu^T \mathbb{O}_Y \Gamma_{S\mu_Y}^\circ A} \mathbb{T}_{STY}^\times \mathbb{STO}_Y \Gamma_{STY}^\circ A
 \end{array}$$

PROOF. The first diagram holds by Lemma C.13 and construction of  $\mathbb{T}_{STY}^\times$ . The second one holds by Lemma C.13 and Corollary C.17. The third diagram holds by Lemma C.13, observing that by construction of the adjunction of Lemma C.15,  $\pi_1 \circ \tilde{h}^\circ_{X,Y}$  is the transpose of

$$\Gamma_{STY} \Sigma X \xrightarrow{in_1} \Gamma_{STY}^\circ \Sigma X \xrightarrow{h_{X,Y}^\circ} \mathbb{ST}\Gamma_{STY}^\bullet X,$$

hence, by Proposition C.16,  $\mathbb{T}_{\eta_Y^{ST}} \mathbb{ST}\Gamma_{ST|Y}^\bullet X \circ \pi_1 \circ \tilde{h}^\circ_{X,Y}$  is the transpose of

$$\Gamma_Y \Sigma X \xrightarrow{\Gamma_{\eta_Y^{ST}} \Sigma X} \Gamma_{STY} \Sigma X \xrightarrow{in_1} \Gamma_{STY}^\circ \Sigma X \xrightarrow{h_{X,Y}^\circ} \mathbb{ST}\Gamma_{STY}^\bullet X.$$

Finally, the last diagram is precisely the transpose of the last diagram of Lemma C.13.  $\square$

#### C.4 The family $\tilde{h}^\omega$

We now want to define the next family,  $\tilde{h}^\omega$ . For this, we start by observing the following: We have:

LEMMA C.22. *For all  $Y$ , the composite functor  $\mathbb{T}_{STY}^\times \mathbb{STO}_Y \Gamma_{STY}^\circ$  is a monad.*

PROOF. We already know that  $ST$  is a monad. Furthermore, precomposing any monad by any option functor  $O_Z$  again yields a monad, by the following (folklore) lemma. We conclude by the adjunction  $\mathbb{T}_{STY}^\times \vdash \Gamma_{STY}^\circ$ .  $\square$

Furthermore, the following result is folklore.

LEMMA C.23. *For all monads  $T$  on a category with binary coproducts, and for all objects  $E$  therein, the natural transformation  $T\uparrow_E : O_ET \rightarrow TO_E$  defined at any  $X$  by*

$$TX + E \xrightarrow{[TX, \eta_E^T]} T(X) + T(E) \xrightarrow{[T(in_1), T(in_2)]} T(X + E)$$

*forms a monad distributive law, thus equipping  $TO_E$  with monad structure.*

We may now define  $\tilde{h}^\omega$ .

Definition C.24. For all  $X, Y \in \mathbb{C}$ , let  $\tilde{h}^\omega_{X,Y} : SX \rightarrow \mathbb{T}_{STY}^\times \mathbb{STO}_Y \Gamma_{STY}^\circ X$  denote the unique monad morphism making the triangle

$$\begin{array}{ccc}
 \Sigma X & \xrightarrow{\eta_{\Sigma, X}} & SX \\
 \tilde{h}^\omega_{X,Y} \searrow & & \downarrow \tilde{h}^\omega_{X,Y} \\
 & & \mathbb{T}_{STY}^\times \mathbb{ST}\Gamma_{ST|Y}^\bullet X
 \end{array}$$

commute, obtained by universal property of  $S$ .

By construction, we have:

LEMMA C.25. *The family  $\tilde{h}^\omega_{X,Y}$  is natural in  $X$ .*

**Notation 7.** Let  $L = \Gamma_{ST}^\circ : \mathbf{C}^2 \rightarrow \mathbf{C}$ ,  $R = \mathbb{T}_{ST}^\times : \mathbf{C} \times \mathbf{C}^{op} \rightarrow \mathbf{C}$ , and  $K_Y(X) = \text{STO}_Y X$  (hence  $K : \mathbf{C}^2 \rightarrow \mathbf{C}$ ). We thus have for all  $Y$  that  $L_Y$  is left adjoint to  $R_Y$ , and that  $K_Y$  is a monad.

Furthermore, we observe the following:

LEMMA C.26. For any  $B \in \mathbf{C}$  and  $A \in K_B$ -alg,  $R_B A$  has a canonical  $\Sigma$ -algebra structure given by

$$\Sigma R_B A \xrightarrow{\tilde{h}^\circ_{R_B A, B}} R_B K_B L_B R_B A \xrightarrow{R_B K_B \varepsilon_B A} R_B K_B A \rightarrow R_B A.$$

This defines a functor  $K_B$ -alg  $\rightarrow \Sigma$ -alg over  $R_B$ , for all  $B$ .

PROOF. Functoriality is straightforward.  $\square$

Unfolding the definition, the family  $\tilde{h}^\circ_{X, Y}$  is in fact constructed in this way, with  $A = K_Y L_Y X$ , whose  $K_Y$ -algebra structure is given by  $\mu_{L_Y X}^{K_Y}$ .

Let us prove that the functorial assignment of Lemma C.26 is in fact also functorial in  $B$ , in the following sense.

LEMMA C.27. For any morphism  $f : A \rightarrow B$  and object  $Z$ , equipping  $K_B Z$  with the  $K_A$ -algebra structure  $K_A K_B Z \xrightarrow{K_f K_B Z} K_B K_B Z \xrightarrow{\mu_Z^{K_B}} K_B Z$ , the morphism

$$R_B K_B Z \xrightarrow{R_f K_B Z} R_A K_B Z$$

is a  $\Sigma$ -algebra morphism.

PROOF. By commutativity of the following diagram

$$\begin{array}{ccc}
 & \Sigma R_f K_B Z & \rightarrow \Sigma R_A K_B Z \\
 & \searrow & \downarrow \tilde{h}^\circ_{R_A K_B Z, A} \\
 \Sigma R_B K_B Z & \xrightarrow{\tilde{h}^\circ_{R_B K_B Z, A}} & R_A K_A L_A R_B K_B Z \xrightarrow{R_A K_A L_B R_f K_B Z} R_A K_A L_A R_A K_B Z \\
 \downarrow \tilde{h}^\circ_{R_B K_B Z, B} & \text{(Lemma C.20)} & \downarrow R_A K_f L_f R_B K_B Z & \text{(Corollary C.17)} & \downarrow R_A K_f \varepsilon_A K_B Z \\
 R_B K_B L_B R_B K_B Z & \xrightarrow{R_f K_B L_B R_B K_B Z} & R_A K_B L_B R_B K_B Z & \xrightarrow{R_A K_B \varepsilon_B K_B Z} & R_A K_B K_B Z \\
 \downarrow R_B K_B \varepsilon_B K_B Z & \text{(interchange)} & \downarrow R_f K_B K_B Z & & \downarrow R_A \mu_Z^{K_B} \\
 R_B K_B K_B Z & \xrightarrow{R_f K_B K_B Z} & R_A K_B K_B Z & & R_A K_B Z \\
 \downarrow R_B \mu_Z^{K_A} & \text{(interchange)} & \downarrow R_f K_B Z & & \downarrow R_f K_B Z \\
 R_B K_B Z & \xrightarrow{R_f K_B Z} & R_A K_B Z & & R_A K_B Z
 \end{array}$$

Moreover, we have:

LEMMA C.28. The family  $\tilde{h}^\circ_{X, Y}$  is extranatural in  $Y$ , i.e., the following square commutes for all  $X$  and  $f : Y \rightarrow Z$ .

$$\begin{array}{ccc}
 SX & \xrightarrow{\tilde{h}^\circ_{X, Y}} & \mathbb{T}_{STY}^\times \text{STO}_Y \Gamma_{STY}^\circ X \\
 \tilde{h}^\circ_{X, Z} \downarrow & & \downarrow \mathbb{T}_{STY}^\times \text{STO}_f \Gamma_{STY}^\circ X \\
 \mathbb{T}_{STZ}^\times \text{STO}_Z \Gamma_{STZ}^\circ X & \xrightarrow{\mathbb{T}_{STf}^\times \text{STO}_Z \Gamma_{STZ}^\circ X} & \mathbb{T}_{STY}^\times \text{STO}_Z \Gamma_{STZ}^\circ X
 \end{array}$$

For proving this, we will need the following result.

LEMMA C.29. For any monad  $S$  on a category with binary coproducts, the following diagram commutes for all objects  $C$  and  $D$ .

$$\begin{array}{ccc}
 & SA + SB & \\
 \uparrow^S & & s \uparrow \\
 S(SA + B) & & S(A + SB) \\
 s(s \uparrow) \downarrow & & \downarrow S \uparrow^S \\
 SS(A + B) & & SS(A + B) \\
 \mu^S \searrow & & \swarrow \mu^S \\
 & S(A + B) &
 \end{array}$$

PROOF. By diagram chasing, termwise, but doing only one term as the other follows by symmetry.

$$\begin{array}{ccccc}
 SA & \xrightarrow{in_1} & SA + SB & \xrightarrow{SA + \eta_{SB}^S} & SA + SSB \\
 \downarrow in_1 & \nearrow \eta^S & \searrow S\eta^S & \searrow Sin_1 & \downarrow [Sin_1, Sin_2] \\
 SA + SB & & SSA & & S(A + SB) \\
 \eta_{SA+SB}^S \downarrow & & \searrow \mu^S & & \downarrow S(\eta_{A+SB}^S) \\
 SSA + SB & & SA & & S(SA + SB) \\
 [Sin_1, Sin_2] \downarrow & \nearrow Sin_1 & \searrow SSin_1 & \searrow Sin_1 & \downarrow S[Sin_1, Sin_2] \\
 S(SA + B) & \xrightarrow{S(SA + \eta_B^S)} & S(SA + SB) & \xrightarrow{S[Sin_1, Sin_2]} & SS(A + B) \\
 & & & & \downarrow \mu^S \\
 & & & & S(A + B)
 \end{array} \quad \square$$

PROOF OF LEMMA C.28. Using the notation, we want to prove that the following square commutes.

$$\begin{array}{ccc}
 SX & \xrightarrow{\tilde{h}_{X,Y}^\omega} & R_Y K_Y L_Y X \\
 \tilde{h}_{X,Z}^\omega \downarrow & & \downarrow R_Y K_f L_f X \\
 R_Z K_Z L_Z X & \xrightarrow{R_f K_Z L_Z X} & R_Y K_Z L_Z X
 \end{array}$$

We first observe that both morphisms have the same restriction to  $\Sigma X$ , by definition of  $\tilde{h}^\omega$  and Lemma C.20. Thus, by

- universal property of  $SX$  as a free  $\Sigma$ -algebra over  $X$ , and
- the fact that for any morphism  $\alpha: T_1 \rightarrow T_2$  of monads, any  $\alpha_X: T_1 X \rightarrow T_2 X$  is a  $T_1$ -algebra morphism, equipping  $T_2 X$  with the  $T_1$ -algebra structure given by

$$T_1 T_2 X \xrightarrow{\alpha_{T_2 X}} T_2 T_2 X \xrightarrow{\mu^{T_2 X}} T_2 X,$$

so that each  $\tilde{h}_{X,Y}^\omega$  is an  $S$ -algebra morphism, or equivalently a  $\Sigma$ -algebra morphism,

it suffices to equip  $R_Y K_Z L_Z X$  with  $\Sigma$ -algebra structure and show that the bottom and right morphisms above, i.e.,  $R_f K_Z L_Z X$  and  $R_Y K_f L_f X$ , are  $\Sigma$ -algebra morphisms. For the  $\Sigma$ -algebra structure on  $R_Y K_Z L_Z X$ , we apply Lemma C.26 with the following  $K_Y$ -algebra structure on  $K_Z L_Z X$ :

$$K_Y K_Z L_Z X \xrightarrow{K_f K_Z L_Z X} K_Z K_Z L_Z X \xrightarrow{\mu^{K_Z}} K_Z L_Z X.$$

The bottom morphism then lifts to  $\Sigma$ -alg by Lemma C.27. Finally, the fact that the right-hand morphism  $R_Y K_f L_f X$  is a  $\Sigma$ -algebra morphism will thus follow from  $K_f L_f X$  being a  $K_Y$ -algebra morphism, which in turn follows by chasing the following diagram.

