



HAL
open science

A categorical framework for congruence of applicative bisimilarity in higher-order languages

Tom Hirschowitz, Ambroise Lafont

► **To cite this version:**

Tom Hirschowitz, Ambroise Lafont. A categorical framework for congruence of applicative bisimilarity in higher-order languages. 2021. hal-02966439v3

HAL Id: hal-02966439

<https://hal.science/hal-02966439v3>

Preprint submitted on 30 Mar 2021 (v3), last revised 14 Jun 2022 (v6)

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 CATEGORICAL FRAMEWORK FOR CONGRUENCE OF APPLICATIVE BISIMILARITY IN HIGHER-ORDER LANGUAGES

TOM HIRSCHOWITZ AND AMBROISE LAFONT

Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA, 73000, Chambéry, France

UNSW, Sydney, Australia

ABSTRACT. Applicative bisimilarity is a coinductive characterisation of observational equivalence in call-by-name lambda-calculus, introduced by Abramsky in 1990. Howe (1989) gave a direct proof that it is a congruence. We propose a categorical framework for specifying operational semantics, in which we prove that (an abstract analogue of) applicative bisimilarity is automatically a congruence. Example instances include standard applicative bisimilarity in call-by-name and call-by-value λ -calculus, as well as in a simple non-deterministic variant.

1. INTRODUCTION

This paper is a contribution to the search for efficient and high-level mathematical tools to specify and reason about programming languages. This search arguably goes back at least to Turi and Plotkin [44], who coined the name “Mathematical Operational Semantics”, and proved a general congruence theorem for bisimilarity. This approach has been deeply investigated, notably for quantitative languages [8]. However, as of today, attempts to apply it to higher-order (e.g., functional) languages have failed.

In previous work [26, 25], the first author has proposed an alternative approach to the problem, dropping the coalgebraic notion of bisimulation used by Turi and Plotkin in favour of a notion based on factorisation systems, similar to Joyal et al.’s [28]. Furthermore, congruence of bisimilarity is notably obtained by assuming that syntax induces a *familial* monad, in the sense of Diers [15, 13, 46].

However, the new approach has only been applied to simple, first-order languages like the π -calculus [34, 41], and Positive GSOS specifications [10]. In this paper, we extend it to functional languages, notably covering the paradigmatic case of *applicative* bisimilarity [1] in call-by-name and call-by-value λ -calculus, as well as in a simple, non-deterministic λ -calculus [39, §7]. We thus, obtain for the first time a generic, categorical congruence result for applicative bisimilarity in functional languages.

A bit more precisely:

- We propose a simple notion of signature for programming languages.

Key words and phrases: Programming languages; categorical semantics; operational semantics; Howe’s method.

- Each signature has a category of models, including an initial one, intuitively its operational semantics.
- An abstract analogue of applicative bisimilarity, called *substitution-closed* bisimilarity, may be defined in any model, and in particular in the initial one.
- Under suitable hypotheses, we show that substitution-closed bisimilarity is a congruence.

Categorically, this unfolds as follows.

(i) We define an abstract notion of (labelled) transition systems, as objects of a category \mathbf{C} , in such a way that

- there is a forgetful functor $\mathbf{C} \rightarrow \mathbf{C}_0$, intuitively returning the (potentially structured) set of states of a transition system;
- bisimulation and bisimilarity may be defined for any transition system.

(ii) Following Fiore, Plotkin, and Turi’s seminal framework [20], we then assume that \mathbf{C}_0 is monoidal, and define models of the syntax to be *monoid algebras* for a given *pointed strong* endofunctor Σ_0 on \mathbf{C}_0 . Monoid algebras, a.k.a. Σ_0 -monoids, are Σ_0 -algebras equipped with compatible monoid structure – this models capture-avoiding substitution. The category $\Sigma_0\text{-mon}$ of models has an initial object \mathbf{Z}_0 , as we prove in Coq [29]. In the case of call-by-name λ -calculus, \mathbf{Z}_0 is precisely the syntax.

(iii) This category $\Sigma_0\text{-mon}$ induces by pullback a category $\Sigma_0\text{-Mon}$ (capital ‘M’!) of transition systems whose states are equipped with Σ_0 -monoid structure. We call these *transition monoid algebras*, or *transition Σ_0 -monoids*. The relevant notions of bisimulation and bisimilarity for such objects are defined as in (i), but for *substitution-closed* relations.

(iv) We then define models of the dynamics to be certain algebras, called *vertical*, for an endofunctor on $\Sigma_0\text{-Mon}$. There is an initial vertical algebra \mathbf{Z} , which in examples is the syntactic transition system (and standard applicative bisimilarity coincides with substitution-closed bisimilarity).

(v) Finally, following an abstract analogue of Howe’s method, we show that, under suitable hypotheses, substitution-closed bisimilarity on \mathbf{Z} is a congruence. One crucial hypothesis is *cellularity*, in a sense closely related to [21].

Related work. Plotkin and Turi’s *bialgebraic semantics* [44] and its few variants [14, 42] prove abstract congruence theorems for bisimilarity. However, they do not cover higher-order languages like the λ -calculus, let alone applicative bisimilarity. This was one of the main motivations for our work. Among more recent work, quite some inspiration was drawn from Ahrens et al. [5, 24], notably in the use of vertical algebras. However, a difference is that we do not insist that transitions be stable under substitution. Links with other relevant work by, e.g., Bodin et al. [11] or Dal Lago et al. [30], though desirable, remain unclear, perhaps because of the very different methods used. Furthermore, the cellularity used here is close to but different from the \mathbf{T}_γ^\vee -familiality of [26]. It would be instructive to better understand potential links between the two. Finally, let us mention recent work which, just like ours, establishes abstract versions of standard constructions and theorems in programming language theory like type soundness [6] or gluing [18, 19].

Relation to conference version. This paper is a bit more than a journal version of our previous work [12]. Here is a brief summary of changes.

(a) In [12], we work with a significant generalisation of monoid algebras to *skew monoidal* categories [43] and *structurally strong* functors. Here, by giving a better type to Σ_1 , the endofunctor for specifying the dynamics, we manage to work with standard monoid algebras.

This has the additional advantages of

- avoiding a slightly *ad hoc* compositionality assumption of [12],
- and relaxing the hypothesis that the tensor product be familial.

(b) In [12], because Howe’s closure operates only at the level of states, we work mostly with *prebisimulations*, in the sense of [26, §5.1]. This notion is designed to detect when the state part of a relation underlies a bisimulation, regardless of what it does on transitions. However, it feels more *ad hoc* than the standard definition of bisimulation by lifting [28]. In this paper, we extend Howe’s closure to transitions, thus avoiding prebisimulations entirely.

(c) In [12], we rely on directed unions of relations, which leads to quite a few, rather painful proofs by induction. Here, we use higher-level methods to construct Howe’s closure, essentially through categorification and algebraicisation. Namely:

- (1) We define bisimilarity as the final object not in some partially-ordered set of relations as usual, but in some category of spans (see also [9]).
- (2) Furthermore, we define Howe’s closure directly as a free monoid algebra for a suitable pointed strong endofunctor on spans.
- (3) More generally, we systematically rely on universal properties, which simplifies a significant number of proofs.

(d) We put less emphasis on cellularity, viewing it only as a sufficient condition for a perhaps more natural hypothesis which already appeared in a slightly different form in [42], namely the fact that Σ_1 preserves functional bisimulations.

(e) We obtain a congruence theorem of similar scope (Theorem 6.15), and cover two new, detailed examples (§8): call-by-value, big-step λ -calculus (which was covered but too naively in [12], as we explain) and a call-by-name λ -calculus with unary, erratic choice from [39, §7].

Plan. In §2, we start by briefly recalling call-by-name λ -calculus and applicative bisimilarity. We then explain how to view the latter as substitution-closed bisimilarity, and sketch Howe’s method. In §3, we then give a brief overview of the new framework by example, including a recap on monoid algebras and a statement of the main theorem. We then dive into the technical core of the paper by presenting our framework for transition systems and bisimilarity (§4), operational semantics (§5), and then substitution-closed bisimilarity and the main result (Theorem 6.15), together with a high-level proof sketch (§6). In §7, we reformulate the main hypothesis of Theorem 6.15 using cellularity, which allows us to use well-known results from weak factorisation systems as sufficient conditions. We then apply our results to examples in §8. The full proof of Theorem 6.15 is given in §9. Finally, we conclude and give some perspectives on future work in §10.

Notation. We often conflate natural numbers $n \in \mathbb{N}$ with corresponding sets $\{1, \dots, n\}$. For all sets X and objects C of a given category, we denote by $X \cdot C$ the X -fold coproduct of C with itself, i.e., $\sum_{x \in X} C$. Let **Gph** denote the category of (directed, multi) graphs, **Cat** the category of small categories, and **CAT** the category of locally small categories. Let $\widehat{\mathbb{C}}$ denote the category of (contravariant) presheaves on \mathbb{C} , and $\mathbf{y}: \mathbb{C} \rightarrow \widehat{\mathbb{C}}$ the Yoneda embedding, mapping c to $\mathbb{C}(-, c)$. Given a presheaf $F \in \widehat{\mathbb{C}}$, an element $x \in F(c)$, and a

morphism $c \xrightarrow{f} c'$, we sometimes denote $F(f)(x)$ by $x \cdot f$. Given two categories C_1 and C_2 , we denote by $[C_1, C_2]$ the functor category between them. In a category with products, we interchangeably use spans $X \leftarrow R \rightarrow Y$ and their pairings $R \rightarrow X \times Y$, sometimes also calling the latter spans.

Any finitary endofunctor F on any cocomplete category admits by [38, Theorem 2.1] an initial algebra, which we denote by \mathbf{Z}_F . Although this is detailed below, we prefer to avoid confusion and warn the reader that we also use \mathbf{Z}_F for the initial F -monoid, for any *pointed strong* endofunctor F on any nice monoidal category. Throughout the paper, when not explicitly attached to some F , \mathbf{Z} is shorthand for \mathbf{Z}_{Σ_1} (see, e.g., Proposition 3.21 or Theorem 5.18).

Finally, let us fix some notation about weak factorisation systems. In any category \mathcal{C} , we say that a morphism $f: A \rightarrow B$ has the (weak) left lifting property w.r.t. $g: C \rightarrow D$ when for all commuting squares as on the right there is a lifting k as shown that makes both triangles commute. Equivalently, we say that g has the right lifting property w.r.t. f , and write $f \pitchfork g$. Given a fixed set \mathbb{J} of morphisms, the set of morphisms g such that $j \pitchfork g$ for all $j \in \mathbb{J}$ is denoted by \mathbb{J}^\pitchfork . Similarly, the set of morphisms f such that $f \pitchfork j$ for all $j \in \mathbb{J}$ is denoted by ${}^\pitchfork\mathbb{J}$. In particular, if $f \in {}^\pitchfork(\mathbb{J}^\pitchfork)$ and $g \in \mathbb{J}^\pitchfork$, then $f \pitchfork g$. If \mathcal{C} is locally presentable [2], then $({}^\pitchfork(\mathbb{J}^\pitchfork), \mathbb{J}^\pitchfork)$ forms a *weak factorisation system*, in the sense that additionally any morphism $f: X \rightarrow Y$ factors as $X \xrightarrow{l} Z \xrightarrow{r} Y$ with $l \in {}^\pitchfork(\mathbb{J}^\pitchfork)$ and $r \in \mathbb{J}^\pitchfork$ (see [27, Theorem 2.1.14]). Morphisms in \mathbb{J}^\pitchfork are generically called *fibrations*, while morphisms in ${}^\pitchfork(\mathbb{J}^\pitchfork)$ are called *cofibrations*.

Let us conclude with the following easy, yet helpful result.

Lemma 1.1. *For any locally finitely presentable category and set \mathbb{J} of maps therein, if the domains and codomains of maps in \mathbb{J} are finitely presentable, then fibrations are closed under filtered colimits in the arrow category.*

Proof. Let us consider any given filtered diagram $(f_i: A_i \rightarrow B_i)_{i \in \mathbb{D}}$ of fibrations, and a colimit $f: A \rightarrow B$ in the arrow category, say \mathbf{C}^\rightarrow . We must show that $j \pitchfork f$ for all $j \in \mathbb{J}$. Let us thus further consider any given commuting square

$$\begin{array}{ccc} X & \xrightarrow{u} & A \\ j \downarrow & & \downarrow f \\ Y & \xrightarrow{v} & B \end{array}$$

with $j \in \mathbb{J}$. Colimits in the arrow category are pointwise, so $A = \text{colim}_i A_i$ and $B = \text{colim}_i B_i$, so, by finite presentability of X and Y , and by filteredness of the diagram $\mathbb{D} \rightarrow \mathbf{C}$, u and v factor through some A_{i_0} and B_{i_1} , respectively. By filteredness of the diagram again, w.l.o.g., we may take $i_0 = i_1$, so that $(u, v): j \rightarrow f$ factors through f_{i_0} . But because f_{i_0} is a fibration, we find a lifting as in

$$\begin{array}{ccccc} & & u & & \\ & \curvearrowright & & \curvearrowleft & \\ X & \xrightarrow{u'} & A_{i_0} & \longrightarrow & A \\ j \downarrow & \dashrightarrow & \downarrow f_{i_0} & & \downarrow f \\ Y & \xrightarrow{v'} & B_{i_0} & \longrightarrow & B \\ & \curvearrowleft & & \curvearrowright & \\ & & v & & \end{array}$$

which provides the desired lifting for the original square. □

2. A BRIEF REVIEW OF HOWE'S METHOD

2.1. Applicative bisimilarity. Let us consider the standard, big-step presentation of call-by-name λ -calculus:

$$\frac{}{\lambda x.e \Downarrow \lambda x.e} \qquad \frac{e_1 \Downarrow \lambda x.e'_1 \quad e'_1[x \mapsto e_2] \Downarrow e_3}{e_1 e_2 \Downarrow e_3}$$

Standardly, the evaluation relation \Downarrow is considered between closed terms only.

Applicative bisimilarity is standardly introduced in two stages, which we now recall. In a first stage, one defines applicative bisimulation on closed terms.

Definition 2.1. A relation R over closed λ -terms is an *applicative bisimulation* iff $e_1 R e_2$ and $e_1 \Downarrow \lambda x.e'_1$ entail the existence of e'_2 such that $e_2 \Downarrow \lambda x.e'_2$ and, for all terms e , $e'_1[x \mapsto e] R e'_2[x \mapsto e]$, and symmetrically.

Applicative bisimulations are closed under unions, and so there is a largest applicative bisimulation, called *applicative bisimilarity* and denoted by \sim .

Then comes the second stage:

Definition 2.2. The *open extension* of a relation R on closed terms is the relation R^\otimes on potentially open terms such that $e R^\otimes e'$ iff for all closed substitutions σ covering all involved free variables we have $e[\sigma] R e'[\sigma]$.

Let us readily notice the following alternative characterisation of open extension.

Definition 2.3. A relation S on open terms is *substitution-closed* iff for all $e S e'$ and (potentially open) substitutions σ , we have $e[\sigma] S e'[\sigma]$.

Lemma 2.4. *The open extension of any relation R is the greatest substitution-closed relation contained in R on closed terms.*

Proof. Let us first show that R^\otimes is substitution-closed. For any $e_1 R^\otimes e_2$ and σ , we want to show $e_1[\sigma] R^\otimes e_2[\sigma]$. For this, we in turn need to show that for all closing substitutions γ , we have $e_1[\sigma][\gamma] R e_2[\sigma][\gamma]$. But $e_i[\sigma][\gamma] = e_i[\gamma \circ \sigma]$ and $\gamma \circ \sigma$ is closing, so because $e_1 R^\otimes e_2$, by definition of open extension, we get $e_1[\sigma][\gamma] R e_2[\sigma][\gamma]$ as desired.

Let us now show that R^\otimes is the greatest substitution-closed relation containing R on closed terms. For this, consider any substitution-closed R' contained in R on closed terms: for all $e R' e'$, by substitution-closedness, we have $e[\sigma] R' e'[\sigma]$ for all closing σ . So because R' is contained in R on closed terms, we further have $e[\sigma] R e'[\sigma]$. This proves $e R^\otimes e'$, and so thus $R' \subseteq R^\otimes$ as desired. \square

The result we wish to prove in the abstract setting is the following (see [37] for a historical account).

Theorem 2.5. *The open extension \sim^\otimes of applicative bisimilarity is a congruence: it is an equivalence relation, and furthermore it is context-closed, i.e.,*

- $e_1 \sim^\otimes e_2$ entails $\lambda x.e_1 \sim^\otimes \lambda x.e_2$ for all x ;
- $e_1 \sim^\otimes e_2$ and $e'_1 \sim^\otimes e'_2$ entail $e_1 e'_1 \sim^\otimes e_2 e'_2$.

Proving that \sim^\otimes is an equivalence relation is in fact straightforward. In the following, we focus on the context-closedness property.

2.2. Howe's method. Howe's method for proving Theorem 2.5 consists in considering a suitable relation \sim^\bullet , closed under substitution and context, and containing \sim^\otimes by construction. He then (roughly) shows that this relation \sim^\bullet is a bisimulation. By maximality of \sim^\otimes , we then also have $\sim^\bullet \subseteq \sim^\otimes$ hence both relations coincide and \sim^\otimes is context-closed as desired. However, as explained in [12, §5.1], the presence of a substitution in the premises of a transition rule seems to require \sim^\bullet to be closed under *heterogeneous* substitution, in the sense that, e.g., if $e_1 \sim^\bullet e'_1$ and $e_2 \sim^\bullet e'_2$ (for open terms), then $e_1[x \mapsto e_2] \sim^\bullet e'_1[x \mapsto e'_2]$. The problem is that building this into the definition of \sim^\bullet leads to difficulties in the proof that it is a bisimulation. Howe's workaround consists in requiring \sim^\bullet to be closed under sequential composition with \sim^\otimes from the outset. Coupling this right action with context closedness, he thus defines \sim^\bullet as the smallest context-closed relation satisfying the rules

$$\frac{}{x \sim^\bullet x} \qquad \frac{e \sim^\bullet e' \quad e' \sim^\otimes e''}{e \sim^\bullet e''} .$$

By construction, \sim^\bullet is reflexive and context-closed. By a simple induction, it also substitution-closed. Furthermore, by reflexivity and the second rule, it also contains \sim^\otimes , and finally the second rule clearly entails $\sim^\bullet; \sim^\otimes \subseteq \sim^\bullet$. It takes an induction to prove stability under heterogeneous substitution, but to give a feel for it, in the base case where $e_1 \sim^\otimes e'_1$, we have

$$e_1[x \mapsto e_2] \sim^\bullet e_1[x \mapsto e'_2] \sim^\otimes e'_1[x \mapsto e'_2]$$

by context closedness of \sim^\bullet and substitution closedness of \sim^\otimes , so we conclude by $\sim^\bullet; \sim^\otimes \subseteq \sim^\bullet$.

The initial plan was to show that \sim^\bullet is a bisimulation and deduce that it coincides with \sim^\otimes . It can in fact be slightly optimised by first showing that \sim^\bullet is a simulation, and then that its transitive closure $(\sim^\bullet)^+$ is symmetric. The relation $(\sim^\bullet)^+$ is also a substitution-closed simulation, hence by symmetry a substitution-closed bisimulation. This entails the last inclusion in the chain $\sim^\otimes \subseteq \sim^\bullet \subseteq (\sim^\bullet)^+ \subseteq \sim^\otimes$, showing that all relations coincide. Finally, because \sim^\bullet is context-closed, so is \sim^\otimes , as desired.

2.3. Non-standard presentation. For technical convenience, we adopt a slightly different presentation of the transition rules, where the evaluation relation relates closed terms to terms with just one potential free variable. The new transition rules are as follows.

$$\frac{}{\lambda x. e \Downarrow e} \qquad \frac{e_1 \Downarrow e'_1 \quad e'_1[e_2] \Downarrow e_3}{e_1 e_2 \Downarrow e_3}$$

Here $e'_1[e_2]$ denotes substitution of the unique potential free variable in e'_1 by e_2 . We will see below that, with this transition system, the essentially standard notion of bisimulation coupled with the substitution-closedness requirement yields applicative bisimilarity.

3. OVERVIEW BY EXAMPLE

In this section, we describe one particular instance of our framework, which models call-by-name λ -calculus.

3.1. Syntax. Let us first define the syntax of λ -calculus, following [20], as an initial¹ object in a suitable category of models. Very roughly, a model of λ -calculus syntax should be something equipped with operations modelling abstraction and application, but also with substitution. Furthermore, certain natural compatibility axioms should be satisfied, e.g.,

$$(e_1 e_2)[\sigma] = e_1[\sigma] e_2[\sigma]. \quad (3.1)$$

A natural setting for specifying such operations is the functor category $\mathbf{C}_0 := [\mathbb{F}, \mathbf{Set}]$, where $\mathbb{F} \hookrightarrow \mathbf{Set}$ denotes the full subcategory spanning all sets of the form n (i.e., $\{1, \dots, n\}$, recalling notation from §1). For any $X \in [\mathbb{F}, \mathbf{Set}]$ and $n \in \mathbb{F}$, we think of $X(n)$ as a set of ‘terms’ with potential free variables in $\{1, \dots, n\}$. The action of X on morphisms $n \rightarrow n'$ is thought of as variable renaming. Returning to operations, being equipped with abstraction is the same as being a Σ_λ -algebra, where $\Sigma_\lambda: \mathbf{C}_0 \rightarrow \mathbf{C}_0$ is defined by $\Sigma_\lambda(X)(n) = X(n+1)$. An algebra structure on any X thus consists of a family of maps $X(n+1) \rightarrow X(n)$, natural in n . Similarly, for specifying both application and abstraction, we consider

$$\Sigma_0(X)(n) = X(n+1) + X(n)^2. \quad (3.2)$$

Let us now consider substitution. The idea here is to equip \mathbf{C}_0 with monoidal structure (\otimes, I) , such that

- elements of $(X \otimes Y)(n)$ are like *explicit substitutions* $x[\sigma]$, where $x \in X(p)$ and $\sigma: p \rightarrow Y(n)$ for some p , considered equivalent up to some standard equations²;
- elements of $I(n) := \{1, \dots, n\}$ are merely variables.

Being equipped with substitution (and variables) is thus the same as being a *monoid* for this tensor product:

- the multiplication $m_X: X \otimes X \rightarrow X$ maps any formal, explicit substitution $x[\sigma]$ to an actual substitution $x[\sigma]$, and
- the unit $e_X: I \rightarrow X$ injects variables into terms.

Finally, how do we enforce equations such as (3.1)? This goes in two stages:

- we first collect the way substitution is supposed to commute with each operation, by providing a *pointed strength*, i.e., a natural transformation with components $st_{X,Y}: \Sigma_0(X) \otimes Y \rightarrow \Sigma_0(X \otimes Y)$, where $X \in \mathbf{C}_0$ and $Y \in I/\mathbf{C}_0$,
- we then use the pointed strength to enforce all equations in one go, by requiring models to have compatible Σ_0 -algebra and substitution structure, in a suitable sense.

Let us first explain the notion of pointed strength.

Application: For modelling Equation (3.1) for application, we would in particular define $st_{X,Y}$ to map any $(in_2(x_1, x_2))[\sigma]$ to $in_2(x_1[\sigma], x_2[\sigma])$, for all $x_1, x_2 \in X(p)$ and $\sigma: p \rightarrow Y(n)$.

Abstraction: For abstraction, let us start by first stating the corresponding equation. We will then define the pointed strength accordingly. Supposing that Y is equipped with a point $e_Y: I \rightarrow Y$, we define $\sigma^\uparrow: p+1 \rightarrow Y(n+1)$ by copairing

$$p \xrightarrow{\sigma} Y(n) \xrightarrow{Y(in_1)} Y(n+1) \quad \text{and} \quad 1 = I(1) \xrightarrow{(e_Y)_1} Y(1) \xrightarrow{Y(in_2)} Y(n+1).$$

¹This pattern is advocated by the approach of Initial Semantics, where initiality provides a recursion principle.

²In [12], we instead considered a skew-monoidal variant where the tensor product does not enforce any standard equation.

The equation is then

$$\lambda(e)[\sigma] = \lambda(e[\sigma^\uparrow]). \quad (3.3)$$

Accordingly, we define the pointed strength to map any $in_1(x)([\sigma])$, where $x \in X(p+1)$ and $\sigma: p \rightarrow Y(n)$, to $in_1(x([\sigma^\uparrow]))$.

Let us now go through the second stage of how we impose the desired equations: a model of syntax will be a monoid X equipped with Σ_0 -algebra structure $\nu_X: \Sigma_0(X) \rightarrow X$, such that the following diagram commutes.

$$\begin{array}{ccc} \Sigma_0(X) \otimes X & \xrightarrow{st_{X,X}} \Sigma_0(X \otimes X) & \xrightarrow{\Sigma_0(m_X)} \Sigma_0(X) \\ \nu_X \otimes X \downarrow & & \downarrow \nu_X \\ X \otimes X & \xrightarrow{m_X} & X \end{array} \quad (3.4)$$

Indeed, suppose given, e.g., $in_1(e)([\sigma]) \in \Sigma_0(X) \otimes X$, by applying the left then bottom morphisms we obtain $\lambda(e)[\sigma]$, while applying the top then right morphisms we obtain $\lambda(e[\sigma^\uparrow])$, as desired.

All in all, we have:

Definition 3.1. For any finitary, pointed strong endofunctor Σ_0 , a *monoid algebra* for Σ_0 , or a Σ_0 -*monoid*, is a Σ_0 -algebra $(X, \nu_X: \Sigma_0(X) \rightarrow X)$, equipped with monoid structure $(m_X: X \otimes X \rightarrow X, e_X: I \rightarrow X)$, such that (3.4) commutes. A Σ_0 -monoid *morphism* is a morphism in \mathbf{C}_0 which is both a monoid morphism and a Σ_0 -algebra morphism.

Let Σ_0 -**mon** denote the category of Σ_0 -monoids and morphisms between them.

Let us conclude by (slightly informally) stating the result exhibiting standard syntax as the initial model [20, 17, 12]. See Proposition 5.1 below for a general and rigorous statement.

Proposition 3.2. *For any finitary, pointed strong endofunctor Σ_0 , under mild hypotheses, the forgetful functor Σ_0 -**mon** $\rightarrow \mathbf{C}_0$ is monadic, and the free Σ_0 -algebra over I (equivalently the initial $(I + \Sigma_0)$ -algebra) is an initial Σ_0 -monoid.*

Example 3.3. In the case of λ -calculus, the initial Σ_0 -monoid is thus the least fixed point $\mathbf{Z}_0 := \mu A.(I + \Sigma_0(A))$, which is isomorphic to the standard, low-level construction of syntax.

From this, one may deduce a characterisation of not only the initial Σ_0 -monoid, but all *free* Σ_0 -monoids, or in other words an explicit formula for the left adjoint to the forgetful functor. Namely:

Proposition 3.4. *The free Σ_0 -monoid, say $\mathcal{L}_0(K)$, over any $K \in \mathbf{C}_0$ is*

$$\mu A.(I + \Sigma_0(A) + K \otimes A).$$

Syntactically, letting $n \vdash_K e$ mean that $e \in \mathcal{L}_0(K)(n)$, $\mathcal{L}_0(K)$ is inductively generated by the following rules [22, §3.1],

$$\begin{array}{c} \frac{}{n \vdash_K x} \quad (x \in n) \qquad \frac{n \vdash_K e_1 \quad \dots \quad n \vdash_K e_p}{n \vdash_K k(e_1, \dots, e_p)} \quad (k \in K(p)) \\ \\ \frac{n \vdash_K e_1 \quad n \vdash_K e_2}{n \vdash_K e_1 e_2} \qquad \frac{n+1 \vdash_K e}{n \vdash_K \lambda(e)} \end{array}$$

modulo the equivalence

$$(f \cdot k)(e_1, \dots, e_q) \sim k(e_{\sigma(1)}, \dots, e_{\sigma(p)}),$$

for all $f: p \rightarrow q$, $k \in K(p)$, and $n \vdash_K e_1, \dots, e_q$, or perhaps more synthetically

$$(f \cdot k)([\sigma]) \sim k([\sigma \circ f]),$$

where $\sigma: q \rightarrow \mathcal{L}_0(K)(n)$ denotes the cotupling of e_1, \dots, e_q viewed as maps $1 \rightarrow \mathcal{L}_0(K)(n)$.

The first rule is the standard rule for variables, while the second one is for “constants”, i.e., elements of K . It corresponds to the term $K \otimes A$ in the above fixed point formula. When $p = 0$, we sometimes shorten the notation from $k(\Downarrow)$ to k . The last two rules are the standard rules for application and abstraction, and they correspond to the term $\Sigma_0(A)$ in the formula. The Σ_0 -monoid structure is syntactically straightforward; notably substitution satisfies $k[e_1, \dots, e_p][\sigma] = k[e_1[\sigma], \dots, e_p[\sigma]]$.

3.2. Transition systems and bisimilarity. The appropriate notion of transition system X for λ -calculus is as follows.

Definition 3.5. A *transition system* X consists of

- a *state object* $X_0 \in \mathbf{C}_0 = [\mathbb{F}, \mathbf{Set}]$,
- a set X_1 of *transitions*, and
- maps $X_0(0) \xleftarrow{s_X} X_1 \xrightarrow{t_X} X_0(1)$ giving the source and target of transitions (cf. §2.3).

Transition systems form a category \mathbf{C} , whose morphisms $X \rightarrow Y$ consist of compatible morphisms $f_0: X_0 \rightarrow Y_0$ and $f_1: X_1 \rightarrow Y_1$, in the sense that both of the following squares commute.

$$\begin{array}{ccc} X_1 & \xrightarrow{f_1} & Y_1 \\ s_X \downarrow & & \downarrow s_Y \\ X_0 & \xrightarrow{f_0} & Y_0 \end{array} \qquad \begin{array}{ccc} X_1 & \xrightarrow{f_1} & Y_1 \\ t_X \downarrow & & \downarrow t_Y \\ X_0 & \xrightarrow{f_0} & Y_0 \end{array}$$

Notation 3.6. We write $r: e \Downarrow f$ for $r \in X_1$ such that $s_X(r) = e$ and $t_X(r) = f$.

Example 3.7. Another example is the *syntactic* transition system has $\mathbf{Z}_0 \in \mathbf{C}_0$ from Example 3.3 as state object, and as transitions all derivations following the transition rules. We will come back to this case in Proposition 3.21.

Our next goal is to introduce bisimulation, for which it is convenient to characterise \mathbf{C} as a presheaf category. This characterisation may be established by abstract means, but let us describe it concretely first. It is clear from the definition that transitions systems are glorified graphs. And they form a presheaf category for essentially the same reason as graphs do. Here is the base category:

Definition 3.8. Let $\mathbb{F}[\Downarrow]$ denote the category obtained by augmenting \mathbb{F} with an object \Downarrow , together with morphisms $0 \xleftarrow{s_\Downarrow} \Downarrow \xrightarrow{t_\Downarrow} 1$, and their formal composites with non-identity morphisms from \mathbb{F} .

More concretely:

- There is exactly one morphism $0 \rightarrow n$ in \mathbb{F} for all n , which is an identity when $n = 0$, so for all $n \neq 0$ we have a morphism $s_{\Downarrow, n}: \Downarrow \rightarrow n$ making the following triangle commute.

$$\begin{array}{ccc} & \Downarrow & \\ s_{\Downarrow} \swarrow & & \searrow s_{\Downarrow, n} \\ 0 & \xrightarrow{\quad} & n \end{array}$$

- There are exactly n morphisms $1 \rightarrow n$ in \mathbb{F} for all $n \notin \{0, 1\}$ (and no morphisms $1 \rightarrow 0$), so for all such n and $i \in n$ we have a morphism $t_{\Downarrow, n, i} : \Downarrow \rightarrow n$ making the following triangle commute.

$$\begin{array}{ccc}
 & \Downarrow & \\
 t_{\Downarrow} \swarrow & & \searrow t_{\Downarrow, n, i} \\
 1 & \xrightarrow{i} & n
 \end{array}$$

Proposition 3.9. *Transition systems are isomorphic to covariant presheaves on $\mathbb{F}[\Downarrow]$.*

Notation 3.10. *We often implicitly convert from transition systems to covariant presheaves, and conversely.*

Proof sketch. This will be proved below by abstract means, but for intuition let us sketch the correspondence. Given a transition system $\langle s_X, t_X \rangle : X_1 \rightarrow X_0(0) \times X_0(1)$, we construct a presheaf X' by setting

- $X'(n) = X_0(n)$ for all $n \in \mathbb{F}$,
- $X'(\Downarrow) = X_1$,
- with the action of $s_{\Downarrow}, t_{\Downarrow} \in \mathbb{F}$ given by s_X and t_X ,
- inducing the action of all $s_{\Downarrow, n}$ and $t_{\Downarrow, n, i}$ by composition.

Conversely, for any presheaf Y , we construct a transition system Y' defined as follows:

- the state object $Y'_0 \in [\mathbb{F}, \mathbf{Set}]$ is given by restriction of Y ;
- the set Y'_1 of transitions is $Y(\Downarrow)$;
- and $s_{Y'}$ and $t_{Y'}$ are $Y(s_{\Downarrow})$ and $Y(t_{\Downarrow})$, respectively. \square

The correspondence yields basic, graph-like examples of transition systems.

Example 3.11.

(a) The representable presheaf \mathbf{y}_0 associated to $0 \in \mathbb{F}$ has a single closed state k_0 and its renamings (i.e., $(\mathbf{y}_0)_0(n) = 1$ for all n and for transitions $(\mathbf{y}_0)_1 = \emptyset$).

(b) The representable presheaf \mathbf{y}_{\Downarrow} consists of a closed state k_0 , a state k_1 with one free variable, their renamings, and a transition $e : k_0 \Downarrow k_1$.

(c) Let $\mathbf{y}_{s_{\Downarrow}} : \mathbf{y}_0 \rightarrow \mathbf{y}_{\Downarrow}$ denote the morphism mapping k_0 to k_0 .

Using these basic examples, we may define bisimulation and bisimilarity by lifting following [28]:

Definition 3.12. A morphism $X \rightarrow Y$ in \mathbf{C} is a *functional bisimulation* when it has the right lifting property w.r.t. the source map $\mathbf{y}_{s_{\Downarrow}} : \mathbf{y}_0 \rightarrow \mathbf{y}_1$. A span $X \leftarrow R \rightarrow Y$ is a *simulation* when its left leg $R \rightarrow X$ is a functional bisimulation, and a *bisimulation* when both legs are.

Remark 3.13. In this case, the Yoneda lemma says that $\mathbf{C}(\mathbf{y}_0, X) \cong X_0(0)$ and $\mathbf{C}(\mathbf{y}_{\Downarrow}, Y) \cong Y_1$. The right lifting property for a morphism $f : X \rightarrow Y$ thus says that given any $e \in Y_1$ whose source $e \cdot s_{\Downarrow}$ is $f(x)$ for some $x \in X_0(0)$, there exists $e' \in X_1$ such that $f(e') = e$ and $e' \cdot s_{\Downarrow} = x$, which matches the usual definition of functional bisimulation.

Definition 3.14. Let $\mathbf{Bisim}(X, Y)$ denote the category of bisimulations, with span morphisms between them.

Proposition 3.15. $\mathbf{Bisim}(X, Y)$ has a terminal object, called bisimilarity and denoted by $\sim_{X, Y}$.

Example 3.16. Bisimilarity on the syntactic transition system merely amounts to simultaneous convergence, because evaluation returns an open term, which does not have any further transition. In this case, a more relevant behavioural equivalence is substitution-closed bisimilarity, which we will define below.

3.3. Operational semantics. Just as we have defined the syntax as an initial Σ_0 -monoid (Example 3.3), let us now define the dynamics by initiality, again starting by finding the right notion of model. First of all, models will be found among transition systems X whose underlying presheaf $X_0 \in [\mathbb{F}, \mathbf{Set}]$ is a Σ_0 -monoid. Let us give these a name.

Definition 3.17. A *transition Σ_0 -monoid* is a transition system X , together with Σ_0 -monoid structure on its state object X_0 . Transition Σ_0 -monoids form a category $\Sigma_0\text{-Mon}$.

The idea is to model the transition rules as an endofunctor on transition Σ_0 -monoids, leaving the underlying Σ_0 -monoid untouched, i.e., a functor making the triangle

$$\begin{array}{ccc} \Sigma_0\text{-Mon} & \xrightarrow{\Sigma_1} & \Sigma_0\text{-Mon} \\ & \searrow \mathcal{D} & \swarrow \mathcal{D} \\ & \Sigma_0\text{-mon} & \end{array}$$

commute, where \mathcal{D} denotes the forgetful functor (i.e., $\mathcal{D}(X) = X_0$).

For call-by-name λ -calculus, the functor $\Sigma_1: \Sigma_0\text{-Mon} \rightarrow \Sigma_0\text{-Mon}$ modelling the non-standard rules at the end of §2 is defined as follows.

- On states, commutation of the above triangle imposes $\Sigma_1(X)_0 = X_0$.
- On transitions, let

$$\Sigma_1(X)_1 = X(1) + A_\beta(X),$$

where $A_\beta(X)$ denotes the set of valid premises for the second rule in §2.3, i.e., triples (r_1, e_2, r_2) such that

- $r_1, r_2 \in X_1$ are transitions,
- $e_2 \in X_0(0)$ is a state, and
- $r_2 \cdot s_\Downarrow = (r_1 \cdot t_\Downarrow)[e_2]$, i.e., the source $r_2 \cdot s_\Downarrow$ of r_2 is obtained by substituting e_2 for the unique free variable in the target of r_1 .

Let us notice that substitution here follows from the monoid structure of X .

- We then define the source and target maps:
 - for the first term $X(1)$,
 - * the source of any $in_1(e)$ is $\lambda_1(e)$, where $\lambda_n: X_0(n+1) \rightarrow X_0(n)$ follows from the Σ_0 -algebra structure of X_0 ;
 - * the target is e itself;
 - for the second term $A_\beta(X)$,
 - * the source of any $in_2(r_1, e_2, r_2)$ is $(r_1 \cdot s_\Downarrow) e_2$, i.e., the application of the source of r_1 to e_2 (again using the Σ_0 -algebra structure of X_0);
 - * the target is $r_2 \cdot t_\Downarrow$.

Accordingly, our notion of model is the following.

Definition 3.18. A *vertical Σ_1 -algebra* is a transition Σ_0 -monoid X equipped with a morphism $\nu_X : \Sigma_1(X) \rightarrow X$ such that $\mathcal{D}(\nu_X) = \text{id}_{X_0}$, or equivalently a map $(\nu_X)_1$ making the triangle

$$\begin{array}{ccc} \Sigma_1(X)_1 & \xrightarrow{\quad} & X_1 \\ & \searrow & \swarrow \\ & X_0(0) \times X_0(1) & \end{array} \quad \text{commute.}$$

In the case of call-by-name λ -calculus, it should be clear that such a vertical algebra is indeed a model of the rules.

However, in order to ensure that the rules are syntax-directed, we want to distinguish, for each rule, the head operator of the source of the conclusion (abstraction for the first rule; application for the second one). Instead of demanding that $\Sigma_1(X)$ have the form $\Sigma_1(X)_1 \rightarrow X_0(0) \times X_0(1)$, we thus rather require something of the form $\Sigma_1(X)_1 \rightarrow \Sigma_0(X_0)(0) \times X_0(1)$:

Definition 3.19.

- A *dynamic signature* consists of
 - a finitary functor $\Sigma_1^F : \Sigma_0\text{-Mon} \rightarrow \mathbf{Set}$, and
 - a natural transformation $(\Sigma_1^\partial)_X : \Sigma_1^F(X) \rightarrow \Sigma_0(X_0)(0) \times X_0(1)$.
- The endofunctor $\tilde{\Sigma}_1$ *induced* by a dynamic signature maps any X to the composite $\Sigma_1^F(X) \xrightarrow{(\Sigma_1^\partial)_X} \Sigma_0(X_0)(0) \times X_0(1) \rightarrow X_0(0) \times X_0(1)$.
- A *vertical algebra* of a dynamic signature is a vertical algebra of the induced endofunctor, in the sense of Definition 3.18.

Concretely, a vertical algebra is a dashed map making the following diagram commute.

$$\begin{array}{ccc} \Sigma_1^F(X) & \text{-----} & X_1 \\ \downarrow & & \downarrow \\ \Sigma_0(X_0)(0) \times X_0(1) & \longrightarrow & X_0(0) \times X_0(1) \end{array}$$

Example 3.20. For call-by-name λ -calculus, we only need to modify the source components of the above definition of Σ_1 , replacing actual operations by formal ones, like so:

- the source of any $in_1(e) \in X(1) + A_\beta(X)$ is $in_1(e) \in \Sigma_0(X)(0) = X(1) + X(0)^2$;
- the source of any $in_2(r_1, e_2, r_2)$ is $in_2((r_1 \cdot s_\Downarrow), e_2) \in \Sigma_0(X)(0)$.

This successfully captures the syntactic transition system:

Proposition 3.21. *The initial $\tilde{\Sigma}_1$ -algebra $\mathbf{Z}_{\tilde{\Sigma}_1}$, or \mathbf{Z} for short, is an initial vertical algebra, and is isomorphic to the transition system of Example 3.7.*

3.4. Substitution-closed bisimilarity. There is an obvious notion of bisimulation for transition Σ_0 -monoids:

Definition 3.22. A morphism in $\Sigma_0\text{-Mon}$ is a *functional bisimulation* iff its underlying morphism in \mathbf{C} is.

However, as foreshadowed by Example 3.16, the relevant notion in this case combines bisimulation with substitution-closedness, in the following sense.