$$\begin{array}{ccc} K_Y K_Y L_Y X & \xrightarrow{K_Y K_f L_f X} & K_Y K_Z L_Z X \\ \downarrow \mu^{K_Y} & & \downarrow K_f K_Z L_Z X \\ K_Y L_Y X & \xrightarrow{K_f L_Y X} & K_Z L_Y X \xrightarrow{K_Z L_f X} K_Z L_Z X \\ & & \downarrow \mu^{K_Z} \\ & & K_Z K_Z L_Z X \end{array} \quad \square$$

LEMMA C.30. *The family  $\tilde{h}^{\circ\omega}$  satisfies the following laws.*

$$\begin{array}{ccc} SX & \xrightarrow{S\eta_X^T} & STX \\ \tilde{h}^{\circ\omega}_{X,Y} \downarrow & & \downarrow ST\alpha_{X,Y}^{\bullet ST} \\ T_{STY}^{\times} STO_Y \Gamma_{STY}^{\circ} X & \xrightarrow{\pi_2} & STO_Y \Gamma_{STY}^{\circ} X \\ \\ SX & \xrightarrow{\tilde{h}^{\circ\omega}_{X,Y}} & T_{STY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ \tilde{h}^{\circ\omega}_{X,SY} \downarrow & & \downarrow T_{\mu_{SY}^S}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ T_{STSY}^{\times} ST\Gamma_{ST|SY}^{\bullet} X & \xrightarrow{T_{STSY}^{\times} \uparrow_{X,Y}^S} & T_{STSY}^{\times} STS\Gamma_{STS|Y}^{\bullet} X \xrightarrow{T_{STSY}^{\times} (S\delta; \mu^{ST})_{S\delta; \mu^{ST}|Y}^{\bullet}} & T_{STSY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ & & \downarrow T_{S\delta_Y}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ \Sigma X & \xrightarrow{\eta_{\Sigma, X}} & SX \\ & & \downarrow \tilde{h}^{\circ\omega}_{X,Y} \\ & & T_{STY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ & & \downarrow \pi_1 \\ & & T_{STY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ & \searrow \tilde{h}_{X,Y} & \downarrow T_{\eta_Y^{ST}}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\ & & T_Y ST\Gamma_{ST|Y}^{\bullet} X \end{array}$$

$$\begin{array}{ccc}
 SX & \xrightarrow{\tilde{h}^{\omega}_{X,TY}} & \mathbb{T}_{STTY}^{\times} ST\Gamma_{ST|TY}^{\bullet} X \\
 \downarrow \tilde{h}^{\omega}_{X,Y} & & \downarrow \mathbb{T}_{STTY}^{\times} ST\uparrow^T \\
 & & \mathbb{T}_{STTY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X \\
 & & \downarrow \mathbb{T}_{STTY}^{\times} S\mu^T \Gamma_{S\mu^T|Y}^{\bullet} X \\
 \mathbb{T}_{STY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X & \xrightarrow{\mathbb{T}_{S\mu^T}^{\times} ST\Gamma_{ST|Y}^{\bullet} X} & \mathbb{T}_{STTY}^{\times} ST\Gamma_{ST|Y}^{\bullet} X
 \end{array}$$

PROOF. The third law is direct by Lemma C.21. For the first two, we proceed similarly to the proof of Lemma C.28, by showing that all the needed morphisms are  $\Sigma$ -algebra morphisms. For the first diagram, the top and right-hand morphisms are free  $\Sigma$ -algebra morphisms, so we reduce to proving that  $\pi_2$  is a  $\Sigma$ -algebra morphism. This holds by commutativity of the diagram in Figure 4 (recalling Notation 7), where the question-marked polygon commutes because the following diagram does, for all  $A$ .

$$\begin{array}{ccc}
 R_Y A & \xrightarrow{\pi_2} & A \\
 \downarrow in_2 & \nearrow \varepsilon_Y A & \downarrow in_1 \\
 L_Y R_Y A & & O_Y A \\
 \downarrow in_1 & & \downarrow O_Y \varepsilon_Y A \\
 O_Y L_Y R_Y A & \xrightarrow{O_Y \varepsilon_Y A} & O_Y A
 \end{array}$$

$\alpha_{R_Y L_Y}^{O_Y L_Y}$  (curved arrow from  $R_Y A$  to  $O_Y L_Y R_Y A$ )

The top triangle commutes by construction of

$$\varepsilon_Y A: \Gamma_{STY}^{\circ} \mathbb{T}_{STY}^{\times} A \rightarrow A:$$

the domain  $\Gamma_{STY}^{\circ} \mathbb{T}_{STY}^{\times} A$  is the coproduct

$$\Gamma_{STY}(\mathbb{T}_{STY}(A) \times A) \quad + \quad \mathbb{T}_{STY}(A) \times A,$$

and the second component of  $\varepsilon_Y A$  is  $\pi_2: \mathbb{T}_{STY}(A) \times A \rightarrow A$ .

Let us now turn to the second diagram. We start by equipping the codomain with  $\Sigma$ -algebra structure, by applying Lemma C.26 with  $B = SY$  and  $A = ST\Gamma_{ST|Y}^{\bullet} X = K_Y \Gamma_{STY}^{\circ} X$ , whose  $K_{SY}$ -algebra structure is given by either of the following three equal composites.



$$\begin{array}{ccc}
 K_{SY}K_YL_YX & K_{SY}K_YL_YX & K_{SY}K_YL_YX \\
 \parallel & \parallel & \parallel \\
 ST(STO_YL_YX + SY) & ST(STO_YL_YX + SY) & ST(STO_YL_YX + SY) \\
 \downarrow ST\uparrow^S & \downarrow ST^{(ST\uparrow)} & \downarrow ST^{(ST\uparrow)} \\
 STSO_YSTO_YL_YX & STST(O_YL_YX + SY) & STST(O_YL_YX + SY) \\
 \downarrow (S\delta; \mu^{ST}) & \downarrow STST\uparrow^S & \downarrow \mu^{ST} \\
 STO_YSTO_YL_YX & STSTS(O_YL_YX + Y) & ST(O_YL_YX + SY) \\
 \downarrow \mu^{K_Y} & \downarrow S\delta; \mu^S & \downarrow ST\uparrow^S \\
 STO_YL_YX & STTS(L_YX + Y + Y) & STS(L_YX + Y + Y) \\
 \parallel & \downarrow S\mu^T & \downarrow (S\delta; \mu^{ST})(L_YX + [Y, Y]) \\
 K_YL_YX & STS(L_YX + Y + Y) & ST(L_YX + Y) \\
 & \downarrow S\delta(L_YX + [Y, Y]) & \parallel \\
 & SST(L_YX + Y) & K_YL_YX \\
 & \downarrow \mu^S & \\
 & ST(L_YX + Y) & \\
 & \parallel & \\
 & K_YL_YX & 
 \end{array}$$

LEMMA C.31. *The above  $K_{SY}$ -algebra structures are indeed equal.*

PROOF. The latter equivalence is clear. For the former, we proceed by diagram chasing as in Figure 5, where  $A = L_YX$ .  $\square$

Returning to the second law of Lemma C.30, the bottom row is the image by  $\mathbb{T}_{STSY}^\times$  of a morphism that lifts to  $K_{SY}$ -alg, by commutation of the diagram in Figure 6. It thus itself lifts to  $\Sigma$ -alg by Lemma C.26.

Finally, we prove that the right-hand composite is a  $\Sigma$ -algebra morphism by diagram chasing as follows.



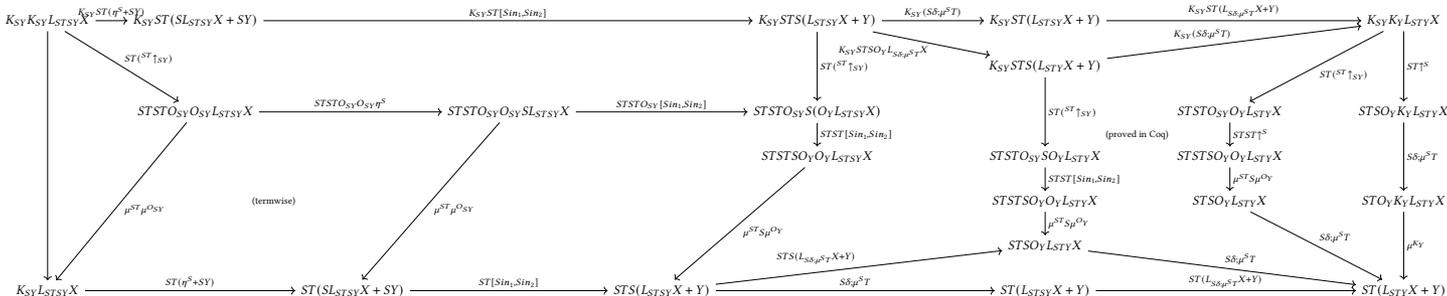


Fig. 6. Lemma C.30: bottom morphism lifts to  $\Sigma$  - alg









LEMMA C.34. The family  $\widetilde{h}^\omega$  satisfies the following coherence laws.

$$\begin{array}{c}
S(X) \xrightarrow{S(\eta_X^T)} ST(X) \\
\alpha_{SX,STY}^{\Gamma^\circ} \downarrow \qquad \qquad \qquad \downarrow ST\alpha_{X,Y}^{\Gamma_{ST}^\circ} \\
\Gamma_{STY}^\circ(S(X)) \xrightarrow{\widetilde{h}_{X,Y}^\omega} ST(\Gamma_{ST|Y}^\bullet(X))
\end{array}$$
  

$$\begin{array}{ccc}
\Gamma_{STSY}^\circ SX \xrightarrow{\Gamma_{S\delta_Y}^\circ SX} \Gamma_{SSTY}^\circ SX \xrightarrow{\Gamma_{\mu_{TY}^S}^\circ SX} \Gamma_{STY}^\circ SX & \Gamma_{STY}^\circ X \xrightarrow{\eta_{\Gamma_{STY}^\circ X}^{ST}} ST\Gamma_{STY}^\circ X \\
\widetilde{h}_{X,SY}^\omega \downarrow & & \Gamma_{STY}^\circ \eta_X^S \downarrow \\
ST\Gamma_{ST|SY}^\bullet X \xrightarrow{ST\uparrow_{X,Y}^S} ST\Gamma_{STS|Y}^\bullet X \xrightarrow{(S\delta; \mu^S T)\Gamma_{S\delta; \mu^S T|Y}^\bullet X} ST\Gamma_{ST|Y}^\bullet X & & \Gamma_{STY}^\circ SX \xrightarrow{\widetilde{h}^\omega} ST\Gamma_{ST|Y}^\bullet X \\
& & \downarrow STin_1
\end{array}$$
  

$$\begin{array}{ccc}
\Gamma_{STTY}^\circ SX & \xrightarrow{\widetilde{h}_{X,TY}^\omega} & ST\Gamma_{ST|TY}^\bullet X \\
\Gamma_{S\mu_{TY}^T}^\circ SX \downarrow & & \downarrow ST\uparrow^T \\
\Gamma_{STY}^\circ SX & \xrightarrow{\widetilde{h}_{X,Y}^\omega} & ST\Gamma_{ST|Y}^\bullet X \\
& & \downarrow S\mu^T \Gamma_{S\mu^T|Y}^\bullet X
\end{array}$$
  

$$\begin{array}{ccc}
\Gamma_{STY}^\circ SSX \xrightarrow{\widetilde{h}_{SX,Y}^\omega} STO_Y \Gamma_{STY}^\circ SX \xrightarrow{STO_Y \widetilde{h}_{X,Y}^\omega} STO_Y STO_Y \Gamma_{STY}^\circ X & & \\
\Gamma_{STY}^\circ \mu_X^S \downarrow & & \downarrow \mu_{\Gamma_{STY}^\circ X}^{STO_Y} \\
\Gamma_{STY}^\circ SX \xrightarrow{\widetilde{h}_{X,Y}^\omega} STO_Y \Gamma_{STY}^\circ X & & 
\end{array}$$
  

$$\begin{array}{ccc}
\Gamma_Y \Sigma X \xrightarrow{\Gamma_{\eta_Y^{ST} \eta_{\Sigma, X}}} \Gamma_{STY}^\circ SX \xrightarrow{in_1} \Gamma_{STY}^\circ SX & & \\
h_{X,Y} \searrow & & \swarrow \widetilde{h}_{X,Y}^\omega \\
& ST\Gamma_{ST|Y}^\bullet X & 
\end{array}$$