Definition 3.23. For any monoid M in \mathbf{C}_0 , an *M -module* is an object X equipped with algebra structure $X \otimes M \rightarrow X$ for the monad $- \otimes M$. A module morphism is an algebra morphism.

Example 3.24. The monoid M is itself an M -module by multiplication, and M -modules are closed under limits in \mathbf{C}_0 , so in particular M^2 is an M -module, with action given by the composite $M^2 \otimes M \xrightarrow{\langle \pi_1 \otimes M, \pi_2 \otimes M \rangle} (M \otimes M)^2 \xrightarrow{m_M^2} M^2$.

Definition 3.25. For any transition monoid M , a span of the form $R \rightarrow M^2$ in \mathbf{C} is *substitution-closed* iff R_0 is an M_0 -module and the morphism $R_0 \rightarrow M_0^2$ is an M_0 -module morphism.

Example 3.26. To see what this definition has to do with substitution-closedness, let us observe that if R is a relation in $[\mathbf{F}, \mathbf{Set}]$, an element of $(R \otimes M)(n)$ is an explicit substitution $r[\sigma]$ with $r \in R(p)$ for some p , and $\sigma: p \rightarrow M(n)$. Now, substitution-closedness amounts to a morphism $R \otimes M \rightarrow R$ commuting with projections, so if R is a relation, we indeed get that $e R e'$ entails $e[\sigma] R e'[\sigma]$.

Proposition 3.27. *For any transition Σ_0 -monoid M , there is a terminal substitution-closed bisimulation \sim_M^\otimes , called substitution-closed bisimilarity.*

Remark 3.28. One may prove that substitution-closed bisimilarity is a relation.

Proposition 3.29. *Substitution-closed bisimilarity $\sim_{\mathbf{Z}}^\otimes$ on the syntactic transition system \mathbf{Z} coincides with applicative bisimilarity.*

Proof. Let us denote the open extension of applicative bisimilarity by \sim_{std}^\otimes , and recall that applicative bisimilarity is denoted by \sim . The relation \sim_{std}^\otimes is straightforwardly a substitution-closed bisimulation, so we have $\sim_{std}^\otimes \subseteq \sim_{\mathbf{Z}}^\otimes$. But conversely any substitution-closed bisimulation relation R (hence $\sim_{\mathbf{Z}}^\otimes$) is in particular a substitution-closed relation contained in \sim on closed terms. It is thus globally contained in \sim_{std}^\otimes by Lemma 2.4. \square

Our main result instantiates to the following.

Theorem 3.30. *Substitution-closed bisimilarity is context-closed. More precisely, it is a transition Σ_0 -monoid, and $\sim_{\mathbf{Z}}^\otimes \rightarrow \mathbf{Z}^2$ is a transition Σ_0 -monoid morphism.*

In particular, there exists a span morphism $\Sigma_0((\sim_{\mathbf{Z}}^\otimes)_0) \rightarrow (\sim_{\mathbf{Z}}^\otimes)_0$.

4. TRANSITION SYSTEMS AND BISIMILARITY

In this section, we start to abstract over the situation of §3, by introducing a general framework for transition systems and bisimilarity.

4.1. Pre-Howe contexts and transition systems.

Definition 4.1. A *pre-Howe context*³ consists of

- a small category \mathbb{C}_0 of *state types*,
- a small category \mathbb{C}_1 of *transition types*,
- and two *source* and *target* functors $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$.

³The Howe contexts of [12] may be defined similarly. The difference is that for them, \mathbf{s} and \mathbf{t} are not necessarily functorial, but $c_1 \mapsto (\mathbf{s}(c_1), \mathbf{t}(c_1))$ defines a functor $\mathbb{C}_1 \rightarrow \mathbb{C}_0 \times \mathbb{C}_0$, where $\mathbb{C}_0 \times \mathbb{C}_0$ denotes the category whose objects are pairs of elements of \mathbb{C}_0 , and morphisms between (a_1, a_2) and (b_1, b_2) consists of pairs of morphisms $a_1 \rightarrow b_i$ and $a_2 \rightarrow b_j$ for some $i, j \in \{1, 2\}$.

Precomposition by \mathbf{s} and \mathbf{t} yields functors $\Delta_{\mathbf{s}}, \Delta_{\mathbf{t}}: \widehat{\mathbb{C}}_0 \rightarrow \widehat{\mathbb{C}}_1$ mapping any $X \in \widehat{\mathbb{C}}_0$ to $X \circ \mathbf{s}$ and $X \circ \mathbf{t}$, respectively. Let Δ denote the pointwise product $\Delta_{\mathbf{s}} \times \Delta_{\mathbf{t}}$.

We use these functors to define transition systems.

Definition 4.2. Given any pre-Howe context, a *transition system* X consists of

- a *state* presheaf $X_0 \in \widehat{\mathbb{C}}_0$,
- a *transition* presheaf $X_1 \in \widehat{\mathbb{C}}_1$,
- and two *source* and *target* natural transformations $X_0 \circ \mathbf{s} \leftarrow X_1 \rightarrow X_0 \circ \mathbf{t}$, or equivalently a natural transformation $X_1 \rightarrow \Delta(X_0)$.

Proposition 4.3. *In any pre-Howe context, transition systems are precisely the objects of the lax limit category $\widehat{\mathbb{C}}_1/\Delta$ of the functor $\widehat{\mathbb{C}}_0 \xrightarrow{\Delta} \widehat{\mathbb{C}}_1$ in \mathbf{CAT} , or equivalently the comma category $\text{id}_{\widehat{\mathbb{C}}_1} \downarrow \Delta$.*

Proof. An object of the lax limit is by construction a triple (X_1, X_0, ∂) , where $\partial: X_1 \rightarrow \Delta(X_0) = X_0 \mathbf{s} \times X_0 \mathbf{t}$. \square

Notation 4.4. *In any pre-Howe context, we let $\mathbf{C}_0 := \widehat{\mathbb{C}}_0$ and $\mathbf{C} := \widehat{\mathbb{C}}_1/\Delta$.*

Proposition 4.5. *The projection functor $-_0: \mathbf{C} \rightarrow \mathbf{C}_0$ has a left adjoint mapping any object X_0 to $\emptyset \rightarrow X_0 \mathbf{s} \times X_0 \mathbf{t}$, which we call the discrete transition system on X_0 .*

Example 4.6. We can get $\mathbf{C} \rightarrow \mathbf{C}_0$ to be the forgetful functor $\mathbf{Gph} \rightarrow \mathbf{Set}$ by taking

- $\mathbf{C}_0 = 1$, so that $\mathbf{C}_0 = \widehat{\mathbb{C}}_0 = \widehat{1} \cong \mathbf{Set}$,
- $\mathbf{C}_1 = 1$, so that $\mathbf{C} = \mathbf{Set}$, and
- $\mathbf{s}, \mathbf{t}: 1 \rightarrow 1$ to be the unique such functor, i.e., the identity.

A transition system thus consists of sets V and E together with a map $E \rightarrow V^2$, i.e., a graph.

Example 4.7. A (harmless) proof-relevant variant of standard labelled transition systems (over any set \mathbb{A} of labels) may be obtained as follows. We take

- $\mathbf{C}_0 = 1$ again,
- $\mathbf{C}_1 = \mathbb{A}$ viewed as a discrete category, and
- $\mathbf{s}, \mathbf{t}: \mathbf{C}_1 \rightarrow \mathbf{C}_0$ the unique such functor.

Thus, a transition system X consists of a set X_0 and sets X_a for all $a \in \mathbb{A}$, together with maps $X_a \rightarrow X_0^2$ returning the source and target of each a -labelled edge.

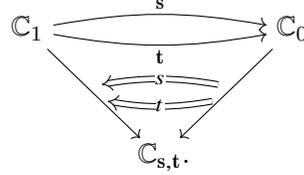
More generally, given any graph \mathbb{L} , taking $\mathbf{s}, \mathbf{t}: \mathbf{C}_1 \rightarrow \mathbf{C}_0$ to be the source and target maps $\mathbb{L}_1 \rightarrow \mathbb{L}_0$ viewed as functors between discrete categories, we obtain for $\mathbf{C} \rightarrow \mathbf{C}_0$ a functor equivalent to $\mathbf{Gph}/\mathbb{L} \rightarrow \mathbf{Set}/\mathbb{L}_0$.

Example 4.8. Let $\mathbf{C}_0 = \mathbb{F}^{op}$ and $\mathbf{C}_1 = 1$, with \mathbf{s} and \mathbf{t} picking respectively 0 and 1. In particular, $\widehat{\mathbb{C}}_1 \cong \mathbf{Set}$. Then, $\Delta(X_0) = X_0(0) \times X_0(1)$ and we recover the category \mathbf{C} of §3.2, and its forgetful functor to $\mathbf{C}_0 = [\mathbb{F}, \mathbf{Set}]$.

4.2. Transition systems as presheaves. Before introducing bisimulation, let us establish an alternative characterisation of the category \mathbf{C} of transition systems.

Proposition 4.9. *The lax limit category $\widehat{\mathbb{C}}_1/\Delta$ of transition systems is isomorphic to a presheaf category $\widehat{\mathbb{C}}_{\mathbf{s}, \mathbf{t}}$.*

Proof. Let $\mathbb{C}_{s,t}$ denote the lax colimit in \mathbf{Cat} of the parallel pair s, t . By definition, it is the universal category with natural transformations



It thus consists of the coproduct $\mathbb{C}_1 + \mathbb{C}_0$, augmented with arrows $s_L: s(L) \rightarrow L$ and $t_L: t(L) \rightarrow L$ for all $L \in \mathbb{C}_1$, naturally in L . Presheaves on $\mathbb{C}_{s,t}$ coincide with $\widehat{\mathbb{C}_1/\Delta}$ because the presheaf construction turns lax colimits into lax limits. \square

Notation 4.10. We often omit the isomorphism $\widehat{\mathbb{C}_1/\Delta} \cong \widehat{\mathbb{C}_{s,t}}$, considering it as an implicit coercion. E.g., \mathbf{y}_P may be used to denote the transition system \underline{P} with $\underline{P}_1 = \emptyset$ and $\underline{P}_0 = \mathbf{y}_P$.

Similarly, \mathbf{y}_L may be used to denote the ‘minimal’ transition system with one transition over L , say \underline{L} , i.e., $\underline{L}_1 = \mathbf{y}_L$, $\underline{L}_0 = \mathbf{y}_{s(L)} + \mathbf{y}_{t(L)}$, and the map $\underline{L}_1 \rightarrow \underline{L}_0 s \times \underline{L}_0 t$ uniquely determined by the element $(in_1(\text{id}_{s(L)}), in_2(\text{id}_{t(L)})) \in \underline{L}_0(s(L)) \times \underline{L}_0(t(L))$.

Finally, $\mathbf{y}_{s_L}: \mathbf{y}_{s(L)} \rightarrow \mathbf{y}_L$ and $\mathbf{y}_{t_L}: \mathbf{y}_{t(L)} \rightarrow \mathbf{y}_L$ denote the Yoneda embedding of the canonical morphisms s_L and t_L from the proof of Proposition 4.9.

By Yoneda, we thus have:

Corollary 4.11. For all X , we have $\mathbf{C}(\mathbf{y}_L, X) \cong X_1(L)$ and $\mathbf{C}(\mathbf{y}_P, X) \cong X_0(P)$.

Notation 4.12. In the case of call-by-name λ -calculus, we call \Downarrow the unique object coming from $\mathbb{C}_1 = 1$.

Remark 4.13. Presheaves on $\mathbb{C}_{s,t}$ are intuitively 2-dimensional; the projection functor forgets dimension 1, while the left adjoint (Proposition 4.5) adds an empty dimension 1, thus lifting its 0-dimensional argument to a 1-dimensional object.

This 2-dimensional intuition leads to the following useful observation on the forgetful functor.

Proposition 4.14. The forgetful functor $\mathbf{C} \rightarrow \mathbf{C}_0$ preserves all colimits, as well as image (in the sense of strong epi, mono) factorisations.

Proof. The forgetful functor $\mathbf{C} \rightarrow \mathbf{C}_0$ is equivalent to the restriction functor $\widehat{\mathbb{C}_{s,t}} \rightarrow \widehat{\mathbb{C}_0}$, which is both a left and right adjoint, hence preserves all limits and colimits. Finally, image factorisations are computed pointwise in presheaf categories, hence are preserved by restriction functors. \square

4.3. Bisimulation and bisimilarity. Morphisms in \mathbf{C} are a generalisation of graph morphisms, which are a proof-relevant version of functional simulations. The analogue of functional bisimulations is as follows.

Definition 4.15. A morphism $f: X \rightarrow Y$ in $\widehat{\mathbb{C}_{s,t}}$ is a *functional bisimulation*, or a *fibration*, iff it enjoys the (weak) right lifting property w.r.t. $\mathbf{y}_{s_L}: \mathbf{y}_{s(L)} \rightarrow \mathbf{y}_L$, for all $L \in \mathbb{C}_1$.

Here is a characterisation of fibrations which will be important. Let us recall that a weak pullback satisfies the same universal property as a pullback, albeit without uniqueness.

Proposition 4.16. *A morphism $f: X \rightarrow Y$ is a functional bisimulation iff the following diagram is a pointwise weak pullback.*

$$\begin{array}{ccc} X_1 & \xrightarrow{f_1} & Y_1 \\ \downarrow & & \downarrow \\ X_0\mathbf{s} & \xrightarrow{f_0\mathbf{s}} & Y_0\mathbf{s} \end{array}$$

Proof. By Yoneda, a lifting problem in \mathbf{C} as on the left below is the same as a cone in $\widehat{\mathbf{C}}_1$ as on the right, and a lifting is the same as a mediating morphism to X_1 .

$$\begin{array}{ccc} s(L) \longrightarrow X & & \mathbf{y}_L \xrightarrow{\quad} X_1 \longrightarrow Y_1 \\ \downarrow \quad \dashrightarrow \quad \downarrow & & \downarrow \quad \quad \downarrow \\ L \longrightarrow Y & & X_0\mathbf{s} \xrightarrow{f_0\mathbf{s}} Y_0\mathbf{s} \end{array} \quad \square$$

We use this to define general bisimulations. Usually, one considers bisimulation relations. Here, we generalise this a bit and consider arbitrary spans:

Definition 4.17. A *simulation* is a span $X \leftarrow R \rightarrow Y$ whose left leg is a fibration. A *bisimulation* is a span of fibrations (equivalently, a simulation whose converse span is also a simulation).

Remark 4.18. Of course, the relevant notion in our applications is substitution-closed bisimulation, to which we will come below.

Lemma 4.19. *Simulation relations and bisimulation relations are stable under unions.*

Proof. By symmetry, it is enough to deal with the case of simulation relations. Consider any family $(R_i \hookrightarrow X \times Y)_{i \in I}$ of simulation relations. Their union is the image of their copairing. But because the domain $\mathbf{y}_{s(L)}$ of s_L is representable for all $L \in \widehat{\mathbf{C}}_1$, any lifting problem $s_L \rightarrow \bigcup_i R_i$ lifts to a lifting problem $s_L \rightarrow \sum_i R_i$, which in turn lifts to a lifting problem to some $s_L \rightarrow R_{i_0}$ with $i_0 \in I$. We then find a lifting for the latter, which yields a lifting for the original. \square

Proposition 4.20. *For all $X, Y \in \mathbf{C}$, the full subcategory $\mathbf{Bisim}(X, Y)$ of spans between X and Y which are bisimulations admits a terminal object $\sim_{X, Y}$, called bisimilarity.*

Proof. As a presheaf category by Proposition 4.9, \mathbf{C} is well-powered, so we may consider the union $\sim_{X, Y}$ of all bisimulation relations, which is again a bisimulation by Lemma 4.19. Finally, $\sim_{X, Y}$ is terminal, because any bisimulation R factors through its image $im(R)$, which is again a bisimulation; as a bisimulation relation, $im(R)$ thus embeds into $\sim_{X, Y}$, hence we obtain a morphism $R \twoheadrightarrow im(R) \hookrightarrow \sim_{X, Y}$, which is unique by monicity of $\sim_{X, Y} \hookrightarrow X \times Y$. \square

5. HOWE CONTEXTS FOR OPERATIONAL SEMANTICS

Operational semantics is the combination of syntax and transition systems, in the sense that it is about transition systems whose states form a model of a certain syntax. Our framework for operational semantics thus combines the frameworks of Fiore et al. [20] for syntax with variable binding, and of §4 for transition systems.

The former is merely an abstract version of §3.1. Here is the important result, which echoes Proposition 3.2.

Proposition 5.1 (see [20, 17, 12]). *For any finitary, pointed strong endofunctor Σ_0 on a monoidal, cocomplete category \mathcal{C}_0 such that the tensor preserves all colimits on the left and filtered colimits on the right, the forgetful functor $\Sigma_0\text{-mon} \rightarrow \mathcal{C}_0$ is monadic, and the free Σ_0 -algebra over I (equivalently the initial $(I + \Sigma_0)$ -algebra) is an initial Σ_0 -monoid.*

Proof. This has been proved in Coq [12]. \square

Notation 5.2. *We denote the initial Σ_0 -monoid by \mathbf{Z}_{Σ_0} , or \mathbf{Z}_0 for short.*

Definition 5.3. A *Howe context* consists of a pre-Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$, together with a monoidal structure on $\widehat{\mathbb{C}}_0$, such that the tensor preserves all colimits on the left and filtered colimits on the right.

Notation 5.4. *As for pre-Howe contexts, we let $\mathbf{C}_0 = \widehat{\mathbb{C}}_0$ and $\mathbf{C} := \widehat{\mathbb{C}}_1/\Delta$.*

Let us assume that some syntax has been specified by a finitary, pointed strong endofunctor Σ_0 on \mathbf{C}_0 . We then define transition Σ_0 -monoids just as in §3.

Definition 5.5. The category $\Sigma_0\text{-Mon}$ of *transition Σ_0 -monoids* is the following pullback in **CAT**.

$$\begin{array}{ccc} \Sigma_0\text{-Mon} & \xrightarrow{\mathcal{D}} & \Sigma_0\text{-mon} \\ \mathcal{U} \downarrow \lrcorner & & \downarrow \\ \mathbf{C} & \longrightarrow & \mathbf{C}_0 \end{array}$$

When Σ_0 is the constantly empty endofunctor, we speak of *transition monoids*: they consist of objects X equipped with monoid structure on X_0 .

Proposition 5.6. *The adjunction between \mathbf{C} and \mathbf{C}_0 (Proposition 4.5) lifts to an adjunction*

$$\Sigma_0\text{-mon} \begin{array}{c} \xleftarrow{\mathcal{M}} \\ \perp \\ \xrightarrow{\mathcal{D}} \end{array} \Sigma_0\text{-Mon}$$

with $\mathcal{D}(\mathcal{M}(X_0)) = X_0$ and $\mathcal{M}(X_0)_1 = \emptyset$.

Proof. The left adjoint \mathcal{M} maps any Σ_0 -monoid M to the discrete transition system on M , say ΔM , equipped with the original Σ_0 -monoid structure on M – which makes sense because $(\Delta M)_0 = M$. \square

Remark 5.7. The names, \mathcal{M} and \mathcal{D} , stand for “monter” and “descendre”, “go up” and “go down” in French.

Proposition 5.8. *The forgetful functor $\mathcal{U}: \Sigma_0\text{-Mon} \rightarrow \mathbf{C}$ is monadic.*

Proof. Transition Σ_0 -monoids are the algebras of an equational system over \mathbf{C} in the sense of Fiore and Hur, to which [16, Theorem 6.1] applies. \square

Notation 5.9. *We denote by \mathcal{L} the left adjoint to \mathcal{U} .*

Similarly, we define abstract dynamic signatures, which abstract over those of Definition 3.19:

Definition 5.10. Given a Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$ and a pointed strong $\Sigma_0: \mathbf{C}_0 \rightarrow \mathbf{C}_0$, a *dynamic signature* $\Sigma_1 = (\Sigma_1^F, \Sigma_1^\partial)$ over Σ_0 consists of a finitary functor $\Sigma_1^F: \Sigma_0\text{-Mon} \rightarrow \widehat{\mathbb{C}}_1$, together with a natural transformation Σ_1^∂ with components $\Sigma_1^F(X) \rightarrow \Sigma_0(X_0)\mathbf{s} \times X_0\mathbf{t}$.

Let us pack up the static and dynamic notions of signature.

Definition 5.11. An *operational semantics signature* (Σ_0, Σ_1) on a given Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$ consists of a pointed strong endofunctor Σ_0 preserving sifted colimits, together with a dynamic signature Σ_1 over it.

In the following, we fix an operational semantics signature (Σ_0, Σ_1) on a Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$.

Remark 5.12. Preservation of sifted colimits [3] is stronger than finitariness for Σ_0 . We need it for Lemma 9.41 below. In a presheaf category like \mathbf{C}_0 , if Σ_0 preserves pullbacks (for example, by familiarity), it is equivalent to being finitary and preserving all epis, as seen from the proof of [4, Theorem 18.1].

Remark 5.13. In [12], we mistakenly only require Σ_0 to be finitary, which yields a gap in the proof of [12, Lemma 5.13].

Example 5.14. The endofunctor $\Sigma_0(X)(n) = X(n+1) + X(n)^2$ on $[\mathbb{F}, \mathbf{Set}]$ preserves sifted colimits. It is easily deduced from the fact that sifted colimits are the ones commuting with products in sets.

Let us now define the category of models of a dynamic signature Σ_1 .

Definition 5.15. Let $\check{\Sigma}_1: \Sigma_0\text{-Mon} \rightarrow \Sigma_0\text{-Mon}$ map any transition Σ_0 -monoid X to the composite

$$\Sigma_1^F(X) \rightarrow \Sigma_0(X_0)\mathbf{s} \times X_0\mathbf{t} \xrightarrow{\nu_{X_0}\mathbf{s} \times X_0\mathbf{t}} X_0\mathbf{s} \times X_0\mathbf{t},$$

where ν_{X_0} denotes the Σ_0 -algebra structure of X_0 .

Proposition 5.16. *The endofunctor $\check{\Sigma}_1$ is finitary and makes the following triangle commute.*

$$\begin{array}{ccc} \Sigma_0\text{-Mon} & \xrightarrow{\check{\Sigma}_1} & \Sigma_0\text{-Mon} \\ & \searrow & \swarrow \\ & \Sigma_0\text{-mon} & \end{array}$$

Proof. Commutativity of the triangle holds by construction, and finitariness follows from finitariness of Σ_1^F . \square

Definition 5.17. A $\check{\Sigma}_1$ -algebra $\check{\Sigma}_1(X) \rightarrow X$ is *vertical* when its image under the forgetful functor $\Sigma_0\text{-Mon} \rightarrow \Sigma_0\text{-mon}$ is the identity. Let $\Sigma_1\text{-alg}_v$ denote the full subcategory of $\check{\Sigma}_1\text{-alg}$ spanning all vertical algebras.

Theorem 5.18. *The forgetful functor $\Sigma_1\text{-alg}_v \rightarrow \Sigma_0\text{-Mon}$ is monadic, and furthermore the initial $\check{\Sigma}_1$ -algebra $\mathbf{Z}_{\check{\Sigma}_1}$, or \mathbf{Z} for short, may be chosen to be vertical, hence is also initial in $\Sigma_1\text{-alg}_v$.*

Proof. For the first statement, vertical algebras may be specified as an equational system, in the sense of [16]. For the second statement, \mathbf{Z} is the colimit of the initial chain

$$\mathbf{Z}_0 \rightarrow \check{\Sigma}_1(\mathbf{Z}_0) \rightarrow \dots \rightarrow \check{\Sigma}_1^n(\mathbf{Z}_0) \rightarrow \dots$$

(where \mathbf{Z}_0 is shorthand for $\mathcal{M}(\mathbf{Z}_0)$, for readability, recalling Proposition 5.6). The image of this chain in $\Sigma_0\text{-mon}$ is the everywhere-identity chain on \mathbf{Z}_0 . \square

By construction, we have:

Proposition 5.19. *We have $\mathcal{D}(\mathbf{Z}) = \mathbf{Z}_0$.*

Let us readily annihilate any hope that vertical Σ_1 -algebras are monadic over \mathbf{C} .

Lemma 5.20. *Consider the operational semantics signature of λ -calculus with Σ_0 from (3.2) and Σ_1 from Example 3.20. Then, the composite forgetful functor $\Sigma_1\text{-alg}_v \rightarrow \Sigma_0\text{-Mon} \rightarrow \mathbf{C}$ is not monadic.*

Proof. Let us draw inspiration from the classical example of a non-monadic composite of monadic functors, namely the composite $\mathbf{Cat} \rightarrow \mathbf{RGph} \rightarrow \mathbf{Set}$, where \mathbf{RGph} denotes the category of reflexive graphs, and the forgetful functor to sets returns the set of arrows. One may show that both functors are monadic, but that their composite is not. By Beck's monadicity theorem, it suffices to find a parallel pair $X \rightrightarrows Y$ in \mathbf{Cat} whose image in sets admits an absolute coequaliser, and show that this coequaliser is not created by the composite forgetful functor. Let us describe a classical example, slightly informally. The idea is to take Y to consist of two arrows

$$1 \xrightarrow{a} 2 \qquad 3 \xrightarrow{b} 4,$$

and $X = 1 + Y$ to have an additional object, say 0. We then define $u, v: X \rightarrow Y$ to be the identity on Y , and respectively map 0 to 2 and 3. The coequalisers in \mathbf{Cat} (top) and \mathbf{Set} (bottom) look as follows, abbreviating each id_C to just C for readability.

$$\begin{array}{ccc} \begin{array}{c} 0 \quad 1 \xrightarrow{a} 2 \quad 3 \xrightarrow{b} 4 \\ \begin{array}{c} \xrightarrow{u} \\ \xrightarrow{v} \end{array} \end{array} & \begin{array}{c} 1 \xrightarrow{a} 2 \quad 3 \xrightarrow{b} 4 \\ \xrightarrow{\quad} \end{array} & \begin{array}{c} 1 \xrightarrow{a} \{2, 3\} \xrightarrow{b} 4 \\ \xrightarrow{b \circ a} \end{array} \\ \begin{array}{c} 0 \quad 1 \quad a \quad 2 \quad 3 \quad b \quad 4 \\ \begin{array}{c} \xrightarrow{\quad} \\ \xrightarrow{\quad} \end{array} \end{array} & \begin{array}{c} 1 \quad a \quad 2 \quad 3 \quad b \quad 4 \\ \xrightarrow{\quad} \end{array} & \begin{array}{c} 1 \quad a \quad \{2, 3\} \quad a \quad 4 \\ \xrightarrow{\quad} \end{array} \end{array}$$

One easily proves that the one in sets is split, hence absolute. Briefly, because of the composite arrow $b \circ a$ that the coequaliser in \mathbf{Cat} must have but the one in sets does not, the coequaliser is not created by the forgetful functor, thus contradicting monadicity.

For proving the lemma, we rely on the β -rule to mimic composition in constructing the following parallel pair $X \rightrightarrows Y$ in $\Sigma_1\text{-alg}_v$.

- Y is the vertical Σ_1 -algebra defined as the λ -calculus extended with two constants a and b , unary operations k and l , and an axiom $b \Downarrow l(x)$;
- X is the vertical Σ_1 -algebra extending Y with a constant c ;
- the Σ_1 -algebra morphisms $u, v: X \rightarrow Y$ respectively replace c with b and $k(a)$.

The coequaliser E of these two morphisms computed in the presheaf category \mathbf{C} is a quotient of Y by the equation $b = k(a)$. It thus has as reductions $[e] \Downarrow [f]$ between equivalence classes all reductions $e' \Downarrow f'$ between representatives $e' \in [e]$ and $f' \Downarrow [f]$. E.g., it has a reduction $[k(a)] \Downarrow [l(x)]$, since $b \in [k(a)]$ and $b \Downarrow l(x)$. However, E lacks a reduction $[(\lambda x.k(x)) a] \Downarrow [l(x)]$. Indeed, $[(\lambda x.k(x)) a]$ has a unique representative, namely $(\lambda x.k(x)) a$, whose evaluation in Y gets stuck at $k(a)$. However, the coequaliser in $\Sigma_1\text{-alg}_v$ has such a reduction by applying the β -rule:

$$\frac{[(\lambda x.k(x)) a] \Downarrow [k(x)] \quad [k(a)] = [b] \Downarrow [l(x)]}{[(\lambda x.k(x)) a] \Downarrow [l(x)]}.$$

Thus, the coequaliser is not created by the forgetful functor.

Finally, this coequaliser is split (hence absolute): the morphism $v: X \rightarrow Y$ replacing c with $k(a)$ has a section $f: Y \rightarrow X$ replacing $k(a)$ with c , and the coequaliser arrow $e: Y \rightarrow Z$ has a section $g: Z \rightarrow Y$ replacing $k(a)$ with b . It is straightforward to check that this indeed defines a split coequaliser, i.e., that $g \circ e = u \circ f$. \square

6. SUBSTITUTION-CLOSED BISIMILARITY

We have now defined bisimilarity (Proposition 4.20) for our generalised transition systems, and also introduced operational semantics signatures, which allow us to efficiently specify relevant syntactic transition systems. In this section, we introduce substitution-closed bisimilarity in the general setting, and state our main theorem, for which we include a high-level proof sketch.

In order to introduce substitution-closed bisimilarity, we first lift the notion of bisimulation to Σ_0 -monoids, following §3.4:

Definition 6.1. A morphism in $\Sigma_0\text{-Mon}$ is a *functional bisimulation* iff its underlying morphism in \mathbf{C} is. A span is a *simulation* iff its left leg is a functional bisimulation, and a *bisimulation* iff both of its legs are functional bisimulations.

Let us readily prove the following characterisation by lifting, recalling from Notation 5.9 that $\mathcal{L}: \mathbf{C} \rightarrow \Sigma_0\text{-Mon}$ is left adjoint to the forgetful functor.

Proposition 6.2. *A morphism in $\Sigma_0\text{-Mon}$ is a functional bisimulation iff it has the right lifting property w.r.t. all $\mathcal{L}(\mathbf{y}_{s_L}): \mathcal{L}(\mathbf{y}_{s(L)}) \rightarrow \mathcal{L}(\mathbf{y}_L)$, for $L \in \mathbb{C}_1$.*

Proof. By adjunction. \square

As seen in Example 3.16, we now want to go beyond bisimilarity, and introduce abstract versions of substitution-closed bisimulation and bisimilarity. For this, let us give the general definition of modules over a monoid.

Definition 6.3. For any monoid M in a monoidal category \mathcal{C} , let $M\text{-Mod}$ denote the category of algebras for the monad $- \otimes M$.

An M -module thus consists of an object X equipped with an action $X \otimes M \rightarrow X$ of M satisfying straightforward coherence conditions.

Example 6.4. M itself is an M -module, with action given by multiplication.

Before going into substitution-closed bisimulation, let us record the following useful properties of modules in a Howe context.

Proposition 6.5. *In any Howe context,*

- *for all monoids M in \mathbf{C}_0 , the forgetful functor $M\text{-Mod} \rightarrow \mathbf{C}_0$ creates all limits and colimits, and furthermore*
- *for all monoids M in \mathbf{C}_0 , the category $M\text{-Mod}$ is regular and the forgetful functor $M\text{-Mod} \rightarrow \mathbf{C}_0$ creates image factorisations.*

Proof. For creation of limits and colimits:

- As algebras for the monad $- \otimes M$, M -modules are closed under limits.

- They are also closed under all types of colimits preserved by $- \otimes M$, i.e., all of them by definition of Howe contexts.

Thus, $M\text{-Mod}$ is complete and cocomplete, hence regularity reduces to showing that the pullback of any regular epi is again a regular epi. So let us consider any pullback square

$$\begin{array}{ccc} A & \xrightarrow{u} & B \\ v \downarrow & \lrcorner & \downarrow f \\ C & \xrightarrow{g} & D \end{array}$$

in $M\text{-Mod}$, with f a regular epi, and show that v must also be a regular epi. By creation, hence preservation, of limits and colimits, the given pullback square is also a pullback in \mathbf{C}_0 and f is a regular epi there too. So by regularity of the presheaf category \mathbf{C}_0 , v is a regular epi in \mathbf{C}_0 . Equivalently, it is a coequaliser of its kernel pair. But by creation of limits the kernel pair uniquely lifts to a kernel pair in $M\text{-Mod}$, and by creation of colimits v is a coequaliser there too. This shows that $M\text{-Mod}$ is regular.

Finally, given $X, Z \in M\text{-Mod}$, let us consider any image factorisation $X \xrightarrow{e} Y \xrightarrow{m} Z$ in \mathbf{C}_0 of a morphism $f: X \rightarrow Z$ in $M\text{-Mod}$, i.e., e is a regular epi and m is a mono in \mathbf{C}_0 . In this situation, e is the coequaliser of its kernel pair in \mathbf{C}_0 , but, as we just saw, this kernel pair lifts to a kernel pair in $M\text{-Mod}$, whose coequaliser is created by the forgetful functor, hence e is a coequaliser, hence a regular epi in $M\text{-Mod}$. Finally, f also coequalises the kernel pair, hence the existence of a unique mediating morphism $Y \rightarrow Z$ in $M\text{-Mod}$, which must be m by uniqueness upon projection in \mathbf{C}_0 . Thus, m is also a morphism in $M\text{-Mod}$. Finally, its monicity follows from the fact that the forgetful functor creates pullbacks. \square

Let us now introduce substitution-closed spans, first in an arbitrary monoidal category, and then in a Howe context. This then leads us to substitution-closed bisimulations.

Definition 6.6. In a monoidal category \mathcal{C} with binary products, given a monoid M and M -modules X and Y , a *substitution-closed span* is a span $R \rightarrow X \times Y$ equipped with M -module structure on R , such that $R \rightarrow X \times Y$ is an M -module morphism.

The last condition is equivalent to commutation of the following diagram.

$$\begin{array}{ccc} R \otimes M & \xrightarrow{\quad\quad\quad} & R \\ \downarrow & & \downarrow \\ (X \times Y) \otimes M & \xrightarrow{\langle \pi_1 \otimes M, \pi_2 \otimes M \rangle} & (X \otimes M) \times (Y \otimes M) \xrightarrow{a_X \times a_Y} X \times Y \end{array}$$

Definition 6.7. Consider any Howe context $\mathbf{s}, \mathbf{t}: \mathbf{C}_1 \rightarrow \mathbf{C}_0$ and transition monoid $M \in \mathbf{C}$. Let $X, Y \in \mathbf{C}$ be equipped with M_0 -module structure on X_0 and Y_0 . A *substitution-closed span* is a span $R \rightarrow X \times Y$ equipped with substitution-closed structure on $R_0 \rightarrow X_0 \times Y_0$.

Definition 6.8. For any Howe context $\mathbf{s}, \mathbf{t}: \mathbf{C}_1 \rightarrow \mathbf{C}_0$ and transition monoid $M \in \mathbf{C}$, a *substitution-closed bisimulation* is a substitution-closed span $R \rightarrow M^2$ (viewing M_0 itself as an M_0 -module) which is a bisimulation.

Let $\mathbf{Bisim}^\otimes(M)$ denote the full subcategory of \mathbf{C}/M^2 spanning substitution-closed bisimulations.

Let us now prove the existence of substitution-closed bisimilarity.

Lemma 6.9. *Substitution-closed simulation and bisimulation relations are stable under unions.*

Proof. By Lemma 4.19, the union of a family of substitution-closed simulation (resp. bisimulation) relations is again a simulation (resp. bisimulation) relation. But by Proposition 6.5, the union in \mathbf{C} is again substitution-closed, which concludes the proof. \square

Proposition 6.10. *For any Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$ and monoid $M \in \mathbf{C}$, the category $\mathbf{Bisim}^\otimes(M)$ of substitution-closed bisimulations over M admits a terminal object \sim_M^\otimes , called substitution-closed bisimilarity.*

Proof. Straightforward generalisation of the proof of Proposition 4.20 using the lemma. \square

Notation 6.11. *When $M = \mathbf{Z}$, we abbreviate $\sim_{\mathbf{Z}}^\otimes$ to just \sim^\otimes .*

We now want to state the abstract version of our main theorem, but we need an additional hypothesis, which we now introduce. The idea is essentially that Σ_1 should preserve functional bisimulations, which does not quite make sense, because the codomain of Σ_1^F is $\widehat{\mathbb{C}}_1$, where no notion of functional bisimulation has been defined yet. Recalling Proposition 4.16, we work around this as follows.

Definition 6.12. A dynamic signature $\Sigma_1 = (\Sigma_1^F, \Sigma_1^\partial)$ *preserves functional bisimulations* iff for any functional bisimulation $R \rightarrow X$ in Σ_0 - \mathbf{Mon} , the following square is a pointwise weak pullback.

$$\begin{array}{ccc} \Sigma_1(R) & \longrightarrow & \Sigma_1(X) \\ \pi_1 \circ (\Sigma_1^\partial)_R \downarrow & & \downarrow \pi_1 \circ (\Sigma_1^\partial)_X \\ \Sigma_0(R_0)\mathbf{s} & \longrightarrow & \Sigma_0(X_0)\mathbf{s} \end{array} \quad (6.1)$$

Remark 6.13. It may not be obvious that the dynamic signature for call-by-name λ -calculus preserves functional bisimulations. We will come back to this in §7 by showing that it satisfies a sufficient condition, cellularity.

Remark 6.14. It may seem linguistically inappropriate to say that Σ_1 preserves functional bisimulations, since we have not even defined fibrations in the codomain category $\widehat{\mathbb{C}}_1$. We will justify this in Proposition 7.7, but for now let us move on directly to the main result.

Theorem 6.15. *If Σ_1 preserves functional bisimulations, then substitution-closed bisimilarity is context-closed. More precisely, it is a transition Σ_0 -monoid, and $\sim^\otimes \rightarrow \mathbf{Z}^2$ is a transition Σ_0 -monoid morphism.*

Proof sketch (see §9 for the full proof).

1. We first define the *Howe closure* H_0 of applicative bisimilarity \sim_0^\otimes on states as the initial Σ_0^H -monoid for the pointed strong endofunctor Σ_0^H on $\mathbf{C}_0/\mathbf{Z}^2$ defined by $\Sigma_0^H(R) = \Sigma_0(R) + (R; \sim_0^\otimes)$. By construction, H_0 is a Σ_0 -monoid and both projections are Σ_0 -monoid morphisms.

2. We then define the *transition Howe closure* H of (the full) applicative bisimilarity \sim^\otimes , as an initial algebra for an endofunctor Σ_1^H on a suitable category \mathbf{C}_Z^H . Very roughly, \mathbf{C}_Z^H is the category of spans $R \rightarrow \mathbf{Z}^2$ whose projection is precisely $H_0 \rightarrow \mathbf{Z}_0^2$, and $\Sigma_1^H(R) = \check{\Sigma}_1(R) + (R; \sim^\otimes)$. We show:

Lemma 6.16. *There exists a span morphism $\sim^\otimes \rightarrow H$.*

3. Next comes the key lemma:

Lemma 6.17. *If Σ_1 preserves functional bisimulations, then the transition Howe closure H is a substitution-closed simulation.*

Remark 6.18. Since H_0 is a Σ_0 -monoid by construction, H is easily seen to be substitution-closed, so the lemma really is about it being a simulation.

The key lemma is proved by characterising H as an initial algebra for a different end-ofunctor on a different category, whose initial chain involves iterated applications of Σ_1 (preserving simulations by hypothesis) to $\pi_1: \mathcal{M}(H_0) \rightarrow \mathcal{M}(\mathbf{Z}_0)$, which is trivially a simulation.

4. In the standard proof method, the next step is to prove that the transitive closure of H_0 is symmetric. But in our case H_0 is a general span, not a relation. In order to avoid some coherence issues, we introduce a suitable notion of *relational* transitive closure for general spans, denoted by $-\bar{\tau}$, for which we show:

Lemma 6.19. *The relational transitive closure $H_0^{\bar{\tau}}$ of the Howe closure H_0 on states is symmetric.*

As substitution-closed simulations are closed under transitive closure, we obtain

Corollary 6.20. *$H^{\bar{\tau}}$ is a substitution-closed simulation which is symmetric on states.*

We then use the following lemma (proved in §9.7).

Lemma 6.21. *For any substitution-closed simulation R such that R_0 is symmetric, there exists a span morphism $R \rightarrow R'$ such that R' is a substitution-closed bisimulation.*

By terminality of \sim^\otimes , we thus get a unique morphism $H^{\bar{\tau}'} \rightarrow \sim^\otimes$ over \mathbf{Z}^2 .

5. From the chain

$$\sim^\otimes \rightarrow H \rightarrow H^{\bar{\tau}} \rightarrow H^{\bar{\tau}'} \rightarrow \sim^\otimes$$

we get by terminality that \sim^\otimes is a retract of a transition Σ_0 -monoid, namely H . The result then readily follows from monadicity of transition Σ_0 -monoids (Proposition 5.8) and the following result, taking $X = H$, $Y = \sim^\otimes$, and $Z = \mathbf{Z}^2$. □

Lemma 6.22. *Consider a monad $T: \mathcal{C} \rightarrow \mathcal{C}$ on any category \mathcal{C} , T -algebras X and Z , and morphisms $X \xrightarrow{e} Y \xrightarrow{m} Z$ in \mathcal{C} such that the composite is a T -algebra morphism, e is a split epi, and m is monic. Then there is a unique T -algebra structure on Y such that e and m both are T -algebra morphisms.*

Proof. Let $s: Y \rightarrow X$ denote any section of e . The desired structure is given by

$$T(Y) \xrightarrow{T(s)} T(X) \rightarrow X \xrightarrow{e} Y,$$

and the rest follows by monicity of m . □

7. PRESERVATION OF FUNCTIONAL BISIMULATIONS, AND CELLULARITY

Let us now consider the main hypothesis of Theorem 6.15, preservation of functional bisimulations. In §7.1, we rephrase the condition in a way that makes more sense linguistically, i.e., by an actual preservation condition. We then work towards a characterisation in terms of cellularity. In §7.2, we first briefly recall *familial* functors [15, 13], and show that the dynamic signature for call-by-name λ -calculus induces familial functors. In §7.3, we restrict attention to the case where both components of the dynamic signature are familial, and show that preservation of functional bisimulations is then equivalent to a *cellularity* condition [21, 12], which itself comes with a useful sufficient condition.