PROOF. By Lemma C.30, Corollary C.17, and the fact that  $\widetilde{h}_{X,Y}^\omega$  is a monad morphism by construction. E.g., the third law holds iff the transposed morphisms coincide. This gives the following diagram,

$$\begin{array}{ccc}
X \xrightarrow{\eta_Y X} R_Y L_Y X \xrightarrow{R_Y \eta_{L_Y X}^{ST}} R_Y ST L_Y X \\
\downarrow \eta_X^S \qquad \qquad \qquad \downarrow R_Y ST in_1 \\
SX \xrightarrow{\widetilde{h}_{X,Y}^\omega} R_Y K_Y L_Y X
\end{array}$$

which is precisely commutation of  $\widetilde{h}^\omega$  with unit. Similarly, the penultimate law holds iff the transposed morphisms coincide. This gives the following diagram,

$$\begin{array}{ccccc}
 SSX & \xrightarrow{\tilde{h}_{SX,Y}^\omega} & R_Y K_Y L_Y SX & \xrightarrow{R_Y K_Y L_Y \tilde{h}_{X,Y}^\omega} & R_Y K_Y L_Y R_Y K_Y L_Y X \\
 \downarrow \mu_X^S & & & & \downarrow R_Y K_Y \epsilon_Y \\
 & & & & R_Y K_Y K_Y L_Y X \\
 & & & & \downarrow R_Y \mu^{K_Y} \\
 SX & \xrightarrow{\tilde{h}_{X,Y}^\omega} & & & R_Y K_Y L_Y X
 \end{array}$$

which precisely commutation of  $\tilde{h}^\omega$  with multiplication.  $\square$

### C.6 The family $h^{\bullet\omega}$

*Definition C.35.* Let  $h_{X,Y}^{\bullet\omega}: \Gamma_{SY} SX \rightarrow ST\Gamma_{ST|Y}^\bullet X$  denote the composite

$$\Gamma_{SY} SX \xrightarrow{\Gamma_{S\eta_Y^T} SX} \Gamma_{STY} SX \xrightarrow{in_1} \Gamma_{STY}^\circ SX \xrightarrow{\tilde{h}_{X,Y}^\omega} ST\Gamma_{ST|Y}^\bullet X.$$

*LEMMA C.36.* The natural transformation  $h_{X,Y}^{\bullet\omega}: \Gamma_{SY}^\bullet SX \rightarrow ST\Gamma_{ST|Y}^\bullet X$  is natural in both variables.

*PROOF.* This follows straightforwardly from Lemma C.33.  $\square$

*LEMMA C.37.* The natural transformation  $h^{\bullet\omega}$  makes all diagrams below commute, for all  $X, Y \in \mathbf{C}$ .

$$\begin{array}{ccc}
 \Gamma_{SSY} SX & \xrightarrow{\Gamma_{\mu^{SY} SX}} & \Gamma_{SY} SX \\
 h_{X,SY}^{\bullet\omega} \downarrow & & \downarrow h_{X,Y}^{\bullet\omega} \\
 ST\Gamma_{ST|SY}^\bullet X & \xrightarrow{ST\uparrow^S} ST\Gamma_{STS|Y}^\bullet X \xrightarrow{s\delta\Gamma_{S\delta|Y}^\bullet X} SST\Gamma_{SST|Y}^\bullet X \xrightarrow{\mu^{ST\Gamma_{\mu^{ST|Y} X}}} & ST\Gamma_{ST|Y}^\bullet X \\
 \Gamma_{SY} SSX & \xrightarrow{h_{SX,Y}^{\bullet\omega}} ST(\Gamma_{STY} SX + SX + Y) \xrightarrow{ST(h_{X,TY}^{\bullet\omega} + S\eta_X^T + \eta_Y^{ST})} & ST(ST\Gamma_{ST|TY}^\bullet X + STX + STY) \\
 \downarrow \Gamma_{SY} \mu_X^S & & \parallel \\
 & & ST(ST(\Gamma_{STTY} X + X + TY) + STX + STY) \\
 & & \downarrow ST(ST\uparrow^T + STX + STY) \\
 & & ST(STT(\Gamma_{STTY} X + X + Y) + STX + STY) \\
 & & \downarrow ST(S\mu^T(\Gamma_{S\mu^T Y} X + X + Y) + STX + STY) \\
 & & ST(ST(\Gamma_{STY} X + X + Y) + STX + STY) \\
 & & \downarrow ST[STin_1, STin_2, STin_3] \\
 & & STST(\Gamma_{STY} X + X + Y + X + Y) \\
 & & \downarrow \mu^{ST}[in_1, in_2, in_3, in_2, in_3] \\
 & & ST(\Gamma_{STY} X + X + Y) \\
 & & \parallel \\
 \Gamma_{SY} SX & \xrightarrow{h_{X,Y}^{\bullet\omega}} & ST\Gamma_{ST|Y}^\bullet X
 \end{array}$$

$$\begin{array}{ccc}
\Gamma_{SY}X & \xrightarrow{\Gamma_{S\eta_X^T X}} \Gamma_{STY}X & \xrightarrow{in_1} \Gamma_{ST|Y}^\bullet X \\
\downarrow \Gamma_{SY}\eta_X^S & & \downarrow \eta^{ST} \\
\Gamma_{SY}SX & \xrightarrow{h_{X,Y}^{\bullet\omega}} & ST\Gamma_{ST|Y}^\bullet X \\
\Gamma_Y\Sigma X & \xrightarrow{\Gamma_{\eta_Y^S}\eta_{\Sigma,X}} & \Gamma_{SY}SX \\
& \searrow h_{X,Y} & \swarrow h_{X,Y}^{\bullet\omega} \\
& & ST\Gamma_{ST|Y}^\bullet X
\end{array}$$

PROOF. The last diagram is direct by Lemma C.34. The third one follows by chasing as follows,

$$\begin{array}{ccccccc}
\Gamma_{SY}X & \xrightarrow{\Gamma_{S\eta_X^T X}} & \Gamma_{STY}X & \xrightarrow{in_1} & \Gamma_{STY}^\circ X & \xrightarrow{in_1} & \Gamma_{ST|Y}^\bullet X \\
\downarrow \Gamma_{SY}\eta_X^S & & \downarrow \Gamma_{STY}\eta_X^S & & \downarrow \Gamma_{STY}^\circ\eta_X^S & \searrow \eta^{ST} & \downarrow \eta^{ST} \\
\Gamma_{SY}SX & \xrightarrow{\Gamma_{S\eta_Y^T SX}} & \Gamma_{STY}SX & \xrightarrow{in_1} & \Gamma_{STY}^\circ SX & \xrightarrow{\widetilde{h}_{X,Y}^{\bullet\omega}} & ST\Gamma_{ST|Y}^\bullet X \\
& & & & & \nearrow STin_1 & \\
& & & & & & ST\Gamma_{STY}^\circ X
\end{array}$$

using Lemma C.34. The first statement follows by chasing the next diagram.

$$\begin{array}{ccccccc}
\Gamma_{SSY}SX & \xrightarrow{\Gamma_{\mu_Y^S} SX} & & & \Gamma_{SY}SX & & \\
\downarrow \Gamma_{S\eta_{SY}^T SX} & \searrow \Gamma_{SS\eta_Y^T SX} & & & \downarrow \Gamma_{S\eta_Y^T SX} & & \\
\Gamma_{STSY}SX & \xrightarrow{\Gamma_{S\delta_Y} SX} & \Gamma_{SSTY}SX & \xrightarrow{\Gamma_{\mu_{TY}^S} SX} & \Gamma_{STY}SX & & \\
\downarrow in_1 & & \downarrow \Gamma_{S\delta_Y}^\circ SX & & \downarrow \Gamma_{\mu_{TY}^S}^\circ SX & & \downarrow in_1 \\
\Gamma_{STSY}^\circ SX & \xrightarrow{\Gamma_{S\delta_Y}^\circ SX} & \Gamma_{SSTY}^\circ SX & \xrightarrow{\Gamma_{\mu_{TY}^S}^\circ SX} & \Gamma_{STY}^\circ SX & & \\
\downarrow \widetilde{h}_{X,SY}^{\bullet\omega} & & & & \downarrow \widetilde{h}_{X,SY}^{\bullet\omega} & & \\
ST\Gamma_{ST|SY}^\bullet X & \xrightarrow{ST\uparrow^S} & STS\Gamma_{ST|Y}^\bullet X & \xrightarrow{S\delta_{\Gamma_{S\delta_Y}^\circ X}} & SST\Gamma_{SST|Y}^\bullet X & \xrightarrow{\mu^S\Gamma_{\mu^S T|Y}^\bullet X} & ST\Gamma_{ST|Y}^\bullet X
\end{array}$$

(Lemma C.34)

For the second statement, we first reduce to the rightmost subdiagram in Figure 9, which further reduces as in Figure 10.  $\square$

### C.7 The incremental lifting

PROPOSITION C.38. For any object  $X$  and  $(T \oplus T')$ -algebra structure

$$TX \xrightarrow{a} X \xleftarrow{b} \Gamma_S X,$$

the  $\Gamma_S$ -algebra structure on  $SX$  given by

$$\Gamma_{SSX}SX \xrightarrow{\Gamma_{\mu_X^S} SX} \Gamma_{SX}SX \xrightarrow{h_{X,X}^{\bullet\omega}} ST\Gamma_{ST|X}^\bullet X \xrightarrow{ST(\Gamma_{SaX+[X,X]})} ST(\Gamma_{SX}X + X) \xrightarrow{ST[b,X]} STX \xrightarrow{Sa} SX$$

defines an incremental lifting of  $S$  to  $T'$ -Alg along  $\delta$ .

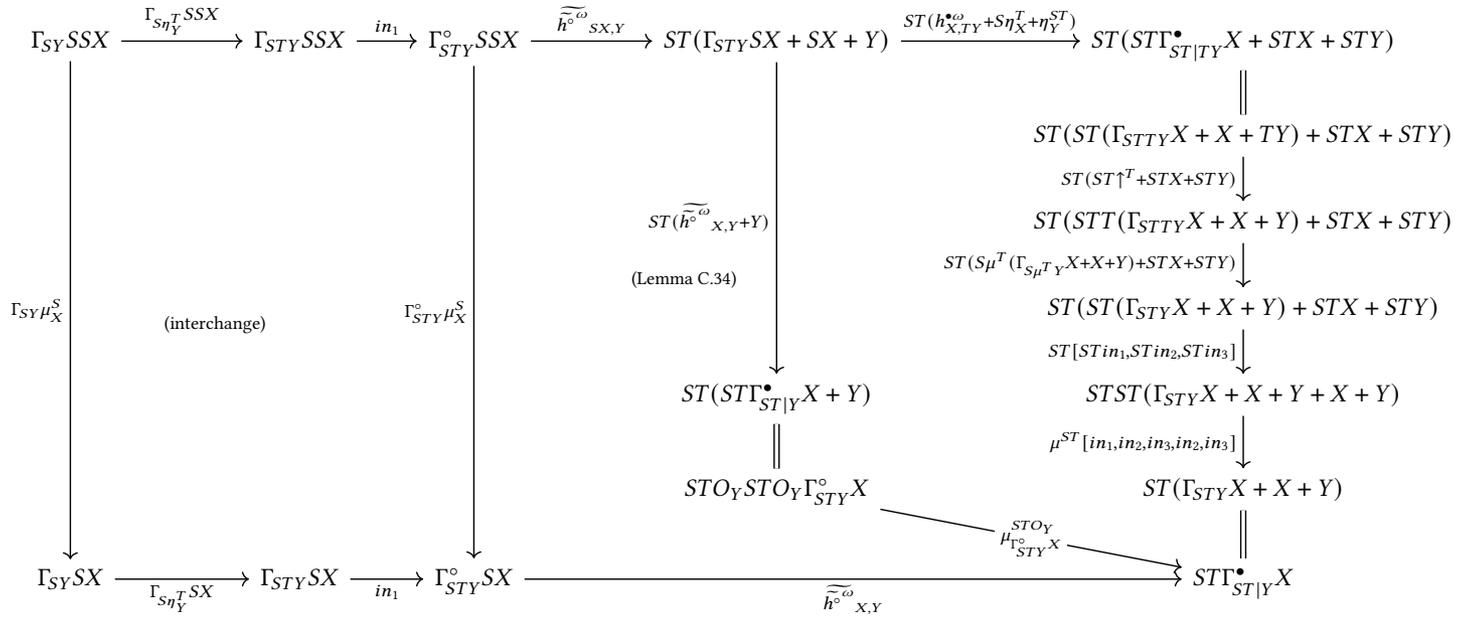
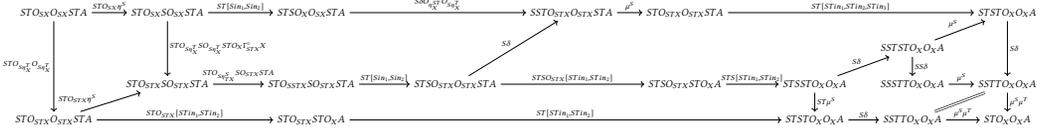