7.1. An alternative characterisation. Let us first give an alternative definition of dynamic signatures.

Definition 7.1. Let $\widehat{\mathbb{C}}_1/\Delta_s$ denote the following lax limit category.

$$\begin{array}{ccc} & \widehat{\mathbb{C}}_1/\Delta_s & \\ \swarrow & \leftarrow & \searrow \\ \widehat{\mathbb{C}}_0 & \xrightarrow{\Delta_s} & \widehat{\mathbb{C}}_1 \end{array}$$

Concretely, an object consists of presheaves X_1 and X_0 , together with a morphism $X_1 \rightarrow X_0$. Just as \mathbf{C} , $\widehat{\mathbb{C}}_1/\Delta_s$ is in fact a presheaf category:

Proposition 7.2. $\widehat{\mathbb{C}}_1/\Delta_s$ is isomorphic to the presheaf category over the lax colimit \mathbb{C}_s of the functor $s: \mathbb{C}_1 \rightarrow \mathbb{C}_0$.

Remark 7.3. Concretely, \mathbb{C}_s is the coproduct of \mathbb{C}_0 and \mathbb{C}_1 , augmented with morphisms $s_L: s(L) \rightarrow L$ for all $L \in \mathbb{C}_1$, naturally in L .

Notation 7.4. In the case of call-by-name λ -calculus, as in \mathbf{C} (Notation 4.12), we call \Downarrow the unique object coming from $\mathbb{C}_1 = 1$.

Proposition 7.5. Any operational semantics signature (Σ_0, Σ_1) induces a finitary functor $\Sigma_1^s: \Sigma_0\text{-Mon} \rightarrow \widehat{\mathbb{C}}_s$ making the following square commute.

$$\begin{array}{ccc} \Sigma_0\text{-Mon} & \xrightarrow{\Sigma_1^s} & \widehat{\mathbb{C}}_s \\ \downarrow & & \downarrow \\ \widehat{\mathbb{C}}_0 & \xrightarrow{\Sigma_0} & \widehat{\mathbb{C}}_0 \end{array} \quad (7.1)$$

In $\widehat{\mathbb{C}}_s$, we may define functional bisimulations by analogy with Definition 4.15.

Definition 7.6. A morphism $f: X \rightarrow Y$ in $\widehat{\mathbb{C}}_s$ is a *functional bisimulation*, or a *fibration*, iff it enjoys the (weak) right lifting property w.r.t. $\mathbf{y}_{sL}: \mathbf{y}_{s(L)} \rightarrow \mathbf{y}_L$, for all $L \in \mathbb{C}_1$.

Proposition 7.7 (Price for our linguistic mischief). *A dynamic signature preserves functional bisimulations (Definition 6.12) iff the induced functor $\Sigma_0\text{-Mon} \rightarrow \widehat{\mathbb{C}}_s$ does.*

Proof. The functor Σ_1^s maps any functional bisimulation $f: R \rightarrow Y$ to the square (6.1), and just as in Proposition 4.16 a morphism in $\widehat{\mathbb{C}}_1/\Delta_s$ is a functional bisimulation iff the corresponding square is a pointwise weak pullback. \square

7.2. Familiarity. In the previous section, we have shown that functional bisimulation may be defined by lifting both in $\Sigma_0\text{-Mon}$ and $\widehat{\mathbb{C}}_1/\Delta_s$. We now want to exploit this to obtain a characterisation of preservation of functional bisimulations, which will then lead us to useful sufficient conditions.

For this, let us briefly recall familial functors, and show that the functors Σ_0 and $\Sigma_1^s: \Sigma_0\text{-Mon} \rightarrow \widehat{\mathbb{C}}_1/\Delta_s$ induced by the dynamic signature for call-by-name λ -calculus are familial.

Familial functors are a generalisation of polynomial functors on sets, i.e., functors of the form $F(X) = \sum_{o \in O} X^{n_o}$, where O is a set of ‘operations’, and $n_o \in \mathbb{N}$ is the ‘arity’ of any $o \in O$. We want to generalise this to presheaf categories.

Example 7.8. Consider for example the ‘free category’ monad T on \mathbf{Gph} . Analysing and abstracting over the definition of T , we will arrive at the notion of familial functor. Let us first recall that graphs are presheaves over the category $[0] \overset{s}{\underset{t}{\rightrightarrows}} [1]$. T does not change the vertex set, and an edge of $T(G)$ is merely a path in G . Indexing this by the length of the path, we obtain

$$T(G)[0] = \mathbf{Gph}(y_{[0]}, G) \quad \text{and} \quad T(G)[1] = \sum_n \mathbf{Gph}([n], G),$$

where $[n]$ denotes the filiform graph $\bullet \rightarrow \bullet \dots \rightarrow \bullet$ with n edges (which is consistent with $[0]$ and $[1]$ through the Yoneda embedding). Furthermore, the source of a path $[n] \rightarrow G$ in $T(G)$ is obtained as the composite

$$[0] \xrightarrow{s_n} [n] \rightarrow G,$$

where the first morphism selects the first vertex of the path. Similarly the target is obtained by precomposition with the morphism, say t_n , selecting the last vertex.

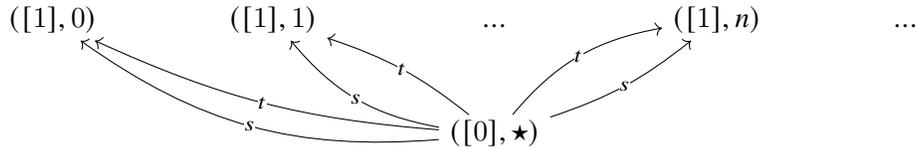
From these observations, we may deduce that the whole of T may be derived from

- the graph $T(1)$, which generalises the set of operations, and
- a functor $\text{el}(T(1)) \rightarrow \mathbf{Gph}$, morally describing the arity of each operation,

where we recall from MacLane and Moerdijk [33]:

Definition 7.9. The *category of elements* $\text{el}(X)$ of a presheaf X over any category \mathbb{C} has pairs (c, x) with $x \in X(c)$ as objects, and a morphism $f \uparrow x': (c, x) \rightarrow (c', x')$ for all $f: c \rightarrow c'$ such that $X(f)(x') = x$.

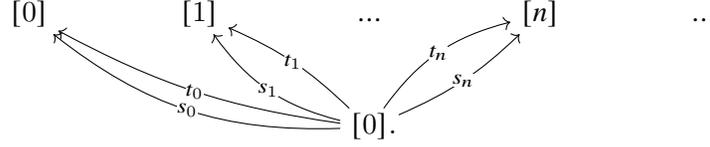
The graph $T(1)$ has a single vertex, and as many paths as we can derive from a single endo-edge on this vertex: \mathbb{N} , because there is one for each length. The category of elements of $T(1)$ thus looks like the following,



and the assignments

$$([0], \star) \mapsto [0] \quad ([1], n) \mapsto [n]$$

extend to a functor $E: \text{el}(T(1)) \rightarrow \mathbf{Gph}$ by mapping each source or target map $([0], \star) \rightarrow ([1], n)$ to the corresponding map $[0] \rightarrow [n]$. This functor may be visualised as



The promised expression of T in terms of $T(1)$ and E is:

$$T(G)(c) \cong \sum_{o \in T(1)(c)} \mathbf{Gph}(E(c, o), G).$$

Definition 7.10. A functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ to some presheaf category is *familial* iff we have a natural isomorphism

$$F(X)(c) \cong \sum_{o \in \mathcal{O}(c)} \mathcal{A}(E(c, o), X),$$

for some presheaf $O \in \widehat{\mathbb{C}}$ and functor $E: \text{el}(O) \rightarrow \mathcal{A}$. The presheaf O is called the *presheaf of operations*, or the *spectrum* [15] of F , while E is called the *exponent*.

Remark 7.11. If \mathcal{A} has a terminal object, we always have $O \cong F(1)$.

Example 7.12. Let us show that the endofunctor $\Sigma_0: \widehat{\mathbb{C}}_0 \rightarrow \widehat{\mathbb{C}}_0$ for λ -calculus is familial. We have

$$\begin{aligned} \Sigma_0(X)(n) &= X(n+1) + X(n)^2 \\ &\cong \widehat{\mathbb{C}}_0(\mathbf{y}_{n+1}, X) + \widehat{\mathbb{C}}_0(2 \cdot \mathbf{y}_n, X) \end{aligned}$$

Thus, we choose:

$$\begin{aligned} O(n) &= \{\text{abs}, \text{app}\} & E(n, \text{abs}) &= \mathbf{y}_{n+1} \\ & & E(n, \text{app}) &= 2 \cdot \mathbf{y}_n. \end{aligned}$$

These definitions can be straightforwardly upgraded to functors $O \in \widehat{\mathbb{C}}_0$ and $E: \text{el}(O) \rightarrow \widehat{\mathbb{C}}_0$, and we get the desired isomorphism.

Example 7.13. Let us now show that the functor $\Sigma_1^s: \Sigma_0\text{-Mon} \rightarrow \widehat{\mathbb{C}}_1/\Delta_s$ for call-by-name λ -calculus is familial. By definition, it maps any X to the set-map $X_0(1) + A_\beta(X) \rightarrow \Sigma_0(X_0)(0)$ defined in Example 3.20. Let us transfer this across the isomorphism $\widehat{\mathbb{C}}_1/\Delta_s \cong \widehat{\mathbb{C}}_s$ (recalling Remark 7.3). On states, we almost may proceed as for Σ_0 , except that the domain category has changed (from $\widehat{\mathbb{C}}_0$ to $\Sigma_0\text{-Mon}$). But recalling that $\mathcal{L}: \mathbf{C} \rightarrow \Sigma_0\text{-Mon}$ denotes the left adjoint to the forgetful functor \mathcal{U} , we have

$$\begin{aligned} \Sigma_1^s(X)(n) &= \Sigma_0((\mathcal{U}(X))_0)(n) \\ &\cong \mathbf{C}(\mathbf{y}_{n+1}, \mathcal{U}(X)) + \mathbf{C}(2 \cdot \mathbf{y}_n, \mathcal{U}(X)) \\ &\cong \Sigma_0\text{-Mon}(\mathcal{L}(\mathbf{y}_{n+1}), X) + \Sigma_0\text{-Mon}(\mathcal{L}(2 \cdot \mathbf{y}_n), X), \end{aligned}$$

so we may (partially) define

$$\begin{aligned} O(n) &= \{\text{abs}, \text{app}\} & E(n, \text{abs}) &= \mathcal{L}(\mathbf{y}_{n+1}) \\ & & E(n, \text{app}) &= \mathcal{L}(2 \cdot \mathbf{y}_n). \end{aligned} \tag{7.2}$$

Now, remembering from Notation 7.4 that we call $\Downarrow \in \mathbb{C}_s$ the unique object of \mathbb{C}_s coming from $\mathbb{C}_1 = 1$, on transitions, we have:

$$\begin{aligned} \Sigma_1^s(X)(\Downarrow) &= X_0(1) + A_\beta(X) \\ &\cong \mathbf{C}(\mathbf{y}_1, \mathcal{U}(X)) + A_\beta(X) \\ &\cong \Sigma_0\text{-Mon}(\mathcal{L}(\mathbf{y}_1), X) + A_\beta(X). \end{aligned}$$

We thus need to find E_β such that $A_\beta(X) \cong \Sigma_0\text{-Mon}(E_\beta, X)$, and then we would complete equations (7.2) with:

$$\begin{aligned} O(\Downarrow) &= \{\lambda\text{-val}, \beta\text{-red}\} & E(\Downarrow, \lambda\text{-val}) &= \mathcal{L}(\mathbf{y}_1) \\ & & E(\Downarrow, \beta\text{-red}) &= E_\beta. \end{aligned}$$

And indeed, let E_β denote the following pushout.

$$\begin{array}{ccccc} \mathcal{L}(\mathbf{y}_0) & \xrightarrow{\bar{\chi}} & \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_1) & \xrightarrow{\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_{\Downarrow})} & \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_{\Downarrow}) \\ \mathcal{L}(\mathbf{y}_{s_{\Downarrow}}) \downarrow & & & & \downarrow \text{in}_2 \\ \mathcal{L}(\mathbf{y}_{\Downarrow}) & \xrightarrow{\text{in}_1} & & & E_\beta \end{array}$$

The morphism $\bar{\chi}$ is defined to be the mate of some morphism $\mathbf{y}_0 \rightarrow \mathcal{U}(\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_1))$, itself corresponding by Yoneda to some element of $\mathcal{U}(\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_1))(0)$. The presheaf $\mathcal{U}(\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_1))$ has as states λ -terms over a closed constant k_0 , and a unary constant k_1 . We pick the element $k_1[k_0]$.

Let us now show that $A_\beta(X) \cong \Sigma_0\text{-Mon}(E_\beta, X)$ for any X : as $\Sigma_0\text{-Mon}(-, X)$ turns colimits into limits, we have the pullback

$$\begin{array}{ccc} [E_\beta, X] & \xrightarrow{\quad} & [\mathcal{L}(\mathbf{y}_{\Downarrow}), X] \\ \downarrow \lrcorner & & \downarrow [\mathcal{L}(\mathbf{y}_{s_{\Downarrow}}), X] \\ [\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_{\Downarrow}), X] & \xrightarrow{\quad} & [\mathcal{L}(\mathbf{y}_0), X] \end{array}$$

$[\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_1), X] \xrightarrow{[\bar{\chi}, X]} [\mathcal{L}(\mathbf{y}_0), X]$

(where we abbreviate $\Sigma_0\text{-Mon}(-1, -2)$ to $[-1, -2]$ for readability). By Yoneda, this reduces to

$$\begin{array}{ccc} [E_\beta, X] & \xrightarrow{\quad} & X(\Downarrow) \\ \downarrow \lrcorner & & \downarrow X(s_{\Downarrow}) \\ X(0) \times X(\Downarrow) & \xrightarrow{X(0) \times X(t_{\Downarrow})} & X(0) \times X(1) \xrightarrow{(t, u) \mapsto u[t]} X(0), \end{array}$$

which shows that we have $A_\beta(X) \cong [E_\beta, X]$ as desired.

We have thus defined the actions of the functors $O \in \widehat{\mathbb{C}}_s$ and $E: \text{el}(O) \rightarrow \Sigma_0\text{-Mon}$ on objects. On morphisms, the only non-obvious point is the image of $s_{\Downarrow} \uparrow \lambda\text{-val}$ and $s_{\Downarrow} \uparrow \beta\text{-red}$. The former morphism is mapped to the identity on $E(0, \mathbf{abs}) = \mathcal{L}(\mathbf{y}_1) = E(\Downarrow, \lambda\text{-val})$. The latter is mapped to the composite

$$\mathcal{L}(2 \cdot \mathbf{y}_0) \xrightarrow{\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_{s_{\Downarrow}})} \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_{\Downarrow}) \xrightarrow{\text{in}_2} E_\beta. \quad (7.3)$$

This achieves the desired isomorphism $\Sigma_1^s(X)(c) \cong \sum_{o \in O(c)} \Sigma_0\text{-Mon}(E(c, o), X)$.

7.3. Cellularity. We now want to exploit familiarity to obtain an alternative characterisation of preservation of functional bisimulations. The starting point is the observation that when a functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ is familial, say as $F(A)(c) = \sum_{o \in O(c)} \mathcal{A}(E(c, o), A)$, then any morphism of the form $f: \mathbf{y}_c \rightarrow F(A)$, corresponding by Yoneda and familiarity to some pair (o, ϕ) with $\phi: E(c, o) \rightarrow A$, factors as

$$\mathbf{y}_c \xrightarrow{(o, \text{id}_{E(c, o)})} F(E(c, o)) \xrightarrow{F(\phi)} F(A).$$

Furthermore, the first component $(o, \text{id}_{E(c, o)})$ is easily seen to be *generic*, in the following sense.

Definition 7.14. Given any functor $F: \mathcal{A} \rightarrow \mathcal{B}$, a morphism $\xi: B \rightarrow F(A)$ is F -generic (or *generic* for short) whenever any square of the form below (solid) admits a unique lifting k (dashed) such that $F(k) \circ \xi = \chi$ and $g \circ k = f$.

$$\begin{array}{ccc} B & \xrightarrow{\chi} & F(C) \\ \xi \downarrow & \nearrow F(k) & \downarrow F(g) \\ F(A) & \xrightarrow{F(f)} & F(D) \end{array}$$

In fact, we have the following important alternative characterisation of familial functors to presheaf categories.

Theorem 7.15. *For any functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ such that \mathcal{A} has a terminal object, F is familial iff all morphisms $f: X \rightarrow F(A)$ factor as*

$$X \xrightarrow{\xi} F(U) \xrightarrow{F(\phi)} F(A),$$

with ξ generic.

Proof. See [45, Theorem 8.1] and [21, §3]. □

Remark 7.16. The factorisation is essentially unique.

Now, recalling Proposition 6.2, let us consider a situation in which both categories \mathcal{A} and $\widehat{\mathbb{C}}$ are equipped with notions of functional bisimulation defined by lifting: we are given sets \mathbb{J} and \mathbb{K} of morphisms in \mathcal{A} and $\widehat{\mathbb{C}}$, respectively. Now for any familial functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ and commuting square

$$\begin{array}{ccc} C & \xrightarrow{u} & F(A) \\ k \downarrow & & \downarrow F(f) \\ D & \xrightarrow{v} & F(B) \end{array} \quad (7.4)$$

with $k \in \mathbb{K}$ and $f \in \mathbb{J}^\natural$, we may take generic factorisations of both horizontal morphisms and use genericness to factor the original square as the solid part of

$$\begin{array}{ccccc} C & \xrightarrow{\xi} & F(X) & \xrightarrow{F(\phi)} & F(A) \\ k \downarrow & & \downarrow F(\delta) & \nearrow F(l) & \downarrow F(f) \\ D & \xrightarrow{\chi} & F(Y) & \xrightarrow{F(\psi)} & F(B). \end{array}$$

Clearly, if $\delta \in {}^\natural(\mathbb{J}^\natural)$, then we find a lifting l as shown, which makes $F(l) \circ \chi$ into a lifting for the original square. We have shown the ‘if’ part of

Definition 7.17. Given categories \mathcal{A} and $\widehat{\mathbb{C}}$ respectively equipped with sets of morphisms \mathbb{J} and \mathbb{K} , such that \mathcal{A} has a terminal object, then a familial functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ is *cellular* iff for all commuting squares

$$\begin{array}{ccc} C & \xrightarrow{k} & D \\ \xi \downarrow & & \downarrow \chi \\ F(X) & \xrightarrow{F(\delta)} & F(Y) \end{array} \quad (7.5)$$

with $k \in \mathbb{K}$ and ξ and χ generic, we have $\delta \in {}^\natural(\mathbb{J}^\natural)$.

Lemma 7.18. *Given categories \mathcal{A} and $\widehat{\mathbb{C}}$ respectively equipped with sets of morphisms \mathbb{J} and \mathbb{K} , such that \mathcal{A} has a terminal object, then a familial functor $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ preserves fibrations iff it is cellular.*

Proof. Conversely, let us assume that F preserves fibrations, and consider any square of the form (7.5). We need to show $\delta \in {}^{\mathfrak{n}}(\mathbb{J}^{\mathfrak{n}})$. But for any commuting square as below left

$$\begin{array}{ccc} X & \xrightarrow{\phi} & A \\ \delta \downarrow & & \downarrow f \\ Y & \xrightarrow{\psi} & B \end{array} \qquad \begin{array}{ccccc} C & \xrightarrow{\xi} & F(X) & \xrightarrow{F(\phi)} & F(A) \\ & & \downarrow F(\delta) & \dashrightarrow & \downarrow F(f) \\ k \downarrow & & \downarrow F(l) & \dashrightarrow & \\ D & \xrightarrow{\chi} & F(Y) & \xrightarrow{F(\psi)} & F(B) \end{array}$$

with $f \in \mathbb{J}^{\mathfrak{n}}$, by pasting this square with our generic square (7.5), we obtain the solid part above right. But because F preserves fibrations, we find a lifting γ as shown, which by genericness of χ (and then ξ) yields the desired lifting l . \square

This yields the announced characterisation of preservation of functional bisimulations:

Corollary 7.19. *In any Howe context, for any operational semantics signature (Σ_0, Σ_1) , if Σ_1^s is familial with exponent $E: \text{el}(\Sigma_1^s(1)) \rightarrow \Sigma_0\text{-Mon}$, then it preserves functional bisimulations iff for all $L \in \mathbb{C}_1$ and $o \in \Sigma_1^s(1)(L)$, $E(s_L \uparrow o): E(\mathbf{s}(L), o \cdot s_L) \rightarrow E(L, o)$ is a cofibration.*

Proof. We apply Lemma 7.18 with

- \mathbb{J} the set of morphisms $\mathcal{L}(\mathbf{y}_{s_L})$ in $\Sigma_0\text{-Mon}$, for $L \in \mathbb{C}_1$, and
- \mathbb{K} the set of morphisms \mathbf{y}_{s_L} in $\widehat{\mathbb{C}}_s$, for $L \in \mathbb{C}_1$.

The only difficulty lies in realising that in a Howe context the statement unfolds to the same as that of Lemma 7.18. Indeed, for any familial $F: \mathcal{A} \rightarrow \widehat{\mathbb{C}}$ (with \mathcal{A} having a terminal object) and generic morphism $\mathbf{y}_c \rightarrow F(A)$ from a representable, we have $A \cong E(c, o)$, where $o \in F(1)(c)$ denotes the operation corresponding to $\mathbf{y}_c \rightarrow F(A) \rightarrow F(1)$ by Yoneda. Furthermore, in a commuting square

$$\begin{array}{ccc} \mathbf{y}_{s(L)} & \xrightarrow{\mathbf{y}_{s_L}} & \mathbf{y}_L \\ \xi \downarrow & & \downarrow \chi \\ F(X) & \xrightarrow{F(\delta)} & F(Y) \end{array}$$

with ξ and χ generic, the operation o associated to the composite $\mathbf{y}_{s(L)} \xrightarrow{\xi} F(X) \rightarrow F(1)$ is $o' \cdot \mathbf{y}_{s_L}$, where o' denotes the one associated to $\mathbf{y}_L \xrightarrow{\chi} F(Y) \rightarrow F(1)$. Finally, in such a situation, δ is $E(\mathbf{y}_{s_L} \uparrow o')$, or, omitting the Yoneda embedding, $E(s_L \uparrow o')$. \square

This characterisation of preservation of functional bisimulations in terms of cofibrations is easier to prove in practice, since cofibrations in turn admit the following well-known characterisation.

Definition 7.20. Consider any set \mathbb{J} of maps in a given category.

- A *basic relative \mathbb{J} -cell complex* is any morphism f obtained by pushing out some morphism from \mathbb{J} along any morphism, as in

$$\begin{array}{ccc}
A & \xrightarrow{j \in \mathbb{J}} & B \\
\downarrow & \lrcorner & \downarrow \\
C & \xrightarrow{f} & D.
\end{array}$$

- A *relative \mathbb{J} -cell complex* is a (potentially transfinite) composite of basic relative \mathbb{J} -cell complexes.

Proposition 7.21 ([27, Lemma 2.1.10]). *For any set \mathbb{J} of maps in a locally presentable category, all relative \mathbb{J} -cell complexes are cofibrations in the generated weak factorisation system.*

8. APPLICATIONS

In this section, we apply our results to show that substitution-closed bisimilarity is a congruence in concrete examples.

8.1. Call-by-name. We have already specified the syntax (§3.1) and transitions (§3.3) of call-by-name λ -calculus. By Theorem 6.15, Corollary 7.19, and Proposition 7.21, congruence of substitution-closed bisimilarity will follow if we prove that the morphisms $E(s_{\Downarrow} \uparrow \sigma)$ corresponding to both transition rules are relative cell complexes.

The morphism $E(s_{\Downarrow} \uparrow \lambda\text{-val})$ is an identity, hence trivially a relative cell complex.

For the β -rule, it is not entirely trivial that the map $E(s_{\Downarrow} \uparrow \beta\text{-red})$ corresponding to the second transition rule, defined as the composite (7.3), is a cofibration. But, as (essentially) noted in [12, Example 5.21], it is a relative cell complex by construction as both components are pushouts of $\mathcal{L}(\mathbf{y}_{s_{\Downarrow}})$, as should be clear from the following diagram.

$$\begin{array}{ccccc}
& & \mathcal{L}(\mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow}) \\
& & \downarrow & & \lrcorner \downarrow \\
\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow} + \mathbf{y}_0) & \longrightarrow & E_{\beta}
\end{array}$$

8.2. Call-by-value. Let us now treat the call-by-value variant of untyped λ -calculus, essentially as in [36, 37]. In this setting, it is important to distinguish substitution by values and by terms. Indeed, letting I denote the identity $\lambda x.x$, the terms $e = \lambda x.I$ and $e' = \lambda x.((\lambda y.I) x)$ are contextually equivalent, since during evaluation in any context, the bound variable will only be replaced with a value. However, if one defines applicative bisimulation naively, i.e., requiring it to be closed under arbitrary substitution, then e and e' are not bisimilar, as I is not bisimilar to $(\lambda y.I) \Omega$ – which diverges. We thus want to restrict to value substitution – which our treatment in [12] overlooks!

Here is one of several ways of doing this. The idea is to have two sorts, one for values and the other for general terms. We thus would like $\widehat{\mathbb{C}}_0 = \mathbf{C}_0 \simeq [\mathbf{Set}^2, \mathbf{Set}^2]_f$ to be equivalent to finitary endofunctors on \mathbf{Set}^2 . But

$$[\mathbf{Set}^2, \mathbf{Set}^2]_f \simeq [2 \cdot \mathbf{Set}^2_f, \mathbf{Set}] \simeq [2 \cdot \mathbb{F}^2, \mathbf{Set}],$$

so we take \mathbb{C}_0 to be the opposite of $2 \cdot \mathbb{F}^2$. By the equivalence, composition of finitary endofunctors equips \mathbf{C}_0 with monoidal structure, and we denote the two sorts by \mathbf{p} and \mathbf{v} , respectively for “programs” and “values”. In the presheaf point of view, we denote by

$(m, n)_{\mathbf{p}}$ and $(m, n)_{\mathbf{v}}$, respectively, objects in the first and second term of $2 \cdot \mathbb{F}^2 = \mathbb{F}^2 + \mathbb{F}^2$, so that $X(m, n)(c)$ in the finitary endofunctor world corresponds to $X(m, n)_c$ in the presheaf world, for $c \in \{\mathbf{p}, \mathbf{v}\}$.

Since abstraction should bind a value variable, the syntax is thus specified by

$$\Sigma_0(X)(m, n)(\mathbf{p}) = X(m, n)(\mathbf{p})^2 + X(m, n)(\mathbf{v}) \quad \text{and} \quad \Sigma_0(X)(m, n)(\mathbf{v}) = X(m, n+1)(\mathbf{p}).$$

A model X of the syntax should thus in particular feature operations

$$\begin{aligned} \mathbf{abs}_{m,n} : X(m, n+1)(\mathbf{p}) &\rightarrow X(m, n)(\mathbf{v}) & \mathbf{app}_{m,n} : X(m, n)(\mathbf{p})^2 &\rightarrow X(m, n)(\mathbf{p}) \\ \mathbf{val}_{m,n} : X(m, n)(\mathbf{v}) &\rightarrow X(m, n)(\mathbf{p}), \end{aligned}$$

where the last operation requires that values should map to programs.

It is important to understand that the notation $X(m, n)$ may be extended to $X(K)$, for any $K \in \mathbf{Set}^2$, (m, n) being only our notation for finitely presentable objects of the skeleton \mathbf{F}^2 . Thus, e.g., (m, n) is isomorphic to $(m, n) + \mathbf{y}_{\mathbf{v}}$, where $\mathbf{y}_{\mathbf{v}}$ denotes the Yoneda embedding of \mathbf{v} along $2 \rightarrow \mathbb{F}^2 \hookrightarrow \mathbf{Set}^2$. So the arity for abstraction in fact yields an operation $X(K + \mathbf{y}_{\mathbf{v}})(\mathbf{p})$.

Denoting composition of finitary endofunctors by \otimes , we define a pointed strength for Σ_0 as follows, for any $X \in \mathbf{C}_0$, $Y \in I/\mathbf{C}_0$, and $K \in \mathbb{F}^2$:

- at \mathbf{p} , we have

$$(\Sigma_0(X) \otimes Y)(K)(\mathbf{p}) = X(Y(K))(\mathbf{p})^2 + X(Y(K))(\mathbf{v}) = \Sigma_0(X \otimes Y)(K)(\mathbf{p}),$$

so we take $st_{X,Y,K,\mathbf{p}}$ to be the identity;

- at \mathbf{v} , we have

$$(\Sigma_0(X) \otimes Y)(K)(\mathbf{v}) = X(Y(K) + \mathbf{y}_{\mathbf{v}})(\mathbf{p}) \quad \text{and} \quad \Sigma_0(X \otimes Y)(K)(\mathbf{v}) = X(Y(K + \mathbf{y}_{\mathbf{v}}))(\mathbf{p}),$$

so we define $st_{X,Y,K,\mathbf{p}}$ by applying $X(-)(\mathbf{p})$ to the copairing of $Y(K) \rightarrow Y(K + \mathbf{y}_{\mathbf{v}})$ and

$$\mathbf{y}_{\mathbf{v}} \rightarrow I(K + \mathbf{y}_{\mathbf{v}}) \rightarrow Y(K + \mathbf{y}_{\mathbf{v}}).$$

We then specify transitions, which we first recall:

$$\frac{}{\lambda x.e \Downarrow e} \qquad \frac{e_1 \Downarrow e'_1 \quad e_2 \Downarrow e'_2 \quad e'_1[\lambda(e'_2)] \Downarrow e_3}{e_1 \quad e_2 \Downarrow e_3}.$$

In order to specify such transitions, we take $\mathbb{C}_1 = 1$, as in the call-by-name case, with \mathbf{s} and \mathbf{t} mapping the unique object to $(0, 0)_{\mathbf{p}}$ and $(0, 1)_{\mathbf{p}}$, respectively: transitions will relate a closed program to a program with one value variable (morally the body of the obtained abstraction). For any $X \in \mathbf{C} = \widehat{\mathbb{C}_1}/\Delta \cong \mathbf{Set}/\Delta$, We take $\Sigma_1^F(X)$ to be the coproduct

$$\Sigma_1^F(X) = X_0(0, 1) + A_{\beta_{\mathbf{v}}}(X),$$

where $A_{\beta_{\mathbf{v}}}(X)$ denotes the set of valid premises for the second rule in §2.3, i.e., triples $(r_1, r_2, r_3) \in X_1$ such that

$$r_3 \cdot s_{\Downarrow} = (r_1 \cdot t_{\Downarrow})[\lambda(r_2 \cdot t_{\Downarrow})].$$

By Theorem 6.15, Corollary 7.19, and Proposition 7.21, congruence of substitution-closed bisimilarity will follow if we can prove that the morphisms $E(s_L \uparrow o)$ corresponding to both transition rules are relative cell complexes. This is again trivial for the first rule, while for the second one we obtain the following morphism, which is a relative cell complex (almost) by construction.

$$\begin{array}{ccc} \mathcal{L}(\mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow}) \\ \xi \downarrow & & \lrcorner \downarrow \\ \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow} + \mathbf{y}_{\Downarrow}) \longrightarrow E_{\beta_{\mathbf{v}}} \end{array}$$

Here, ξ corresponds by adjunction and Yoneda to the closed term

$$(r_1 \cdot t_{\Downarrow})[\lambda(r_2 \cdot t_{\Downarrow})] \in \mathcal{U}(\mathcal{L}(\mathbf{y}_{\Downarrow} + \mathbf{y}_{\Downarrow}))(0),$$

where r_1 and r_2 denote the two transition constants generating $\mathbf{y}_{\Downarrow} + \mathbf{y}_{\Downarrow}$.

8.3. Erratic non-determinism. The non-deterministic λ -calculus investigated in [39, §7], may be presented by adding a unary operation \uplus to the syntax of pure λ -calculus, using two labels for transitions, i.e., having two transition relations \Downarrow_{λ} (between a closed term and a term with one free variable, as before) and \Downarrow_{τ} (between closed terms), and using the following rules, where Ω denotes any diverging term.

$$\begin{array}{c} \frac{}{\uplus e \Downarrow_{\tau} e} \quad \frac{}{\uplus e \Downarrow_{\tau} \Omega} \quad \frac{}{\lambda x.e \Downarrow_{\lambda} e} \quad \frac{e_1 \Downarrow_{\lambda} e'_1 \quad e'_1[e_2] \Downarrow_{\lambda} e_3}{e_1 e_2 \Downarrow_{\lambda} e_3} \\ \\ \frac{e_1 \Downarrow_{\lambda} e_3 \quad e_3[e_2] \Downarrow_{\tau} e_4}{e_1 e_2 \Downarrow_{\tau} e_4} \quad \frac{e_1 \Downarrow_{\tau} e_3}{e_1 e_2 \Downarrow_{\tau} e_3 e_2} \quad \frac{}{e \Downarrow_{\tau} e} \quad \frac{e_1 \Downarrow_{\tau} e_2 \quad e_2 \Downarrow_{\tau} e_3}{e_1 \Downarrow_{\tau} e_3} . \end{array}$$

Clearly, \Downarrow_{τ} includes β -reduction; the last two rules ensure that it is reflexive and transitive; and the previous one that it applies to the function part of an application. It thus coincides with Sangiorgi's relation \Rightarrow (albeit with redundant rules). The syntax is easily modelled by taking $\Sigma_0(X)(n) = X(n+1) + X(n)^2 + X(n)$, and the rules are easily seen to fit into a Σ_1 such that Σ_1^s preserves functional bisimulations by Corollary 7.19 and Proposition 7.21. Finally, substitution-closed bisimilarity in the initial model coincides with Sangiorgi's applicative bisimilarity, so we again deduce congruence of applicative bisimilarity by Theorem 6.15.

9. CONGRUENCE OF SUBSTITUTION-CLOSED BISIMILARITY

In this section, we elaborate on the proof sketch of Theorem 6.15 given in §6. The overall structure remains the same, and the final part of the proof sketch is complete, so we mainly elaborate on items (1)–(4).

9.1. Preliminaries on spans. In this section, we fix a bicomplete category \mathcal{C} , and develop some tools about spans, including categorified notions of reflexivity, transitivity, symmetry, and transitive closure. We switch freely from spans $X \leftarrow S \rightarrow Y$ to their pairings $S \rightarrow X \times Y$ in $\mathcal{C}/X \times Y$. Furthermore, we denote sequential composition of any $R \rightarrow X \times Y$ and $S \rightarrow Y \times Z$ (constructed by pullback) by $R; S \rightarrow X \times Z$.

Definition 9.1. A span $X \leftarrow S \rightarrow X$ (a.k.a. a graph $S \rightrightarrows X$) is *reflexive* if there is a morphism from the diagonal to S in \mathcal{C}/X^2 . It is *transitive* if there is a morphism $S; S \rightarrow S$ over X^2 . Finally, it is *symmetric* if there is a morphism $S^{\dagger} \rightarrow S$, where $(-)^{\dagger}$ denotes the functor swapping projections.

For potentially non-reflexive spans, we will use the following reflexive transitive closure.

Definition 9.2. The *reflexive transitive closure* S^* of any span $X \leftarrow S \rightarrow X$ is the coproduct $\sum_{n \in \mathbb{N}} S^n$, or for short $\sum_{n \in \mathbb{N}} S^n$ when the context is clear, where S^n denotes iterated span composition of S with itself, with $S^0 = X$.

At some point, we will also use the following relational notion of transitive closure.

Notation 9.3. Given a span S on X , we denote by \overline{S} the induced relation on X by the image factorisation of $S \rightarrow X \times X$.

Definition 9.4. The relational transitive closure $S^{\overline{\cdot}}$ of a span S on X is the union $\bigcup_{n>0} \overline{S}^n$.

We note that the relational transitive closure commutes with symmetry.

Lemma 9.5. For all spans $S \rightarrow X^2$, we have $S^{\overline{\cdot}} \cong S^{\overline{\cdot}\dagger}$.

The following lemma will later be used to exploit preservation of sifted colimits by Σ_0 .

Lemma 9.6. If S is a reflexive span on X , then $S^{\overline{\cdot}}$ is the (filtered) colimit of the chain

$$X \rightarrow \overline{S} \cong \overline{S}; X \rightarrow \overline{S}; S \cong \overline{S}; S; X \rightarrow \overline{S}; S; S \rightarrow \dots$$

The next result will be useful to show that the relational transitive closure of the Howe closure of substitution-closed bisimilarity is symmetric on states.

Lemma 9.7. For any reflexive span $R \rightarrow X^2$, $R^{\overline{\cdot}}$ is symmetric if there exists a span morphism $R \rightarrow R^{\overline{\cdot}\dagger}$.

Proof. Given a morphism $j: R \rightarrow R^{\overline{\cdot}\dagger}$, we consider the composite

$$R^{\overline{\cdot}} \cong \bigcup_{n>0} \overline{R}^n \rightarrow \bigcup_{n>0} \overline{(R^{\overline{\cdot}\dagger})^n} \cong \bigcup_{n>0} \overline{((R^{\overline{\cdot}})^n)^\dagger} \rightarrow \bigcup_{n>0} \overline{R^{\overline{\cdot}\dagger}} \cong \bigcup_{n>0} R^{\overline{\cdot}\dagger} \cong R^{\overline{\cdot}\dagger},$$

where the first morphism is obtained from j , and the second one is obtained from morphisms $(R^{\overline{\cdot}})^n \rightarrow R^{\overline{\cdot}}$. \square

9.2. Howe closure on states. We fix an operational semantics signature (Σ_0, Σ_1) on a Howe context $\mathbf{s}, \mathbf{t}: \mathbb{C}_1 \rightarrow \mathbb{C}_0$.

Definition 9.8. Let $\Sigma_0^H: \mathbf{C}_0/\mathbf{Z}_0^2 \rightarrow \mathbf{C}_0/\mathbf{Z}_0^2$ map any span $X \rightarrow \mathbf{Z}_0^2$ to

$$\Sigma_0(X) + (X; \sim_0^\otimes) \rightarrow \Sigma_0(\mathbf{Z}_0)^2 + \mathbf{Z}_0^2 \rightarrow \mathbf{Z}_0^2.$$

This functor Σ_0^H is clearly inspired from the standard Howe closure. We now want to prove that it is pointed strong, which requires us to equip $\mathbf{C}_0/\mathbf{Z}_0^2$ with monoidal structure. But \mathbf{Z}_0^2 is a monoid, and it is well-known [45, §2] that any slice of a monoidal category over any monoid M is again monoidal. The tensor of $X \rightarrow M$ and $Y \rightarrow M$ is simply $X \otimes Y \rightarrow M \otimes M \rightarrow M$, and the unit is $I \rightarrow M$. We may thus state the following result.

Proposition 9.9. The functor Σ_0^H is pointed strong.

For proving this, we first need the following.

Lemma 9.10. For any monoid X in any monoidal category \mathcal{C} with finite limits, there is a natural transformation with components $\delta_{U,V,W}: (U; V) \otimes W \rightarrow U \otimes W; V \otimes X$ in \mathcal{C}/X^2 .

Proof. Let $m: X \otimes X \rightarrow X$ denote the multiplication. By tensoring the defining pullback of $U; V$ with W we obtain the back face below.

$$\begin{array}{ccccc}
(U; V) \otimes W & \xrightarrow{\quad} & V \otimes W & \xrightarrow{V \otimes \pi_2} & V \otimes X \\
\downarrow & \dashrightarrow^{\delta_{U,V,W}} & \downarrow & & \downarrow \\
(U \otimes W); (V \otimes X) & \xrightarrow{\quad} & (U \otimes W); (V \otimes X) & \xrightarrow{\quad} & (U \otimes W); (V \otimes X) \\
\downarrow & & \downarrow \pi_1 \otimes W & & \downarrow m \circ (\pi_1 \otimes X) \\
U \otimes W & \xrightarrow{\quad} & X \otimes W & \xrightarrow{m \circ (X \otimes \pi_2)} & X \\
\downarrow & & \downarrow \pi_2 \otimes W & & \downarrow \\
U \otimes W & \xrightarrow{\quad} & U \otimes W & \xrightarrow{m \circ (\pi_2 \otimes \pi_2)} & X
\end{array}$$

By universal property of pullback, we then get the dashed arrow making all faces commute, which gives our candidate $\delta_{U,V,W}$. Naturality follows by universal property of pullback. \square

Proof of Proposition 9.9. Pointed strong endofunctors are closed under coproducts, so it suffices to show that both terms of the sum are pointed strong. The first one inherits the pointed strength of Σ_0 , while the pointed strength of $-; \sim_0^\otimes$ follows from Lemma 9.10 and substitution-closedness of $\sim_0^\otimes: (U; \sim_0^\otimes) \otimes V \rightarrow (U \otimes V); (\sim_0^\otimes \otimes \mathbf{Z}_0) \rightarrow (U \otimes V); \sim_0^\otimes$. \square

Presheaf categories being well-known to be closed under the slice construction, we have the following.