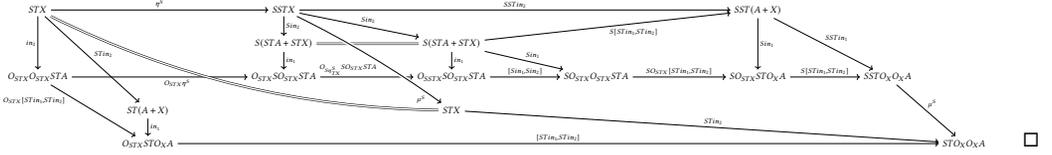
Fig. 9. Proof of Lemma C.37



PROOF. The given composite readily equips  $SX$  with  $(T \oplus T')$ -algebra structure by Lemma 4.7 and the isomorphism  $\Gamma_S - \text{alg} \cong T' - \text{Alg}$ . It remains to verify that the unit and multiplication are  $\Gamma_S$ -algebra morphisms. The former follows easily by Lemma C.37, and the latter by Figure 11. There, the middle, unlabelled polygon is chased as follows,



where the middle, bottom polygon is the image by  $ST$  of a diagram whose domain is a coproduct, and whose commutativity may be checked termwise. E.g., the second term is chased as follows.



It remains to show:

PROPOSITION C.39. *The distributive law  $d_{\delta}: (T \oplus T')S \rightarrow S(T \oplus T')$  satisfies (5), which we repeat here for convenience.*

$$\begin{array}{ccc}
 \Gamma_X \Sigma X & \xrightarrow{d_{X,X}} & ST(\Gamma_{STX} X + X + X) \\
 \downarrow \Gamma_{\eta_X}^S \eta_{\Sigma, X} & & \downarrow ST(\Gamma_{STX} \eta_X^T + [X, X]) \\
 \Gamma_{SX} SX & & ST(\Gamma_{STX} TX + X) \\
 \downarrow \Gamma_{\eta_{SX}}^S SX & & \downarrow ST(\eta_{\Gamma_S, TX+X}) \\
 \Gamma_{SSX} SX & & ST(T' TX + X) \\
 \downarrow \eta_{\Gamma_S, SX} & & \downarrow ST(in_2 in_1 X + X) \\
 T' SX & & ST((T \oplus T')(T \oplus T')X + X) \\
 \downarrow in_2 SX & & \downarrow Sin_1[\mu_X^{T \oplus T'}, \eta^{T \oplus T'} X] \\
 (T \oplus T') SX & \xrightarrow{d_{\delta} X} & S(T \oplus T')(T \oplus T')X \\
 & & \downarrow S\mu_X^{T \oplus T'}
 \end{array}$$

We need the following intermediate result.

LEMMA C.40. *For any object  $Y$ , the following diagram commutes.*

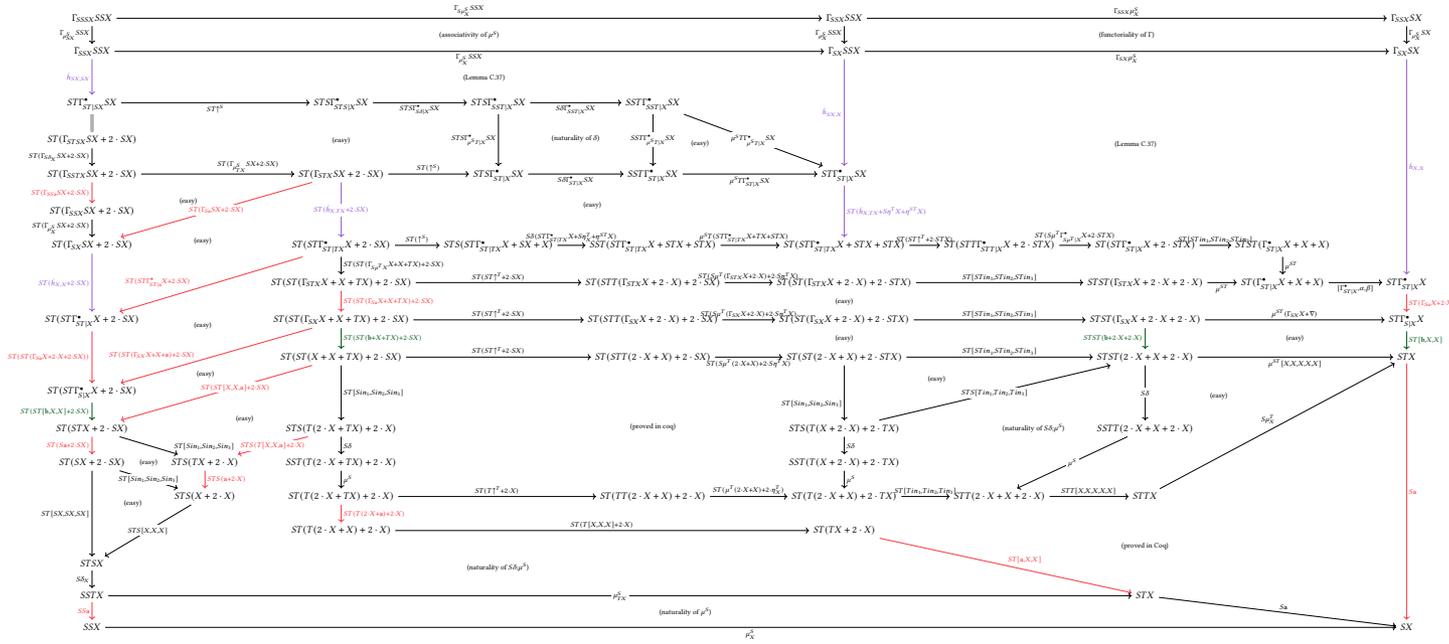


Fig. 11. Multiplication is an algebra morphism

A unified treatment of structural definitions on syntax

$$\begin{array}{ccc}
\Gamma_{SSY}SY & \xrightarrow{\Gamma_{\mu^S Y}SY} & \Gamma_{SY}SY & \xrightarrow{d_{Y,Y}^{\bullet\omega}} & ST(\Gamma_{STY}Y + Y + Y) \\
\eta_{\Gamma_S,SY} \downarrow & & & & \downarrow ST(\Gamma_{STY}\eta_Y^T + [Y,Y]) \\
T'SY & & & & ST(\Gamma_{STY}TY + Y) \\
& & & & \downarrow ST(\eta_{\Gamma_S,TY+Y}) \\
& & & & ST(T'TY + Y) \\
in_2SY \downarrow & & & & \downarrow Sin_1[(\mu_Y^{T\oplus T'} \circ in_2 in_1), \eta_Y^{T\oplus T'}] \\
(T \oplus T')SY & \xrightarrow{d_{\delta Y}} & S(T \oplus T')Y & & S(T \oplus T')(T \oplus T')Y \\
& & & & \downarrow S\mu_Y^{T\oplus T'} \\
& & & & S(T \oplus T')Y
\end{array}$$

PROOF. Up to some easy rewriting of the right-hand composite, this is proved in Figure 12.  $\square$

PROOF OF PROPOSITION C.39. By diagram chasing, as follows.

$$\begin{array}{ccc}
\Gamma_X \Sigma X & \xrightarrow{d_{X,X}} & ST(\Gamma_{STX}X + X + X) \\
\Gamma_{\eta^S} \eta_{\Sigma,X} \downarrow & \searrow d_{X,X}^{\bullet\omega} & \downarrow ST(\eta_{\Gamma_S} \eta_X^T + [X,X]) \\
\Gamma_{SX} SX & \xrightarrow{d_{X,X}^{\bullet\omega}} & ST(T'TX + X) \\
\Gamma_{\eta_{SX}^S} SX \downarrow & \searrow d_{X,X}^{\bullet\omega} & \downarrow ST(in_2 in_1 + X) \\
\Gamma_{SSX} SX & \xrightarrow{\Gamma_{\mu_X^S} SX} & \Gamma_{SX} SX & \searrow d_{X,X}^{\bullet\omega} & ST((T \oplus T')(T \oplus T')X + X) \\
\eta_{\Gamma_S, SX} \downarrow & & \text{(Lemma C.40)} & & \downarrow Sin_1[\mu_X^{T\oplus T'}, \eta_X^{T\oplus T'}] \\
T'SX & & & & S(T \oplus T')(T \oplus T')X \\
in_2 \downarrow & & & & \downarrow S\mu_X^{T\oplus T'} \\
(T \oplus T')SX & \xrightarrow{d_{\delta X}} & S(T \oplus T')X & & S(T \oplus T')X
\end{array}$$

## D PROOF OF THEOREM 4.13

We fix a given incremental structural law

$$d: \Gamma_Y(\Sigma(X)) \rightarrow ST(\Gamma_{STY}(X) + X + Y)$$

over  $\delta: TS \rightarrow ST$ , and let  $T' = \Gamma_S^*$ .

Given any distributive law, algebras for the composite monad admit the following well-known characterisation.

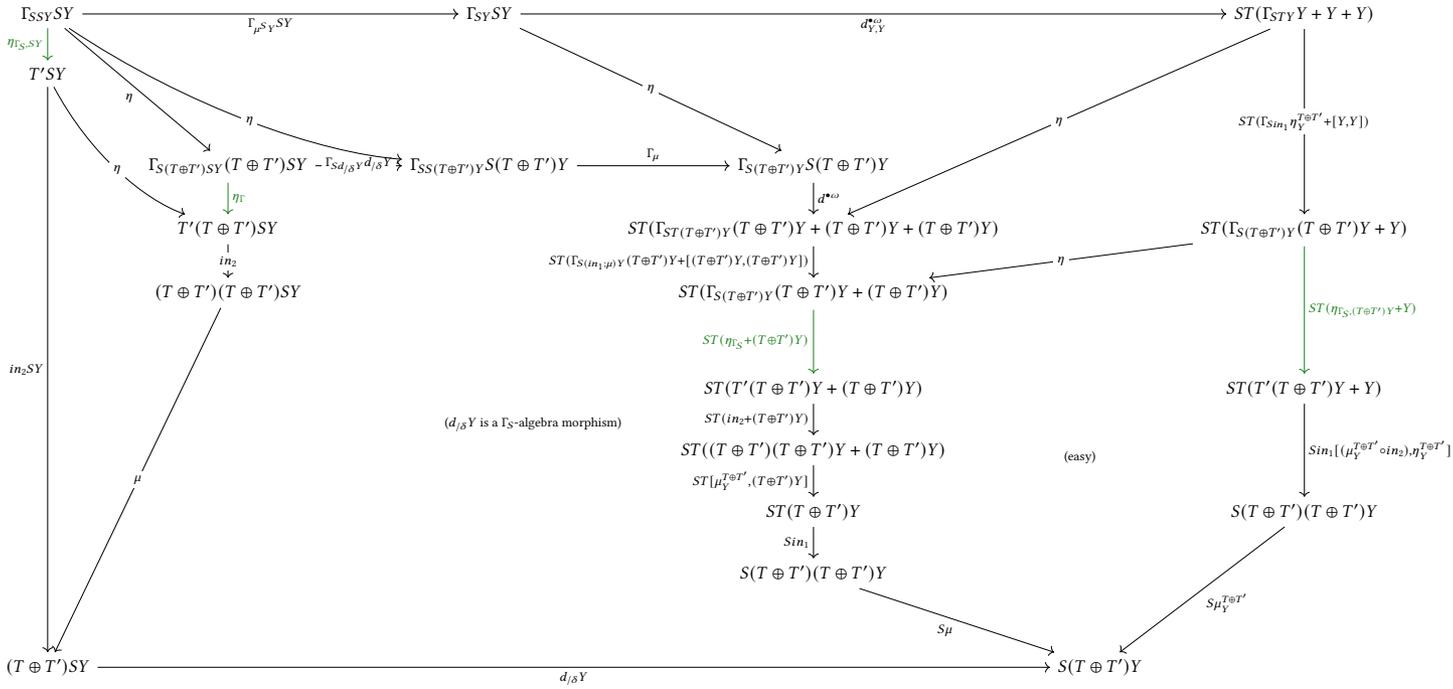


Fig. 12. Proof of Lemma C.40

*Definition D.1.* Given any monad distributive law  $\delta: RS \rightarrow SR$ , a  $\delta$ -**algebra** is an object  $X \in \mathbf{C}$  of the underlying category, equipped with  $S$ - and  $R$ -algebra structures, say  $\mathbf{a}: SX \rightarrow X$  and  $\mathbf{b}: RX \rightarrow X$ , satisfying the following law.