Lemma 9.11. *The category $\mathbf{C}_0/\mathbf{Z}_0^2$ is a presheaf category.*

This allows us to deduce the following.

Proposition 9.12. *The endofunctor Σ_0^H is finitary.*

Proof. By commutation of filtered colimits with finite limits in presheaf categories. \square

By Proposition 9.9, 9.9, and 9.12, the following is legitimate.

Definition 9.13. Let $H_0 = \mathbf{Z}_{\Sigma_0^H}$ denote the initial Σ_0^H -monoid.

By Proposition 5.1, we also get the following for free.

Proposition 9.14. *The object $H_0 \rightarrow \mathbf{Z}_0^2$ is an initial algebra for the endofunctor $\mathbf{C}_0/\mathbf{Z}_0^2 \rightarrow \mathbf{C}_0/\mathbf{Z}_0^2$ mapping any $X \rightarrow \mathbf{Z}_0^2$ to $I + \Sigma_0^H(X) \rightarrow \mathbf{Z}_0^2$.*

Proposition 9.15. *The underlying object H_0 is a Σ_0 -monoid.*

Proof. Directly follows from the Σ_0^H -monoid structure. \square

Next, we exhibit an alternative characterisation of H_0 , which relies on the following result.

Lemma 9.16 (Packing lemma). *Consider finitary endofunctors F and G on a cocomplete category \mathbf{C} , and let $G^\star(A) \cong \mu S.(A + G(S))$ denote the ‘free G -algebra’ monad [38, Theorem 2.1]. Then we have $\mu S.(F(S) + G(S)) \cong \mu S.G^\star(F(S))$.*

Proof. Indeed, we have

$$\begin{aligned}
\mu S.G^\star(F(S)) &\cong \mu S.\mu U.(F(S) + G(U)) \\
&\cong \mu S.(F(S) + G(S)) \quad (\text{by the Diagonal rule [7, Theorem 16]}). \quad \square
\end{aligned}$$

Proposition 9.17. *The object $H_0 \rightarrow \mathbf{Z}_0^2$ is an initial algebra for the endofunctor $\Sigma_0^{H'}: \mathbf{C}_0/\mathbf{Z}_0^2 \rightarrow \mathbf{C}_0/\mathbf{Z}_0^2$ mapping any $X \rightarrow \mathbf{Z}_0^2$ to $I; \sim_0^{\otimes*} + \Sigma_0(X); \sim_0^{\otimes*} \rightarrow \mathbf{Z}_0^2$.*

Proof. Taking $F(S) = I + \Sigma_0(S)$ and $G(S) = S; \sim_0^{\otimes}$ in Lemma 9.16, and observing that G preserves coproducts (because pullback along the first projection $\sim_0^{\otimes} \rightarrow \mathbf{Z}_0$, as a left adjoint, preserves colimits), so that $G^\star(U) \cong \sum_n G^n(U)$, we have (by commutation of coproducts with $U; -$)

$$G^\star(U) \cong \sum_n G^n(U) \cong \sum_n U; (\sim_0^{\otimes})^n \cong U; \sum_n (\sim_0^{\otimes})^n = U; \sim_0^{\otimes*}.$$

Thus, $G^\star(F(S)) \cong (I + \Sigma_0(S)); \sim_0^{\otimes*} \cong (I; \sim_0^{\otimes*}) + (\Sigma_0(S); \sim_0^{\otimes*})$, as desired. \square

9.3. Double categorical notation. Our next goal is to define the Howe closure on transitions. For this, we appeal to Morton’s *double bicategories* [35]. They are a refinement of double categories, in which both the horizontal and vertical categories are bicategories. We rely in particular on his Theorem 4.1.3, which (when dualised) states that for any category \mathcal{C} with pullbacks, there is a double bicategory $2Sp(\mathcal{C})$:

- objects are objects of \mathcal{C} ,
- both the vertical and horizontal bicategories are $Span(\mathcal{C})$,
- cells, called *double spans*, are precisely commuting diagrams of the following form.

$$\begin{array}{ccccc} A & \longleftarrow & B & \longrightarrow & C \\ \uparrow & & \uparrow & & \uparrow \\ A' & \longleftarrow & B' & \longrightarrow & C' \\ \downarrow & & \downarrow & & \downarrow \\ A'' & \longleftarrow & B'' & \longrightarrow & C'' \end{array} \tag{9.1}$$

We will not need the rest of the structure. All we need to know is that cells compose horizontally and vertically just as in a double category. We will use the double bicategories $2Sp(\mathbf{C}_0)$ and $2Sp(\widehat{\mathbf{C}}_1)$.

Notation 9.18. *We use the following notational conventions.*

- We denote cells $2Sp(\mathbf{C}_0)$ such as (9.1) above by

$$\begin{array}{ccc} A & \xrightarrow{B} & C \\ A' \downarrow & & \downarrow C' \\ A'' & \xrightarrow{B''} & C'' \end{array}$$

- Furthermore, cells in $\widehat{\mathbf{C}}_1$ of the form below left will be denoted as below right.

$$\begin{array}{ccc} X_0\mathbf{s} & \longleftarrow S_0\mathbf{s} & \longrightarrow Y_0\mathbf{s} \\ \uparrow & \vdots & \uparrow \\ X_1 & \longleftarrow S_1 & \longrightarrow Y_1 \\ \downarrow & \vdots & \downarrow \\ X'_0\mathbf{t} & \longleftarrow S'_0\mathbf{t} & \longrightarrow Y'_0\mathbf{t} \end{array} \qquad \begin{array}{ccc} X_0 & \xrightarrow{S_0} & Y_0 \\ x_1 \downarrow & S_1 & \downarrow y_1 \\ X'_0 & \xrightarrow{S'_0} & Y'_0 \end{array} \tag{9.2}$$

Explicitly, spans of the form $X_0 \mathbf{s} \leftarrow X_1 \rightarrow X'_0 \mathbf{t}$ are denoted by $X_0 \xrightarrow{X_1} X'_0$, while spans of the form $X_0 \leftarrow S_0 \rightarrow Y_0$ are still denoted by $X_0 \xrightarrow{S_0} Y_0$, but silently coerced by $\Delta_{\mathbf{s}}$ or $\Delta_{\mathbf{t}}$ depending on context.

- For both types of cells, we collapse identity borders, as usual.
- When a span is trivial on one side, we use standard arrows for its borders, and a double arrow for its middle arrow, all in the relevant direction. E.g., the diagram below left may be depicted as below right.

$$\begin{array}{ccc} A \leftarrow B \longrightarrow C & & A \xrightarrow{B} C \\ a \uparrow & \uparrow b & \uparrow c \\ A' \leftarrow B' \longrightarrow C' & & A' \xrightarrow{B'} C' \end{array}$$

Cells of the form (9.2) live in $2Sp(\widehat{\mathbb{C}}_1)$, hence may be composed horizontally. Relevant examples of vertical composition will be obtained by embedding cells of the form (9.1) along $\Delta_{\mathbf{s}}$ (resp. $\Delta_{\mathbf{t}}$), and vertically composing with cells of the form (9.2) in $2Sp(\widehat{\mathbb{C}}_1)$. This yields a top (resp. bottom) action of $2Sp(\mathbb{C}_0)$, which we both denote by mere pasting.

Lemma 9.19. *Given a composable pasting diagram made of cells of both types, any two parsings agree up to isomorphism.*

Proof. By interchange of limits. □

9.4. Howe closure on transitions. Let us now define the Howe closure on transitions. First, we delineate an ambient category $\mathbf{C}_{\mathbf{Z}}^H$. The idea is that objects of this category should be transition systems $S \rightarrow \mathbf{Z}^2$ over \mathbf{Z}^2 whose image under the projection $\mathbf{C}/\mathbf{Z}^2 \rightarrow \mathbf{C}_0/\mathbf{Z}_0^2$ is precisely $H_0 \rightarrow \mathbf{Z}_0^2$. Thus, an object of $\mathbf{C}_{\mathbf{Z}}^H$ consists of an object $S_1 \in \mathbf{C}_1$, equipped with a dashed cone to the outer part of the diagram below.

$$\begin{array}{ccccc} \mathbf{Z}_0 \mathbf{s} & \leftarrow & H_0 \mathbf{s} & \longrightarrow & \mathbf{Z}_0 \mathbf{s} \\ \uparrow & & \uparrow \cdots & & \uparrow \\ \mathbf{Z}_1 & \leftarrow \cdots & S_1 & \cdots \longrightarrow & \mathbf{Z}_1 \\ \downarrow & & \downarrow \cdots & & \downarrow \\ \mathbf{Z}_0 \mathbf{t} & \leftarrow & H_0 \mathbf{t} & \longrightarrow & \mathbf{Z}_0 \mathbf{t} \end{array} \tag{9.3}$$

Equivalently, they are morphisms over the limit, so that we may define $\mathbf{C}_{\mathbf{Z}}^H$ as a slice category by merely stating the following.

Definition 9.20. Let $R^{\partial H}$ denote the limit of the outer part of (9.3).

Definition 9.21. $\mathbf{C}_{\mathbf{Z}}^H$ is the category of cones over the outer part of (9.3), or equivalently, it is the slice category $\mathbf{C}_1/R^{\partial H}$.

Proposition 9.22. *The initial object in $\mathbf{C}_{\mathbf{Z}}^H$ is the span $\mathbf{Z} \leftarrow H_0 \rightarrow \mathbf{Z}$, i.e., the one with $S_1 = 0$.*

Definition 9.23. Let $\Sigma_1^H : \mathbf{C}_Z^H \rightarrow \mathbf{C}_Z^H$ map any object $\mathbf{Z} \leftarrow S \rightarrow \mathbf{Z}$ to the coproduct of the following two pastings.

Proposition 9.24. The functor $\Sigma_1^H : \mathbf{C}_Z^H \rightarrow \mathbf{C}_Z^H$ is finitary.

Proof. The forgetful functor $\mathbf{C}_Z^H \cong \mathbf{C}_1/R^{\partial H} \rightarrow \mathbf{C}_1$ creates colimits, so it suffices to show that the composite $\mathbf{C}_Z^H \xrightarrow{\Sigma_1^H} \mathbf{C}_Z^H \rightarrow \mathbf{C}_1$ is finitary. This functor maps any S to $\Sigma_1^F(S) + S_1; \sim_1^{\otimes}$, hence is finitary because Σ_1^F is and $-; \sim_1^{\otimes}$ is cocontinuous. \square

The last result legitimates the following definition.

Definition 9.25. Let H denote the initial (vertical) Σ_1^H -algebra.

We readily can prove the following.

Lemma 6.16. There exists a span morphism $\sim^{\otimes} \rightarrow H$.

Proof. By construction, the underlying object of H is in particular a $\check{\Sigma}_1$ -algebra, so by initiality we obtain a unique span morphism $\mathbf{Z} \rightarrow H$ – in other words H is reflexive. Furthermore, again by construction, H is an algebra for the endofunctor $-; \sim^{\otimes}$ on \mathbf{C}/\mathbf{Z}^2 . We thus may form the composite $\sim^{\otimes} \cong \mathbf{Z}; \sim^{\otimes} \rightarrow H; \sim^{\otimes} \rightarrow H$. \square

9.5. Alternative characterisations of the Howe closure. In this section, we exhibit a few alternative characterisations of the Howe closure on transitions. The definition in the previous section is convenient for proving that the transitive closure is symmetric, while our final alternative characterisation will enable a conceptual proof of the simulation property.

First of all, by the packing Lemma 9.16, we have:

Lemma 9.27. The Howe closure H is (isomorphic to) the initial algebra of the endofunctor $\Sigma_{1,pack}^H : \mathbf{C}_Z^H \rightarrow \mathbf{C}_Z^H$ mapping any $\mathbf{Z} \leftarrow S \rightarrow \mathbf{Z}$ to the following pasting.

For the proof, we need an intermediate result, Corollary 9.29 below, which relies on the following lemma.

Lemma 9.28. *Consider the following diagram of functors and natural transformations*

$$\begin{array}{ccc} \mathcal{A} & \xrightarrow{F} & \mathcal{A} \\ K \uparrow & \Downarrow \alpha & \uparrow K \\ \mathcal{B} & \xrightarrow{G} & \mathcal{B} \end{array}$$

such that F and G are finitary endofunctors on cocomplete categories \mathcal{A} and \mathcal{B} . Let M be the endofunctor on the comma category \mathcal{A}/K mapping $a \xrightarrow{f} Kb$ to $Fa \xrightarrow{Ff} FKb \xrightarrow{\alpha_b} KGb$. Then, given an object $f : a \rightarrow Jb$ of \mathcal{A}/K , there is a unique morphism $f^* : F^*a \rightarrow KG^*b$ such that

- $(FF^*a \rightarrow F^*a, GG^*b \rightarrow G^*b)$ is a morphism from Mf^* to f^* ;
- $(a \rightarrow F^*a, b \rightarrow G^*b)$ is a morphism from f to f^* .

Furthermore, $M^*f = f^*$.

Proof. The first thing to note is that K lifts to a functor $\overline{K} : G\text{-alg} \rightarrow F\text{-alg}$, mapping $Gx \rightarrow x$ to $FKx \xrightarrow{\alpha_x} KGx \rightarrow Kx$, so that $M\text{-alg}$ is isomorphic (in the obvious way) to the comma category $F\text{-alg}/\overline{K}$.

Through this isomorphism, we want to show that $(F^*a \rightarrow \overline{K}G^*b)$ is the free M -algebra on $a \rightarrow b$. Here, existence and the claimed uniqueness property of the morphism $F^*a \rightarrow \overline{K}G^*b$ follows from the universal property of F^*a as a free algebra on a .

We now check the universal property of $(F^*a \rightarrow \overline{K}G^*b)$. A M -algebra morphism from it to a M -algebra $a' \rightarrow \overline{K}b'$ consists of

- a F -algebra morphism $F^*a \rightarrow a'$;
- a G -algebra morphism $G^*b \rightarrow b'$;
- such that the following diagram commutes.

$$\begin{array}{ccc} F^*a & \longrightarrow & a' \\ \downarrow & & \downarrow \\ \overline{K}G^*b & \longrightarrow & \overline{K}b' \end{array}$$

By universal properties of F^*a and G^*b as free algebras, the first two items amount to morphisms $a \rightarrow a'$ and $b \rightarrow b'$ in \mathcal{A} and \mathcal{B} , respectively. As for the required commuting diagram, note that it can be lifted in the category of F -algebras. Then, it can be shown to be equivalent to the commutation of the following diagram, exploiting again the universal property of F^*a .

$$\begin{array}{ccc} a & \longrightarrow & a' \\ \downarrow & & \downarrow \\ Kb & \longrightarrow & Kb' \end{array}$$

Finally, M -algebra morphisms from $F^*a \rightarrow \overline{K}G^*b$ to $a' \rightarrow \overline{K}b'$ are in one to one correspondence to morphisms from $a \rightarrow Kb$ to $a' \rightarrow Kb'$. It is straightforward to check that the first direction correspond to precomposition with $(a \rightarrow a', b \rightarrow b')$, as expected. \square

Corollary 9.29. *Consider the following diagram of functors and natural transformations*

$$\begin{array}{ccccc}
 \mathcal{A} & \xrightarrow{U} & \mathcal{A} & & \\
 J \downarrow & & \Downarrow \alpha & & \downarrow J \\
 \mathcal{B} & \xrightarrow{V} & \mathcal{B} & & \\
 K \uparrow & & \Downarrow \beta & & \uparrow K \\
 \mathcal{C} & \xrightarrow{W} & \mathcal{C} & &
 \end{array}$$

Let M be the endofunctor on the comma category J/K mapping $Ja \xrightarrow{f} Kc$ to $(JUa \xrightarrow{\alpha_a} VJa \xrightarrow{Vf} VKc \xrightarrow{\beta_c} KWc)$.

Suppose that J has a right adjoint, that \mathcal{A} and \mathcal{C} are cocomplete and U and W are finitary. Then, given an object $f : Ja \rightarrow Kc$ of J/K , there is a unique morphism $f^\star : JU^\star a \rightarrow KW^\star b$ such that

- $(UU^\star a \rightarrow U^\star a, WW^\star b \rightarrow W^\star b)$ is a morphism from Mf^\star to f^\star ;
- $(a \rightarrow U^\star a, b \rightarrow W^\star b)$ is a morphism from f to f^\star .

Furthermore, $M^\star f = f^\star$.

Proof. This follows from Lemma 9.28 with $F = U, G = W, K$ as RK where R is the right adjoint of J , by considering the mate $\alpha' : UR \rightarrow RV$ of $\alpha : JU \rightarrow VJ$, defined as $UR \xrightarrow{\eta UR} RJUR \xrightarrow{R\alpha R} RVJR \xrightarrow{RV\varepsilon} RV$, (where η and ε denote the unit and the counit of the adjunction $J \dashv R$) and composing it with β to get a natural transformation $URK \rightarrow RKW$. \square

Proof of Lemma 9.27. Let us denote by M the endofunctor on \mathbf{C}_Z^H mapping an object $\mathbf{Z} \leftarrow S \rightarrow \mathbf{Z}$ to the the right pasting of Diagram 9.4:

$$M \left(\begin{array}{ccc} \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 \\ \mathbf{z}_1 \downarrow & S_1 & \downarrow \mathbf{z}_1 \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 \end{array} \right) = \begin{array}{ccccc} & & H_0 & & \\ & & \uparrow & & \downarrow \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 & \xrightarrow{\sim_0^{\otimes}} & \mathbf{Z}_0 \\ \mathbf{z}_1 \downarrow & S_1 & \downarrow \mathbf{z}_1 & \sim_1^{\otimes} & \downarrow \mathbf{z}_1 \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 & \xrightarrow{\sim_0^{\otimes}} & \mathbf{Z}_0 \\ & & \downarrow & & \uparrow \\ & & H_0 & & \end{array}$$

Now, by the packing lemma, it is enough to show that

$$M^\star \left(\begin{array}{ccc} \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 \\ \mathbf{z}_1 \downarrow & S_1 & \downarrow \mathbf{z}_1 \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 \end{array} \right) = \begin{array}{ccccc} & & H_0 & & \\ & & \uparrow & & \downarrow \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 & \xrightarrow{\sim_0^{\otimes*}} & \mathbf{Z}_0 \\ \mathbf{z}_1 \downarrow & S_1 & \downarrow \mathbf{z}_1 & \sim_1^{\otimes*} & \downarrow \mathbf{z}_1 \\ \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 & \xrightarrow{\sim_0^{\otimes*}} & \mathbf{Z}_0 \\ & & \downarrow & & \uparrow \\ & & H_0 & & \end{array}$$

To this end, we are going to organise \mathbf{C}_Z^H as a comma category on which M acts, so as to apply Corollary 9.29. Let \mathcal{A} denote the category of objects S_1 equipped with a span $\mathbf{Z}_1 \leftarrow S_1 \rightarrow \mathbf{Z}_1$ and \mathcal{B} denote the product category $\mathcal{B}_s \times \mathcal{B}_t$, where \mathcal{B}_σ is the category of objects S_1 equipped with a span $\mathbf{Z}_0\sigma \leftarrow S_1 \rightarrow \mathbf{Z}_0\sigma$.

Let $J : \mathcal{A} \rightarrow \mathcal{B}$ denote the functor mapping $\mathbf{Z}_1 \leftarrow S_1 \rightarrow \mathbf{Z}_1$ to $(\mathbf{Z}_0s \leftarrow S_1 \rightarrow \mathbf{Z}_0s, \mathbf{Z}_0t \leftarrow S_1 \rightarrow \mathbf{Z}_0t)$, by postcomposing with the relevant morphisms. As the forgetful functor from a category of spans creates colimits, J is cocontinuous and thus has a right adjoint, since its domain is a locally presentable category. Let $K : 1 \rightarrow \mathcal{B}$ be the functor selecting the pair $(\mathbf{Z}_0s \leftarrow H_0 \rightarrow \mathbf{Z}_0s, \mathbf{Z}_0t \leftarrow H_0 \rightarrow \mathbf{Z}_0t)$. Now, it is straightforward to check that \mathbf{C}_Z^H is isomorphic to the comma category J/K .

Next, we reconstruct M as acting on J/K through this isomorphism in order to fit the setting of Corollary 9.29.

Let

- $U : \mathcal{A} \rightarrow \mathcal{A}$ denote the functor mapping $\mathbf{Z}_1 \leftarrow S_1 \rightarrow \mathbf{Z}_1$ to $\mathbf{Z}_1 \leftarrow S_1 \leftarrow S_1; \sim_1^\otimes \rightarrow \sim_1^\otimes \rightarrow \mathbf{Z}_1$;
- $V : \mathcal{B} \rightarrow \mathcal{B}$ denote the functor $V_s \times V_t$, where $V_\sigma : \mathcal{B}_\sigma \rightarrow \mathcal{B}_\sigma$ maps $\mathbf{Z}_0\sigma \leftarrow S_1 \rightarrow \mathbf{Z}_0\sigma$ to $\mathbf{Z}_0\sigma \leftarrow S_1; \sim_0^\otimes \sigma \rightarrow \mathbf{Z}_0\sigma$;
- $W : 1 \rightarrow 1$ denote the identity endofunctor.