$$\begin{array}{ccc}
 RSX & \xrightarrow{\delta_X} & SRX \\
 \text{Ra} \downarrow & & \downarrow \text{Sb} \\
 RX & & SX \\
 & \searrow \mathbf{b} & \swarrow \mathbf{a} \\
 & X & 
 \end{array} \tag{13}$$

A  $\delta$ -**algebra morphism**  $X \rightarrow Y$  is a morphism between underlying objects which is both an  $S$ - and  $R$ -algebra morphism.

We let  $\delta$ -**Alg** denote the category of  $\delta$ -algebras and morphisms between them.

LEMMA D.2 ([BECK 1969, §2]). *Given any monad distributive law  $\delta: RS \rightarrow SR$ , and  $SR$ -algebra  $\mathbf{x}: SRX \rightarrow X$ , the derived  $S$ - and  $R$ -algebra structures*

$$SX \xrightarrow{S\eta_X^T} SRX \xrightarrow{\mathbf{x}} X \qquad RX \xrightarrow{\eta_{RX}^S} SRX \xrightarrow{\mathbf{x}} X$$

*equip  $X$  with  $\delta$ -algebra structure. Furthermore, this underlies an isomorphism  $SR\text{-Alg} \rightarrow \delta\text{-Alg}$  of categories over  $\mathbf{C}$ .*

In our situation, applying this to the distributive law  $d_{/\delta}: (T \oplus T')S \rightarrow S(T \oplus T')$  from Theorem 4.10, we obtain:

COROLLARY D.3. *An  $S(T \oplus T')$ -algebra is equivalently an  $S$ -algebra  $\mathbf{a}: SX \rightarrow X$ , equipped with  $(T \oplus T')$ -algebra structure satisfying the pentagon (13) with  $R = T \oplus T'$ .*

But we can say more, by Corollary B.3:

LEMMA D.4. *An  $S(T \oplus T')$ -algebra is equivalently an object  $X$ , equipped with morphisms*

$$\mathbf{a}: SX \rightarrow X \qquad \mathbf{b}: TX \rightarrow X \qquad \mathbf{c}: \Gamma(X, SX) \rightarrow X,$$

*the first two of which are monad algebra structures, satisfying the pentagon (13) with  $R = T$ , together with the following diagram,*

$$\begin{array}{ccccc}
 \Gamma_{SSX}SX & \xrightarrow{\Gamma_{\mathbf{S}\mathbf{a}\mathbf{a}}} & & & \Gamma_{SX}X \\
 \downarrow \Gamma_{\mu_X^S} & & & & \downarrow \mathbf{c} \\
 \Gamma_{SX}SX & & (d') & & \\
 \downarrow d_{X,X}^\omega & & & & \\
 ST\Gamma_{ST|X}^\bullet X & \xrightarrow{ST(\mathbf{b}\mathbf{c})} & STX & \xrightarrow{\mathbf{Sb}} & SX & \xrightarrow{\mathbf{a}} & X
 \end{array}$$

*recalling  $\mathbf{b} \triangleright \mathbf{c}$  from Definition 4.11.*

PROOF. The pentagon (13) with  $R = T \oplus T'$  equivalently states that the  $S$ -algebra structure  $\mathbf{a}: SX \rightarrow X$  is a morphism of  $(T \oplus T')$ -algebras. But by Lemma 4.7 this is equivalent to being both a  $T$ - and  $\Gamma_S\Delta$ -algebra morphism. The former is taken care of by (13) with  $R = T$ , the latter by the given diagram.  $\square$

It remains to show that, given an object  $X$  equipped with  $S$ - and  $T$ -algebra structures  $\mathbf{a}: SX \rightarrow X$  and  $\mathbf{b}: TX \rightarrow X$  satisfying (d1), the following are equivalent:

- a morphism  $\mathbf{c}: \Gamma(X, X) \rightarrow X$  satisfying (d2), and

- a morphism  $c: \Gamma(X, SX) \rightarrow X$  satisfying (d2');

and furthermore the notions of morphisms agree.

For any  $c: \Gamma(X, X) \rightarrow X$  satisfying (d2), we define  $\check{c}$  to be the following composite

$$\Gamma(X, SX) \xrightarrow{\Gamma(X,a)} \Gamma(X, X) \xrightarrow{c} X,$$

and conversely for any  $c: \Gamma(X, SX) \rightarrow X$  satisfying (d2'), we define  $\hat{c}$  to be

$$\Gamma(X, X) \xrightarrow{\Gamma(X,\eta_X^S)} \Gamma(X, SX) \xrightarrow{c} X.$$

It thus suffices to prove:

- the assignments  $c \mapsto \check{c}$  and  $c \mapsto \hat{c}$  are mutually inverse,
- for any  $c: \Gamma(X, X) \rightarrow X$  satisfying (d2),  $\check{c}$  satisfies (d2'), and conversely
- for any  $c: \Gamma(X, SX) \rightarrow X$  satisfying (d2'),  $\hat{c}$  satisfies (d2);
- for any  $(d, \delta)$ -algebras  $X$  and  $Y$ , a morphism  $f: X \rightarrow Y$  which is both a  $T$ -algebra morphism, an  $S$ -algebra morphism, and a  $\Gamma$ -algebra morphism is a morphism between the corresponding  $\Gamma_S$ -algebras, and
- conversely for any  $d/\delta$ -algebras  $X$  and  $Y$ , a morphism  $f: X \rightarrow Y$  which is both a  $T$ -algebra morphism, an  $S$ -algebra morphism, and a  $\Gamma_S$ -algebra morphism is a morphism between the corresponding  $\Gamma$ -algebras.

Statements (d) and (e) follow easily by naturality of  $\eta^S$  and the fact that the considered morphism is an algebra morphism.

Statement (b) follows from commutation of the diagram in Figure 13, and  $d_{X,Y}^{\bullet\omega}$  as in Definition C.35. Subdiagram (A) commutes by chasing as in Figure 14. Subdiagram (A') commutes as

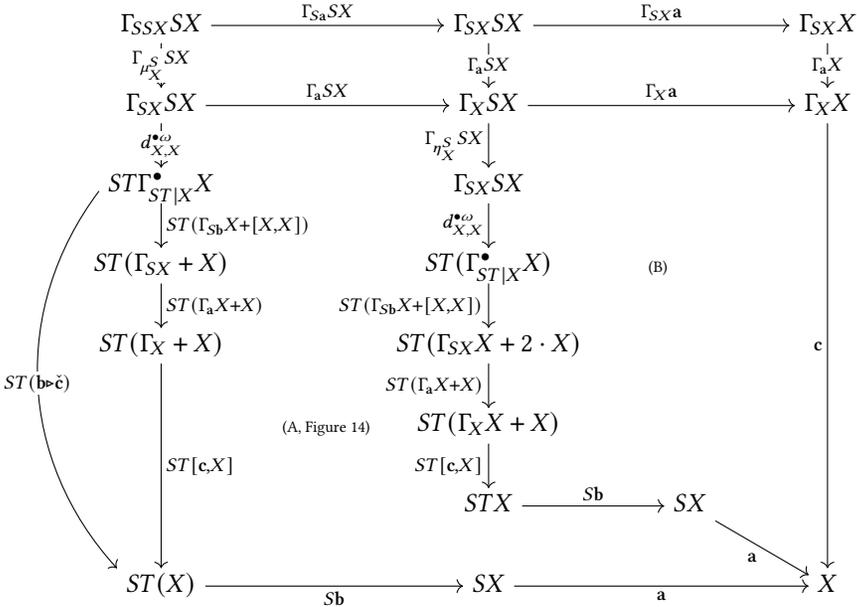


Fig. 13. Main diagram for proving (d2') for  $\check{c}$

shown in Figure 15. Subdiagram (B) commutes by Lemma D.7 below.