Now we apply Corollary 9.29 with suitable $\alpha : JU \rightarrow VJ$ and $\beta : WK \rightarrow KW$ so that M corresponds to our endofunctor through the isomorphism $\mathbf{C}_Z^H \cong J/K$. Since $U^*(\mathbf{Z}_1 \leftarrow S_1 \rightarrow \mathbf{Z}_1) = \mathbf{Z}_1 \leftarrow S_1; \sim_1^{\otimes*} \rightarrow \mathbf{Z}_1$, the only thing to check is that the proposed definition for $M^*(S_1)$ indeed defines a M -algebra, and that $S_1 \rightarrow U^*(S_1)$ induces a morphism $S_1 \rightarrow M^*(S_1)$, which is straightforward. \square

Let us now turn to our final characterisation of H , which relies on the following category, which is a relaxation of \mathbf{C}_Z^H , in which the left-hand object in (9.3) is only forced to coincide with \mathbf{Z} on $\widehat{\mathbb{C}}_0$.

Definition 9.30. Let \mathbf{C}^H denote the category whose objects consist of objects X_1 and S_1 in \mathbf{C}_1 , equipped with dashed arrows making the following diagram commute.

$$\begin{array}{ccccc}
 \mathbf{Z}_0s & \leftarrow & H_0s & \longrightarrow & \mathbf{Z}_0s \\
 \uparrow \vdots & & \uparrow \vdots & & \uparrow \vdots \\
 X_1 & \leftarrow \cdots & S_1 & \cdots \rightarrow & \mathbf{Z}_1 \\
 \downarrow \vdots & & \downarrow \vdots & & \downarrow \vdots \\
 \mathbf{Z}_0t & \leftarrow & H_0t & \longrightarrow & \mathbf{Z}_0t
 \end{array} \tag{9.6}$$

Remark 9.31. Using the notation of §9.3, an object of \mathbf{C}^H is a cell of the form

$$\begin{array}{ccc}
 \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0 \\
 X_1 \downarrow & S_1 & \downarrow \mathbf{Z}_1 \\
 \mathbf{Z}_0 & \xrightarrow{H_0} & \mathbf{Z}_0.
 \end{array}$$

Proposition 9.32. *The initial object in \mathbf{C}^H is the span $\mathbf{Z}_0 \leftarrow H_0 \rightarrow \mathbf{Z}$, i.e., the one with $X_1 = S_1 = 0$.*

Let us now introduce the endofunctor of which our characterisation of H will be an initial algebra.

Definition 9.33. Let $\Sigma_{1,lax}^H : \mathbf{C}^H \rightarrow \mathbf{C}^H$ map any object $X \leftarrow S \rightarrow \mathbf{Z}$ to the following pasting.

Remark 9.34. The difference with $\Sigma_{1,pack}^H$ is that, X being different from \mathbf{Z} in general, we cannot use any $\check{\Sigma}_1$ -algebra structure on the left.

Proposition 9.35. *The functor $\Sigma_{1,lax}^H : \mathbf{C}^H \rightarrow \mathbf{C}^H$ is finitary.*

Proof. Just as Proposition 9.24. □

Here is our final characterisation.

Lemma 9.36. *The Howe closure is isomorphic to the initial algebra for the functor $\Sigma_{1,lax}^H$.*

Proof. In order to apply Corollary 9.29, we organise \mathbf{C}^H as a comma category J/\mathbf{C}_l^H , decomposing its left and middle/right parts, with $J : \mathbf{C}_r^H \rightarrow \mathbf{C}_l^H$ defined as follows:

- $\mathbf{C}_l^H = \widehat{\mathbb{C}}_1/\Delta(\mathbf{Z}_0)$;
- \mathbf{C}_r^H is the category whose objects consist of a presheaf $S_1 \in \widehat{\mathbb{C}}_1$, together with dashed maps making the following diagram commute;

$$\begin{array}{ccc}
 H_0s & \longrightarrow & \mathbf{Z}_0s \\
 \vdots & & \uparrow \\
 S_1 & \dashrightarrow & \mathbf{Z}_1 \\
 \vdots & & \downarrow \\
 H_0t & \longrightarrow & \mathbf{Z}_0t
 \end{array}$$

- $J : \mathbf{C}_r^H \rightarrow \mathbf{C}_l^H$ maps any such object to the following diagram.

$$\begin{array}{ccc}
 \mathbf{Z}_0s & \longleftarrow & H_0s \\
 & & \uparrow \\
 & & S_1 \\
 & & \downarrow \\
 \mathbf{Z}_0t & \longleftarrow & H_0t
 \end{array}$$

We note that J is cocontinuous since colimits are computed pointwise in \mathbf{C}_l^H and \mathbf{C}_r^H , and thus has a right adjoint R , since its domain is locally presentable. Furthermore, \mathbf{C}_Z^H is isomorphic to the comma category J/\mathbf{Z} .⁴

We then define functors $\Sigma_{1,r}^H$ and $(\check{\Sigma}_1)_{|\mathbf{Z}_0}$ and a natural transformation h as in

⁴Note that this decomposition in a comma category differs from the one in the proof of Lemma 9.27.

By Lemma 9.38, it suffices to show that all non-identity cells in (9.8) are simulations if S is. Let us run through them, left-to-right, top-to-bottom:

- The top cell is a simulation because
 - \mathbf{Z}_0 is the initial $(I + \Sigma_0)$ -algebra, so $I \rightarrow \mathbf{Z}_0 \leftarrow \Sigma_0(\mathbf{Z}_0)$ is a coproduct diagram;
 - similarly, by Proposition 9.17, H_0 is the initial algebra for the endofunctor $X \mapsto (I; \sim_0^{\otimes*} + \Sigma_0(X); \sim_0^{\otimes*})$, so $I; \sim_0^{\otimes*} \rightarrow H_0 \leftarrow \Sigma_0(H_0); \sim_0^{\otimes*}$ is a coproduct diagram;
 - so, by extensivity of \mathbf{C}_0 , both squares in

$$\begin{array}{ccccc}
 I; \sim_0^{\otimes*} & \longrightarrow & H_0 & \longleftarrow & \Sigma_0(H_0); \sim_0^{\otimes*} \\
 \downarrow \lrcorner & & \downarrow & & \lrcorner \downarrow \\
 I & \longrightarrow & \mathbf{Z}_0 & \longleftarrow & \Sigma_0(\mathbf{Z}_0)
 \end{array}$$

are pullbacks. The right-hand one is mapped by Δ_s (which, as a right adjoint, preserves pullbacks) precisely to the top left square of the top cell.

- The first little cell $\Sigma_1^F(S)$ is a simulation by hypothesis.
- The middle little cell is trivially a simulation.
- The third little cell $\sim_1^{\otimes*}$ is a simulation because simulations are closed under transitive closure.
- Finally, the bottom cell is trivially a simulation. □

Finally:

Proof of Lemma 6.17. Because the domains and codomains of all s_L are representable, hence finitely presentable, functional bisimulations are closed under filtered colimits in the arrow category by Lemma 1.1. But H is the initial chain of $\Sigma_{1, lax}^H$, so by Lemma 9.39 it suffices to show that the initial object of \mathbf{C}^H underlies a simulation. By Proposition 9.32, this amounts to showing that the first projection $H_0 \rightarrow \mathbf{Z}_0$ is a functional simulation, which is trivially the case as there are no transitions. □

9.7. Symmetry of transitive closure. In this section, we prove the remaining Lemmas 6.19 and 6.21. Let us first recall the former:

Lemma 6.19. *The relational transitive closure $H_0^{\bar{\dagger}}$ of the Howe closure H_0 on states is symmetric.*

By Lemma 9.7, Lemma 6.19 will follow if we construct a span morphism $H_0 \rightarrow H_0^{\bar{\dagger}}$. As H_0 is an initial algebra for $I + \Sigma_0^H$ (Proposition 9.14), it suffices to prove the following lemma.

Lemma 9.41. *The span $(H^{\bar{\dagger}})_0 = H_0^{\bar{\dagger}}$ has an algebra structure for $(I + \Sigma_0^H)$.*

This relies on the following lemmas, used in particular with $F = \Sigma_0$. The first one is well known:

Lemma 9.42. *Given an endofunctor F on some category \mathcal{C} , the forgetful functor $F\text{-alg} \rightarrow \mathcal{C}$ creates limits, and all colimits that F preserves.*

Lemma 9.43. *Given an endofunctor F on some category \mathcal{C} with pullbacks and coequalisers, if F preserves reflexive coequalisers, then the forgetful functor from the category of F -algebras creates image factorisations.*

Proof. Suppose given an algebra morphism $A \xrightarrow{f} B$. The image factorisation is obtained as the (reflexive) coequaliser of the kernel pair $A \times_f A \rightrightarrows A$. The diagram of this reflexive coequaliser lifts to F -alg, hence so does the coequaliser, by the previous lemma. \square

Proof of Lemma 9.41. We need to find algebra structures on $H_0^{\bar{\tau}\dagger}$ for I , Σ_0 , and $-, \sim_0^\otimes$. For I , we have the morphism $I \rightarrow \mathbf{Z}_0 \rightarrow H_0^\dagger \rightarrow H_0^{\bar{\tau}\dagger}$.

For Σ_0 , note that by Lemmas 9.6 and 9.5, $H_0^{\bar{\tau}\dagger}$ is the colimit of the chain

$$\mathbf{Z}_0 \rightarrow \overline{H_0^\dagger} \cong \overline{H_0^\dagger}; \mathbf{Z}_0 \rightarrow \overline{H_0^\dagger}; H_0^\dagger \cong \overline{H_0^\dagger}; H_0^\dagger; \mathbf{Z}_0 \rightarrow \overline{H_0^\dagger}; H_0^\dagger; H_0^\dagger \rightarrow \dots$$

As it is filtered and thus sifted, and Σ_0 preserves sifted colimits by hypothesis, by Lemma 9.42, it is enough to show that each $\overline{H_0^\dagger}; \dots; H_0^\dagger$ has a structure of Σ_0 -algebra (morphisms in the above chain are then automatically algebra morphisms because the involved spans are relations). But, Σ_0 also preserves reflexive coequalisers (which are sifted colimits), so, by Lemma 9.43, the forgetful functor from Σ_0 -algebras creates image factorisations. It is thus enough to equip $H_0^\dagger; \dots; H_0^\dagger$ with Σ_0 -algebra structure, which is straightforward because H_0 is already an algebra and algebras are stable under pullbacks (Lemma 9.42).

It remains to find a suitable morphism $H_0^{\bar{\tau}\dagger}; \sim_0^\otimes \rightarrow H_0^{\bar{\tau}\dagger}$, or equivalently, by applying $-\dagger$, a morphism $\sim_0^{\otimes\dagger}; H_0^{\bar{\tau}} \rightarrow H_0^{\bar{\tau}}$. But by symmetry of \sim_0^\otimes , we have the composite

$$\sim_0^{\otimes\dagger}; H_0^{\bar{\tau}} \rightarrow \sim_0^\otimes; H_0^{\bar{\tau}} \rightarrow H_0; H_0^{\bar{\tau}} \rightarrow H_0^{\bar{\tau}}. \quad \square$$

Finally, it remains to prove:

Lemma 6.21. *For any substitution-closed simulation R such that R_0 is symmetric, there exists a span morphism $R \rightarrow R'$ such that R' is a substitution-closed bisimulation.*

Proof. First, consider the relation \bar{R} induced by R by the image factorisation $R \twoheadrightarrow \bar{R} \hookrightarrow \mathbf{Z} \times \mathbf{Z}$. \bar{R} is still a substitution-closed simulation and \bar{R}_0 is symmetric. Now, we define R' as follows:

- $R'_0 = \bar{R}_0$
- R'_1 is the limit of the following diagram:

$$\begin{array}{ccccc} \mathbf{Z}_0 \mathbf{s} & \longleftarrow & R'_0 \mathbf{s} & \longrightarrow & \mathbf{Z}_0 \mathbf{s} \\ \uparrow & & \vdots & & \uparrow \\ \mathbf{Z}_1 & \longleftarrow & R'_1 & \dashrightarrow & \mathbf{Z}_1 \\ \downarrow & & \vdots & & \downarrow \\ \mathbf{Z}_0 \mathbf{t} & \longleftarrow & R'_0 \mathbf{t} & \longrightarrow & \mathbf{Z}_0 \mathbf{t}. \end{array}$$

More concretely, an element of $R'_1(c_1)$ is a pair of transitions at c_1 with related sources and targets. The morphism $R \rightarrow R'$ is obtained by the composite $R \rightarrow \bar{R} \rightarrow R'$, where the last morphism exploits the definition of R'_1 as a limit. It is straightforward to check that R' is a substitution-closed simulation. Moreover, it is symmetric (even at the level of transitions), so it is a bisimulation. \square

10. CONCLUSION

We have introduced the notion of Howe context, in which we have defined transition monoids, an abstract notion of labelled transition system whose states feature some sort of substitution. For them, we have introduced an abstract variant of applicative bisimilarity called substitution-closed bisimilarity.

Furthermore, we have introduced dynamic signatures as a device for specifying syntax with variable binding and operational semantics. We have finally shown that if a dynamic signature preserves functional bisimulations, then substitution-closed bisimilarity on the generated transition monoid is a congruence.

This all follows the pattern of our previous work [12], but simplifying the framework and relaxing some hypotheses, as explained in the introduction.

We hope these simplifications pave the way for more abstract results in the same vein. E.g., congruence for other forms of bisimilarity like normal-form [31], environmental [40], or contextual [32] bisimilarity. We also consider investigating abstract type soundness or compiler correctness results.

REFERENCES

- [1] Samson Abramsky. The lazy lambda calculus. In D. A. Turner, editor, *Research Topics in Functional Programming*. Addison–Wesley, 1990.
- [2] J. Adámek and J. Rosický. *Locally Presentable and Accessible Categories*. Cambridge University Press, 1994.
- [3] J. Adámek, J. Rosický, and E. Vitale. What are sifted colimits? *Theory and Applications of Categories*, 23, 2010.
- [4] J. Adámek, J. Rosický, E. M. Vitale, and F. W. Lawvere. *Algebraic Theories: A Categorical Introduction to General Algebra*. Cambridge Tracts in Mathematics. Cambridge University Press, 2010.
- [5] Benedikt Ahrens, André Hirschowitz, Ambroise Lafont, and Marco Maggesi. Reduction monads and their signatures. *PACMPL*, 4(POPL):31:1–31:29, 2020.
- [6] Nathanael Arkor and Marcelo Fiore. Algebraic models of simple type theories: A polynomial approach. In Hermanns et al. [23], pages 88–101.
- [7] Roland Carl Backhouse, Marcel Bijsterveld, Rik van Geldrop, and Jaap van der Woude. Categorical fixed point calculus. In David H. Pitt, David E. Rydeheard, and Peter T. Johnstone, editors, *Proceedings of the 6th International Conference on Category Theory and Computer Science*, volume 953 of *LNCS*, pages 159–179. Springer, 1995.
- [8] Falk Bartels. *On Generalised Coinduction and Probabilistic Specification Formats*. PhD thesis, Vrije Universiteit Amsterdam, 2004.
- [9] Henning Basold, Damien Pous, and Jurriaan Rot. Monoidal company for accessible functors. In Filippo Bonchi and Barbara König, editors, *Proc. 7th International Conference on Algebra and Coalgebra in Computer Science*, volume 72 of *LIPICs*, pages 5:1–5:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.
- [10] B. Bloom, S. Istrail, and A. Meyer. Bisimulation can’t be traced. *Journal of the ACM*, 42:232–268, 1995.
- [11] Martin Bodin, Philippa Gardner, Thomas Jensen, and Alan Schmitt. Skeletal semantics and their interpretations. *PACMPL*, 3(POPL):44:1–44:31, 2019.
- [12] Peio Borthelle, Tom Hirschowitz, and Ambroise Lafont. A cellular Howe theorem. In Hermanns et al. [23].
- [13] Aurelio Carboni and Peter Johnstone. Connected limits, familial representability and artin glueing. *Mathematical Structures in Computer Science*, 5(4):441–459, 1995.
- [14] Andrea Corradini, Reiko Heckel, and Ugo Montanari. Compositional SOS and beyond: a coalgebraic view of open systems. *Theoretical Computer Science*, 280(1-2):163–192, 2002.
- [15] Yves Diers. Spectres et localisations relatifs à un foncteur. *Comptes rendus hebdomadaires des séances de l’Académie des sciences*, 287(15):985–988, 1978.

- [16] Marcelo Fiore and Chung-Kil Hur. On the construction of free algebras for equational systems. *Theoretical Computer Science*, 410:1704–1729, 2009.
- [17] Marcelo Fiore and Philip Saville. List objects with algebraic structure. In Dale Miller, editor, *Proc. 2nd International Conference on Formal Structures for Computation and Deduction*, volume 84 of *LIPICs*, pages 16:1–16:18. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.
- [18] Marcelo Fiore and Philip Saville. Coherence and normalisation-by-evaluation for bicategorical cartesian closed structure. In Hermanns et al. [23], pages 425–439.
- [19] Marcelo Fiore and Philip Saville. Relative full completeness for bicategorical cartesian closed structure. volume 12077 of *LNCS*, pages 277–298. Springer, 2020.
- [20] Marcelo P. Fiore, Gordon D. Plotkin, and Daniele Turi. Abstract syntax and variable binding. In *Proc. 14th Symposium on Logic in Computer Science IEEE*, 1999.
- [21] Richard H. G. Garner and Tom Hirschowitz. Shapely monads and analytic functors. *Journal of Logic and Computation*, 28(1):33–83, 2018.
- [22] Makoto Hamana. Free s-monoids: A higher-order syntax with metavariables. In Wei-Ngan Chin, editor, *Proc. 2nd Asian Symposium on Programming Languages and Systems*, volume 3302 of *LNCS*, pages 348–363. Springer, 2004.
- [23] Holger Hermanns, Lijun Zhang, Naoki Kobayashi, and Dale Miller, editors. *Proc. 35th ACM/IEEE Symposium on Logic in Computer Science ACM*, 2020.
- [24] André Hirschowitz, Tom Hirschowitz, and Ambroise Lafont. Modules over monads and operational semantics. In Zena M. Ariola, editor, *5th International Conference on Formal Structures for Computation and Deduction, FSCD 2020, June 29-July 6, 2020, Paris, France (Virtual Conference)*, volume 167 of *LIPICs*, pages 12:1–12:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.
- [25] Tom Hirschowitz. Cellular Monads from Positive GSOS Specifications. In *26th International Workshop on Expressiveness in Concurrency and 16th Workshop on Structural Operational Semantics*, volume 300, pages 1–18, Amsterdam, Netherlands, 2019.
- [26] Tom Hirschowitz. Familial monads and structural operational semantics. *PACMPL*, 3(POPL):21:1–21:28, 2019.
- [27] Mark Hovey. *Model Categories*, volume 63 of *Mathematical Surveys and Monographs, Volume 63, AMS (1999)*. American Mathematical Society, 1999.
- [28] André Joyal, Mogens Nielsen, and Glynn Winskel. Bisimulation and open maps. In *Proc. 8th Symposium on Logic in Computer Science*, pages 418–427. IEEE, 1993.
- [29] Ambroise Lafont. Initial semantics for substitution skew-monoids. <https://github.com/amblafont/Skew-MonoidalCategories>, 2020. Coq code.
- [30] Ugo Dal Lago, Francesco Gavazzo, and Paul Blain Levy. Effectful applicative bisimilarity: Monads, relators, and howe’s method. In *Proc. 32nd Symposium on Logic in Computer Science*, pages 1–12. IEEE, 2017.
- [31] S. Lassen. Bisimulation in untyped lambda calculus: Böhm trees and bisimulation up to context. In Stephen D. Brookes, Achim Jung, Michael W. Mislove, and Andre Scedrov, editors, *Proc. 15th Conference on Mathematical Foundations of Programming Semantics, MFPS 1999, Tulane University, New Orleans, LA, USA, April 28 - May 1, 1999*, volume 20 of *Electronic Notes in Theoretical Computer Science*, pages 346–374. Elsevier, 1999.
- [32] Sergueï Lenglet and Alan Schmitt. Howe’s method for contextual semantics. volume 42, pages 212–225. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.
- [33] Saunders Mac Lane and Ieke Moerdijk. *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. Universitext. Springer, 1992.
- [34] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, I/II. *Information and Computation*, 100(1):1–77, 1992.
- [35] Jeffrey C. Morton. Double bicategories and double cospans. *Journal of Homotopy and Related Structures*, 4(1):389–428, 2009.
- [36] C.-H. L. Ong. The concurrent lambda calculus I: A general precongruence theorem for applicative bisimulation. In *Proceedings on Seminars on Parallel Programming Systems*, pages 139–164. Department of Information Systems and Computer Science, National University of Singapore, 1992.
- [37] Andrew M. Pitts. *