Statement (c) follows from commutation of the diagram in Figure 16, where

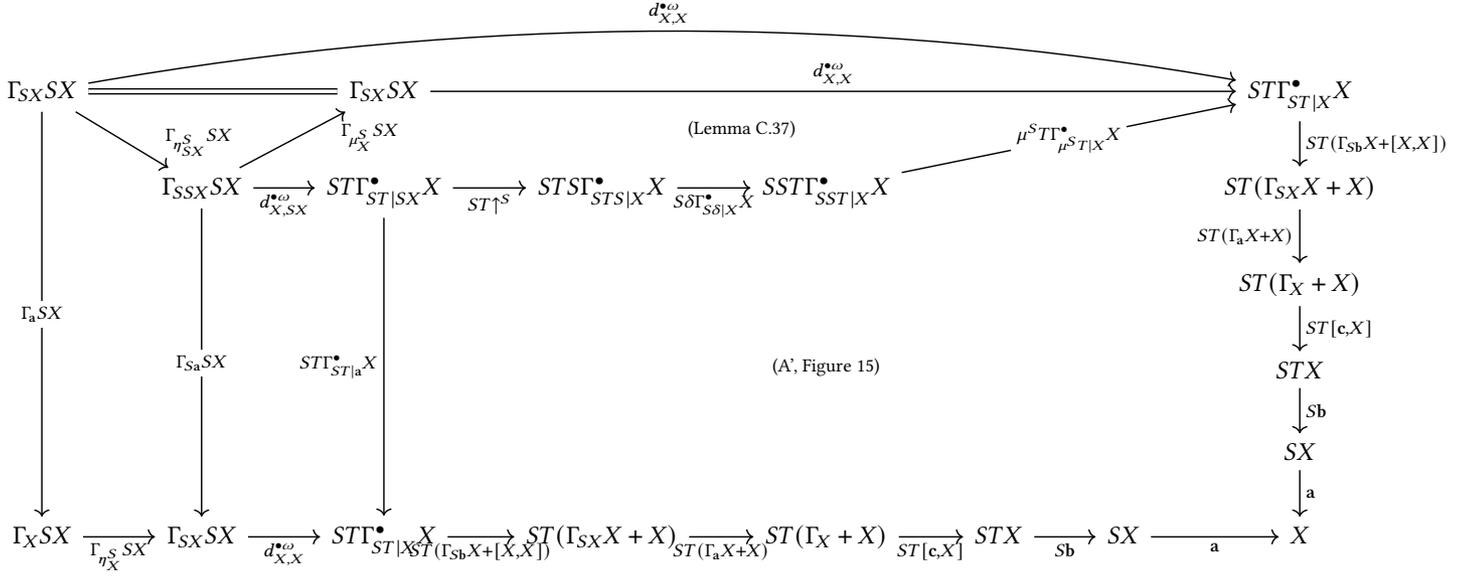


Fig. 14. Chasing subdiagram A

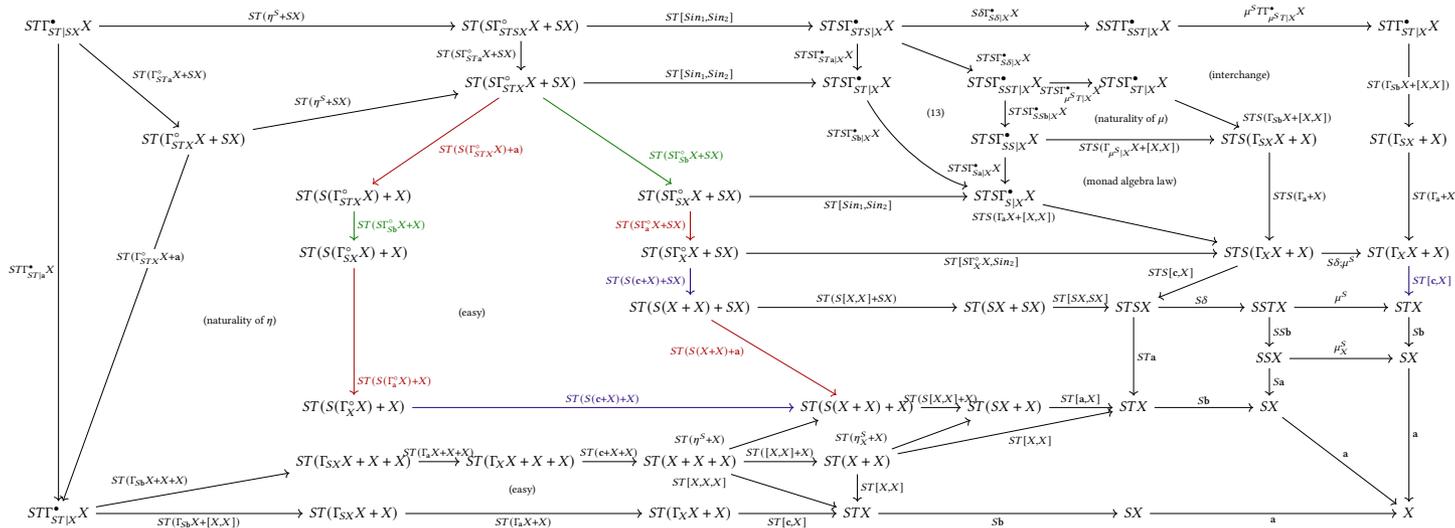


Fig. 15. Chasing subdiagram (A')

- the top part commutes by Lemma C.37 and
- the bottom right part commutes by Lemma D.6 below.

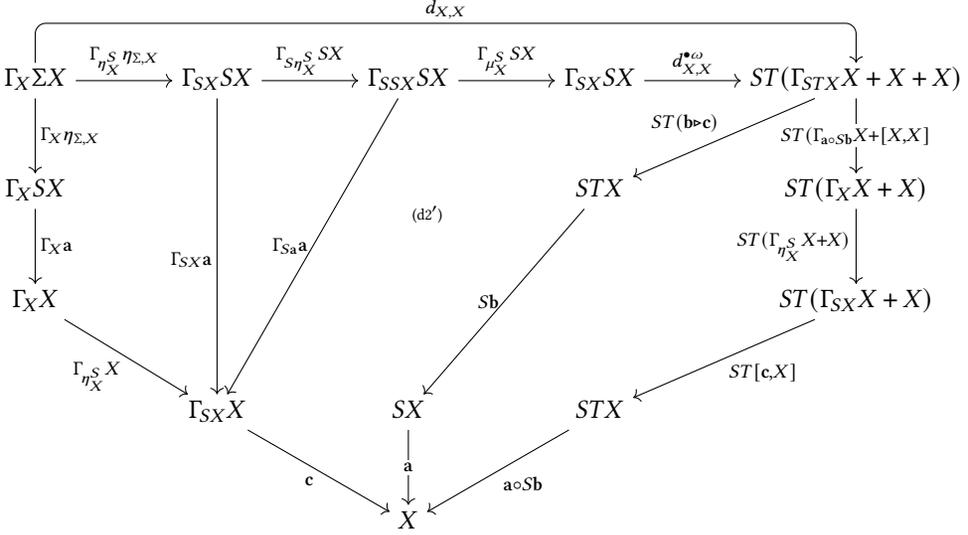


Fig. 16. Diagram for (c)

Finally, (a) follows from Lemma D.5 and D.6 below.

LEMMA D.5. Given an algebra  $SX \xrightarrow{a} X$ , precomposition with morphisms  $\Gamma_{S a}$  and  $\Gamma_{\eta^S}$  induces a bijection (natural in the algebra) between morphisms  $\Gamma_X X \rightarrow X$  and morphisms  $\Gamma_{S X} X \rightarrow X$  making the following diagram commute.

$$\begin{array}{ccccc}
 \Gamma_{S X} X & \xrightarrow{\Gamma_a X} & \Gamma_X X & \xrightarrow{\Gamma_{\eta_X^S} X} & \Gamma_{S X} X \\
 & \searrow c & & \swarrow c & \\
 & & X & & 
 \end{array} \tag{14}$$

PROOF. Straightforward. □

LEMMA D.6. Given an algebra for  $S(T \oplus T')$  presented as in Lemma D.4 by compatible algebra structures

$$\mathbf{a}: SX \rightarrow X \qquad \mathbf{b}: TX \rightarrow X \qquad \mathbf{c}: \Gamma_{S X} X \rightarrow X,$$

the diagram (14) commutes.

PROOF. We show commutation of the diagram by precomposing both sides by the split epimorphism  $\Gamma_{S a} a$ , and observing that they both are equal to the left bottom composite in (d2'). For  $c$ , we readily get the top right part of (d2'), hence the result. For the top right part of (14), the result follows by chasing the diagram below.

$$\begin{array}{ccccccc}
 \Gamma_{SSX}SX & \xrightarrow{\Gamma_{Sa}SX} & \Gamma_{SX}SX & \xrightarrow{\Gamma_{Sx}a} & \Gamma_{SX}X \\
 \downarrow \Gamma_{\mu_X^S}SX & & \downarrow \Gamma_aSX & & \downarrow \Gamma_aX \\
 \Gamma_{SX}SX & \xrightarrow{\Gamma_aSX} & \Gamma_XSX & \xrightarrow{\Gamma_Xa} & \Gamma_XX \\
 \downarrow \Gamma_{\eta_X^S}SX & & \downarrow \Gamma_{\eta_X^S}SX & & \downarrow \Gamma_{\eta_X^S}X \\
 \Gamma_{SSX}SX & \xrightarrow{\Gamma_{Sa}SX} & \Gamma_{SX}SX & \xrightarrow{\Gamma_{Sx}a} & \Gamma_{SX}X \\
 \downarrow \Gamma_{\mu_X^S}SX & & & & \downarrow c \\
 \Gamma_{SX}SX & & & & X \\
 \downarrow d_{X,X}^{\bullet\omega} & & & & \\
 ST\Gamma_{ST|X}^{\bullet}X & \xrightarrow{ST(b \circ c)} & STX & \xrightarrow{Sb} & SX & \xrightarrow{a} & X
 \end{array}$$

□

LEMMA D.7. For any  $(\delta, d)$ -algebra  $X$  with structure induced by

$$\mathbf{a}: SX \rightarrow X$$

$$\mathbf{b}: TX \rightarrow X$$

$$\mathbf{c}: \Gamma_X X \rightarrow X,$$

the following diagram commutes

$$\begin{array}{ccc}
 \Gamma_X SX & \xrightarrow{\Gamma_{\eta_X^S}SX} & \Gamma_{SX}SX & \xrightarrow{d_{X,X}^{\bullet\omega}} & ST\Gamma_{ST|X}^{\bullet}X \\
 \downarrow \Gamma_X a & & & & \downarrow ST(\Gamma_{a \circ Sb}X + [X, X]) \\
 & & & & ST\Gamma_X^{\circ}X \\
 & & & & \downarrow ST[c, X] \\
 & & & & STX \\
 & \searrow c & & \swarrow a \circ Sb & \\
 & & X & & 
 \end{array}$$

PROOF. We start by reducing to Subdiagram (C) as below.

$$\begin{array}{ccccc}
 & & \Gamma_{SX}SX & & \\
 & \nearrow \Gamma_{\eta_X^S}SX & & \searrow d_{X,X}^{\bullet\omega} & \\
 \Gamma_X SX & \xrightarrow{\Gamma_{\eta_X^S}SX} & \Gamma_{STX}^{\circ}SX & \xrightarrow{\widetilde{d}_{X,X}^{\bullet\omega}} & ST\Gamma_{ST|X}^{\bullet}X \\
 \downarrow \Gamma_X a & \searrow in_1 \circ (\Gamma_{\eta_X^S}STX) & \downarrow \Gamma_{STX}^{\circ}a & & \downarrow ST(\Gamma_{a \circ Sb}X + [X, X]) \\
 \Gamma_X X & \xrightarrow{in_1 \circ (\Gamma_{\eta_X^S}STX)} & \Gamma_{STX}^{\circ}X & \text{(C)} & ST\Gamma_X^{\circ}X \\
 & \searrow in_1 & \downarrow \Gamma_{a \circ Sb}X + X & & \downarrow ST[c, X] \\
 & & \Gamma_X^{\circ}X & & STX \\
 & \searrow c & \downarrow [c, X] & & \swarrow a \circ Sb \\
 & & X & & 
 \end{array}$$

Commutation of (C) is then equivalent to commutation of the transposed diagram below.

$$\begin{array}{ccccc}
 SX & \xrightarrow{\widetilde{d}^\omega} & \mathbb{T}_{STX}^\times ST\Gamma_{ST|X}^\bullet X & \xrightarrow{\mathbb{T}_{STX}^\times ST(\Gamma_{\text{aoSb}} X + [X, X])} & \mathbb{T}_{STX}^\times ST\Gamma_X^\circ X \\
 \downarrow \text{a} & & & & \mathbb{T}_{STX}^\times ST[c, X] \downarrow \\
 X & & & & \mathbb{T}_{STX}^\times STX \\
 \downarrow \eta_{STX, X} & & & & \mathbb{T}_{STX}^\times (\text{aoSb}) \downarrow \\
 \mathbb{T}_{STX}^\times \Gamma_{STX}^\circ X & \xrightarrow{\mathbb{T}_{STX}^\times \Gamma_{\text{aoSb}}^\circ X} & \mathbb{T}_{STX}^\times \Gamma_X^\circ X & \xrightarrow{\mathbb{T}_{STX}^\times [c, X]} & \mathbb{T}_{STX}^\times X
 \end{array}$$

Because  $\widetilde{d}_{X, X}^\omega$  is a monad morphism by construction, one easily proves that both sides have the same restriction along  $\eta_X^S: X \rightarrow SX$ , so it suffices to prove that both sides are  $S$ -algebra morphisms. By construction,  $\text{a}$  and  $\widetilde{d}_{X, X}^\omega$  are, so we focus on other morphisms.

- We first recall Lemma C.26, which says that  $\mathbb{T}_{STX}^\times$  lifts to a functor  $STO_X\text{-Alg} \rightarrow \Sigma\text{-alg}$ , by considering, for any  $STO_X$ -algebra  $\mathbf{e}: STO_X A \rightarrow A$ , the following  $\Sigma$ -algebra structure:

$$\begin{array}{c}
 \Sigma \mathbb{T}_{STX}^\times A \\
 \downarrow \widetilde{d}_{X, X}^\omega \mathbb{T}_{STX}^\times A \\
 \mathbb{T}_{STX}^\times STO_X \Gamma_{STX}^\circ \mathbb{T}_{STX}^\times A \\
 \downarrow \mathbb{T}_{STX}^\times STO_X \varepsilon_{STX} A \\
 \mathbb{T}_{STX}^\times STO_X A \\
 \downarrow \mathbb{T}_{STX}^\times \mathbf{e} \\
 \mathbb{T}_{STX}^\times A.
 \end{array}$$

- For proving that the top right composite is a morphism of  $\Sigma$ -algebras, it thus suffices to prove that the underlying composite  $ST\Gamma_{ST|X}^\bullet X \rightarrow X$  is a morphism of  $STO_X$ -algebras. But this is in turn equivalent to being both a morphism of  $ST$ - and  $O_X$ -algebras. This easy to see for  $O_X$ , and the given composite is a morphism of  $ST$ -algebras as a composite of two free  $ST$ -algebra morphisms, and the  $ST$ -algebra structure  $STX \rightarrow X$ .
- For the bottom left composite, we show that it is a  $\Sigma$ -algebra morphism by chasing the diagram in Figure 17, where the bottom right subdiagram commutes by Lemma D.8 below.

□

LEMMA D.8. For any  $(\delta, d)$ -algebra  $X$  with structure induced by

$$\text{a: } SX \rightarrow X \qquad \text{b: } TX \rightarrow X \qquad \text{c: } \Gamma_X X \rightarrow X,$$

the following diagram commutes.

$$\begin{array}{ccc}
 \Gamma_{STX}^\circ \Sigma X & \xrightarrow{d_{X, X}^\circ} & ST\Gamma_{ST|X}^\bullet X \\
 \Gamma_{STX}^\circ (\text{ao}\eta_{\Sigma, X}) \downarrow & & \downarrow ST(\Gamma_{\text{aoSb}} X + [X, X]) \\
 \Gamma_{STX}^\circ X & & ST\Gamma_X^\circ X \\
 \Gamma_{\text{aoSb}}^\circ X \downarrow & & \downarrow ST[c, X] \\
 \Gamma_X^\circ X & & STX \\
 [c, X] \swarrow & & \swarrow \text{aoSb} \\
 & X &
 \end{array}$$

PROOF. Since the domain is a coproduct, we proceed termwise. On  $\Sigma X$ , the result is straightforward. On  $\Gamma_{STX}^\circ \Sigma X$ , we observe that the diagram in Figure 18 commutes for all  $X, Y$ , so that the

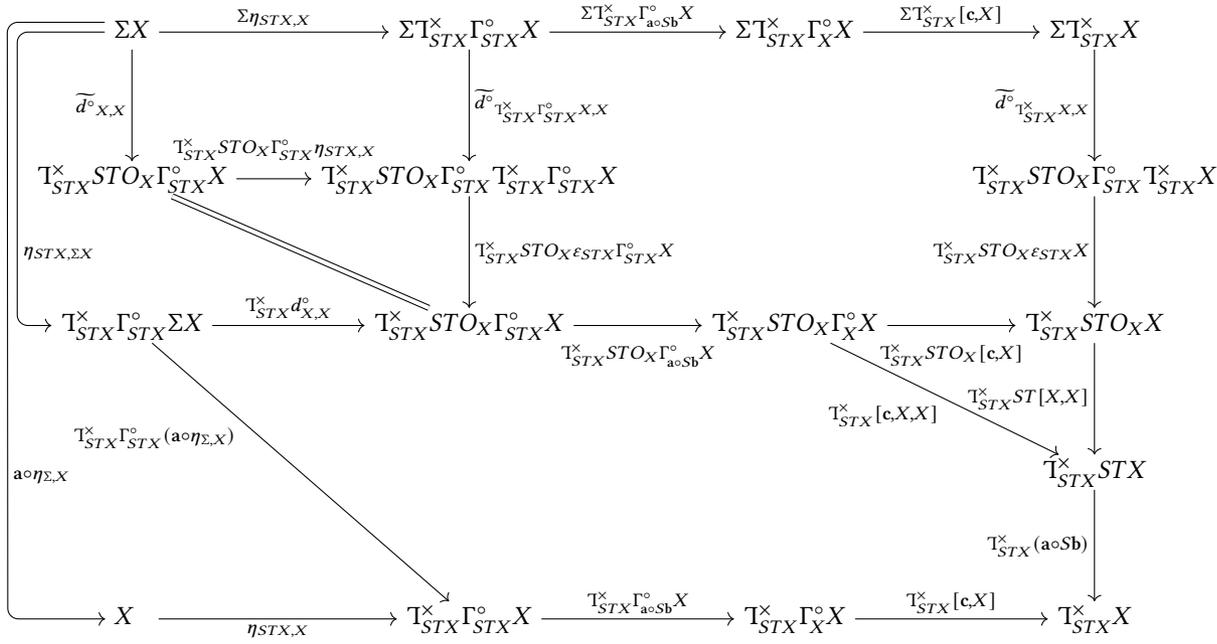


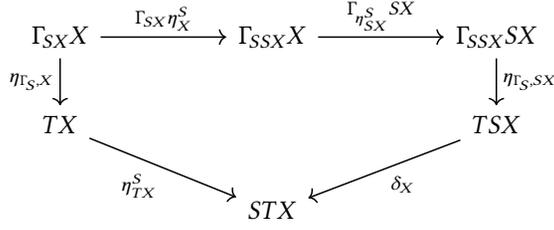
Fig. 17. Diagram for Lemma D.7

lemma reduces (using interchange) to commutation of the following diagram,

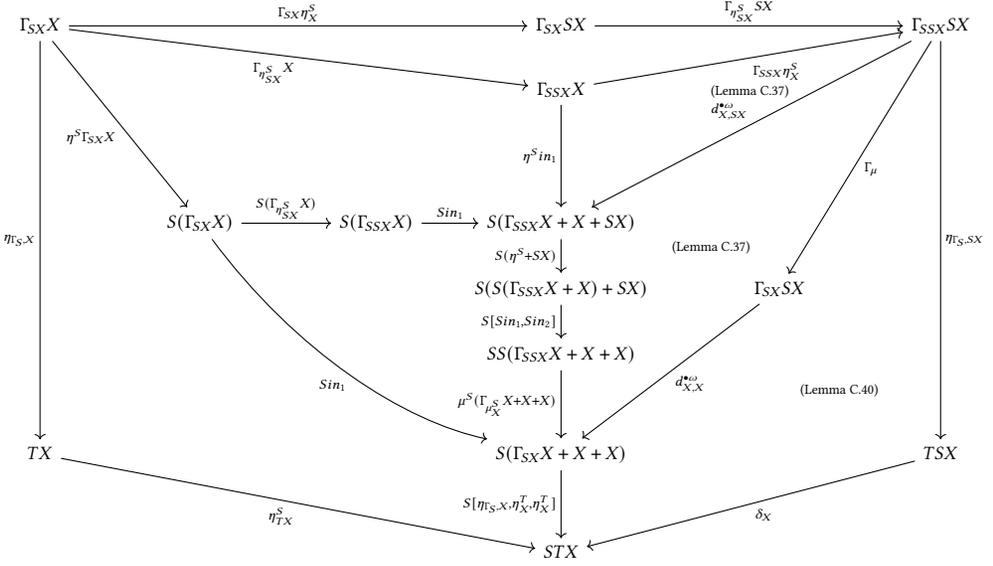




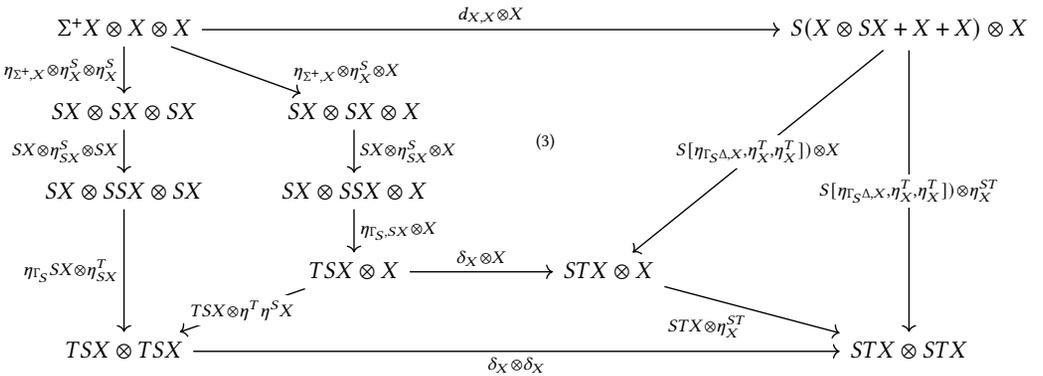




PROOF. By diagram chasing, as follows.



In Figure 20, the subdiagram marked  $(d/\delta \otimes \delta)$  commutes by chasing as follows.



The subdiagram marked  $(st/\delta)$  has a coproduct as domain, so we check its commutation termwise in Figure 21.

The subdiagram  $(\Gamma; \Gamma)$  commutes as in Figure 22.

## F.2 Coherence of $R_1$

We now check coherence of  $R_1$ , in Figure 23.

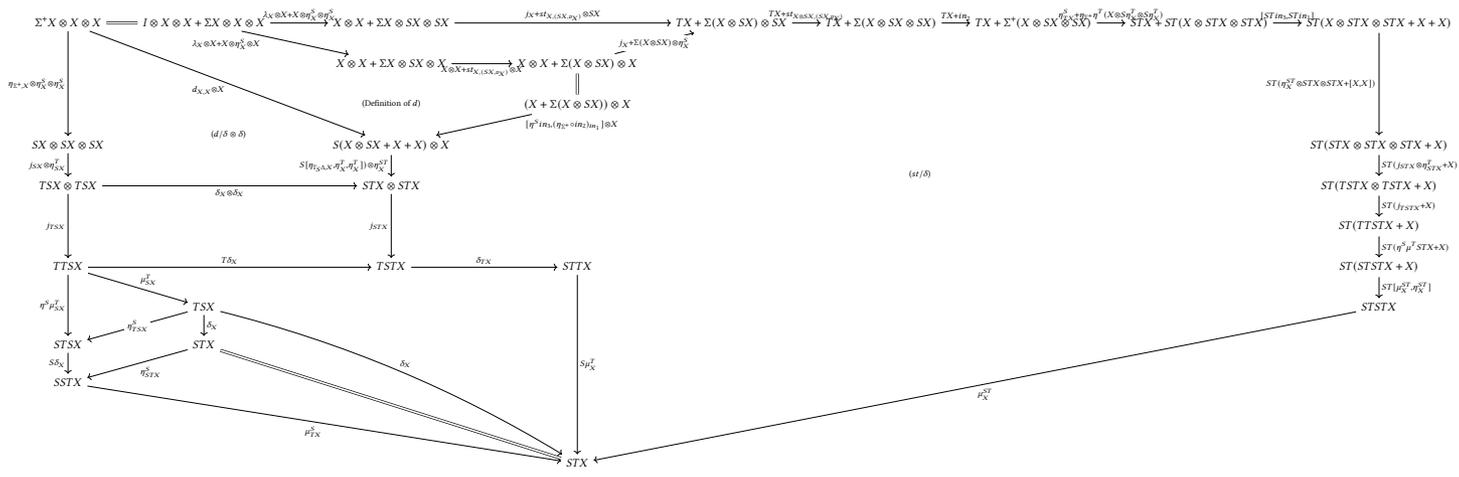


Fig. 20. Coherence of  $L_1$



A unified treatment of structural definitions on syntax

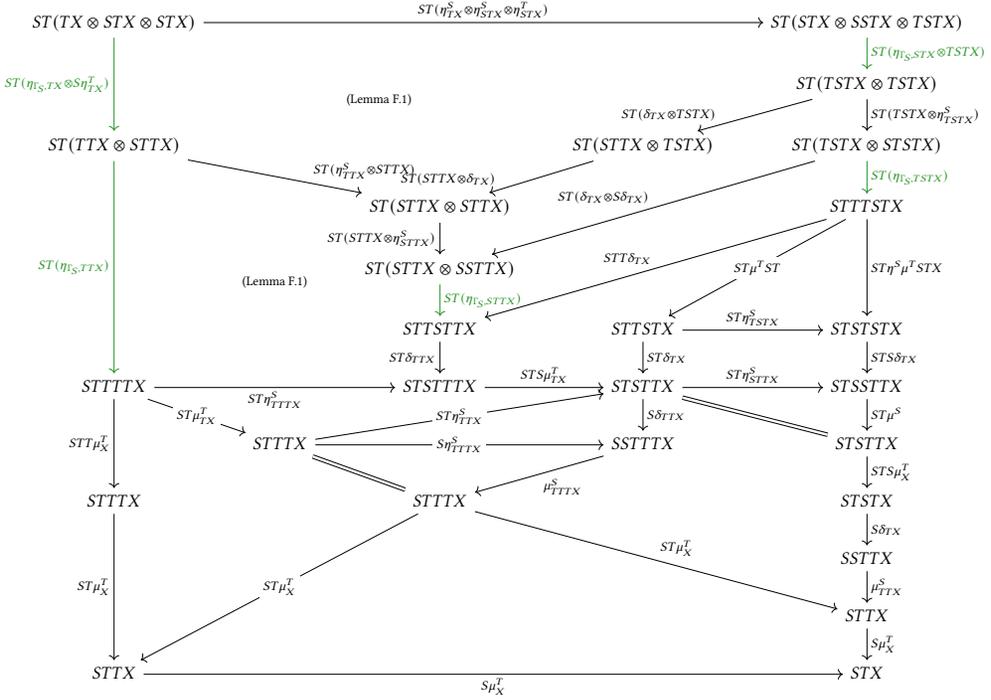
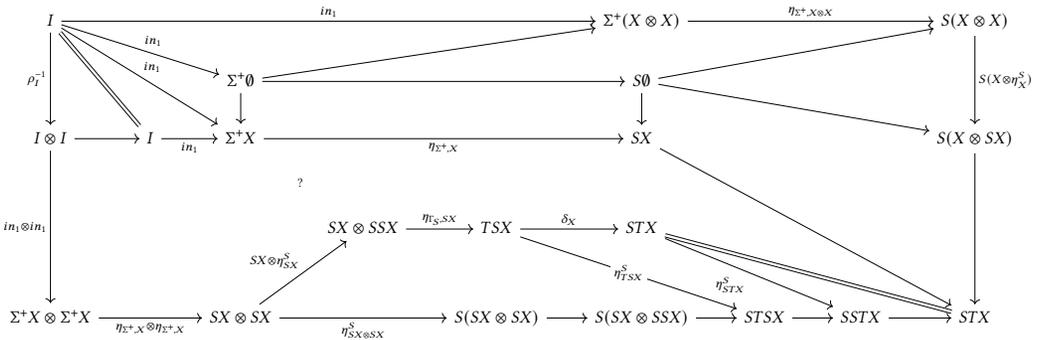


Fig. 22. Subdiagram ( $\Gamma$ ;  $\Gamma$ )

There, the top subdiagram with red arrows is the associativity axiom for pointed strengths. We then use naturality of  $st$  three times easily, and a fourth time much less easily. The point is that we use naturality at the pair of morphisms

$$X \xrightarrow{id_X} X \quad S(X \otimes X) \xrightarrow{S(X \otimes \eta_X^S)} S(X \otimes SX) \xrightarrow{S\eta_{\Gamma_S, X}} STX,$$

which requires us to prove that the second morphism is a morphism of pointed objects. We do this by chasing the following diagram,



whose question marked subdiagram commutes as follows.

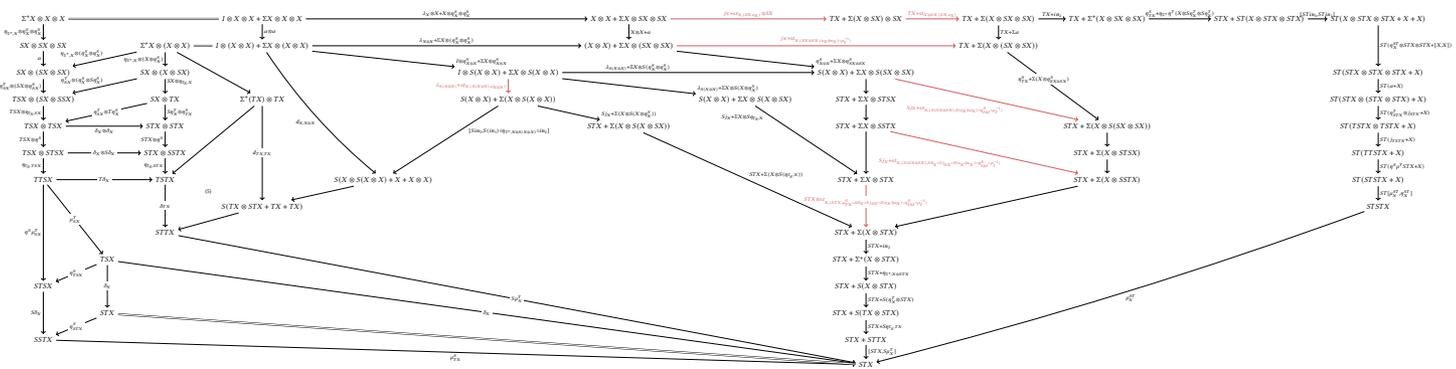
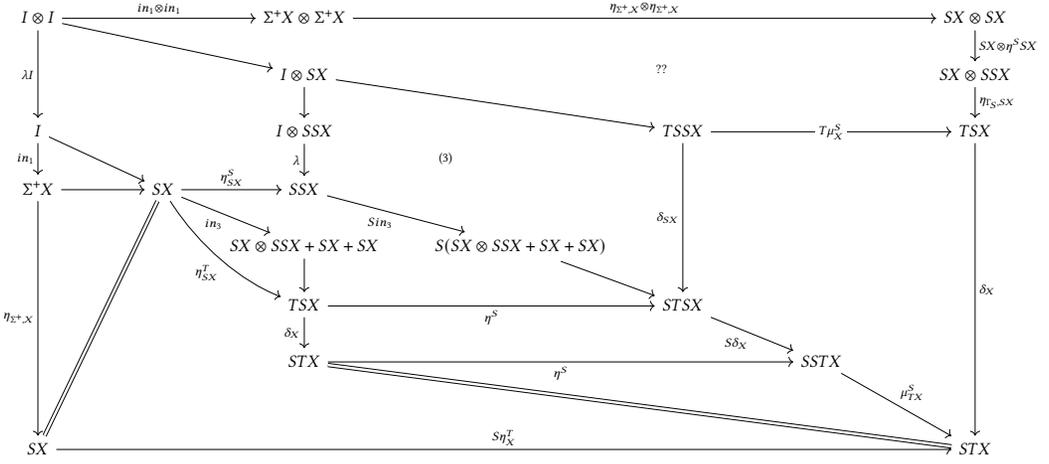
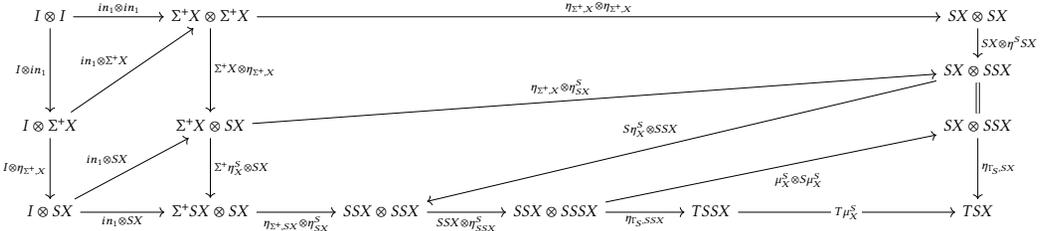


Fig. 23. Coherence of  $R_1$

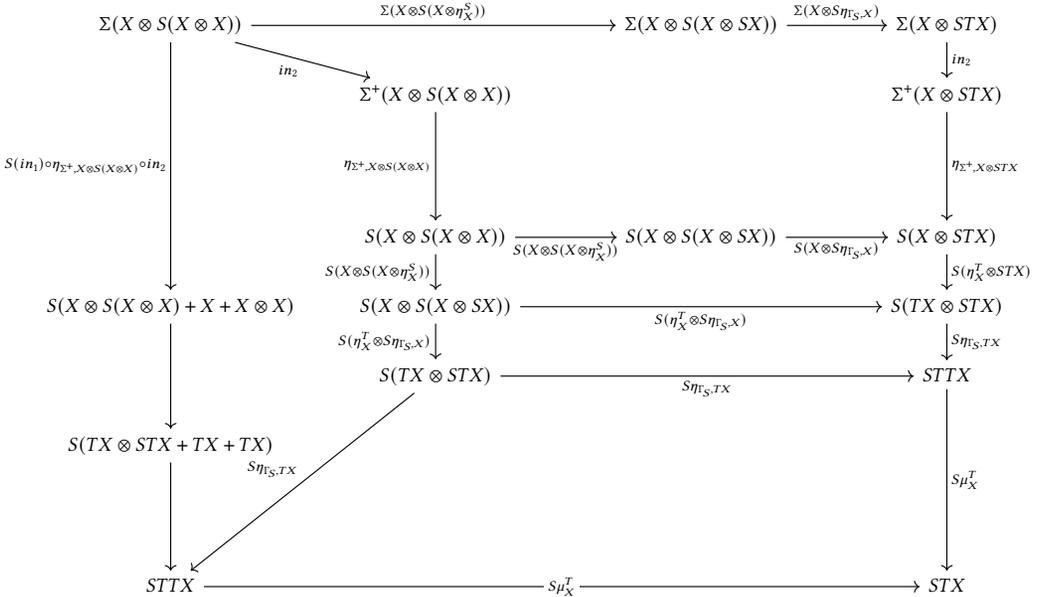
A unified treatment of structural definitions on syntax



The double question marked subdiagram of the latter diagram in turn commutes as follows.



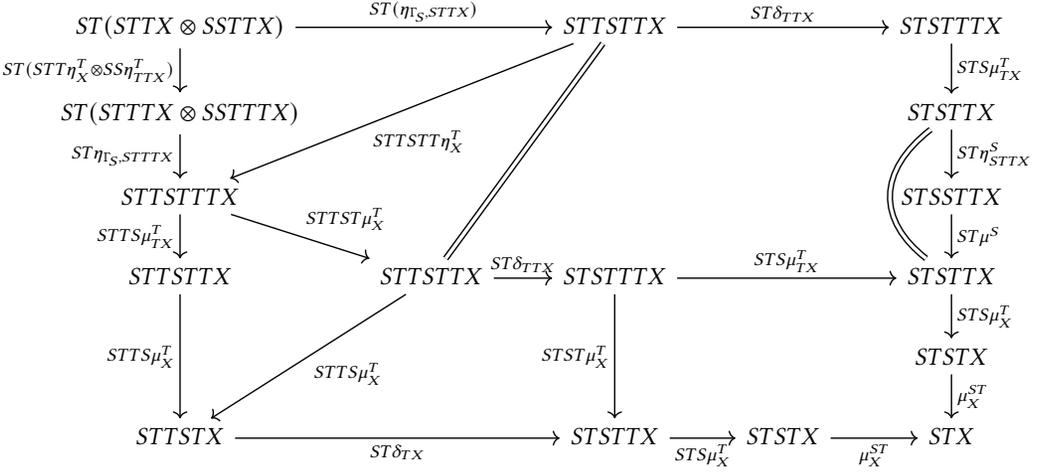
It remains to prove that both large subdiagrams at the bottom of Figure 23 commute. For the left-hand one, the first term is almost trivial, and the second term is easy:



In order to prove the right-hand one, we need the following lemma.

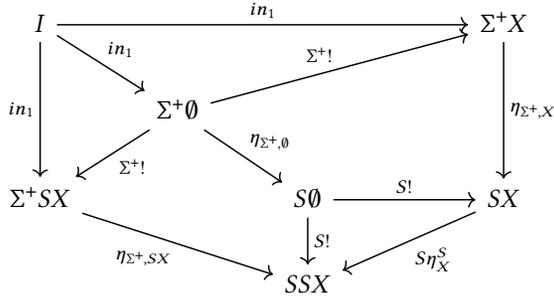


There the bottom right subdiagram commutes as follows.



### F.3 Coherence of $L_2$

We then check coherence of  $L_2$  in Figure 24, where, in the middle triangle which unfolds  $d_{SX,SX}$ , we use  $\eta_{SX}^S \circ \eta_{\Sigma^+,X} \circ in_1$  instead of  $v_{SX}$  as a point. This is justified because these morphisms are in fact equal, as the diagram below shows.



The left-hand, question-marked subdiagram of Figure 24 only commutes when postcomposed with  $\delta_X \circ T\mu_X^S = \mu_{TX}^S \circ S\delta_X \circ \delta_{SX}$ , as follows.







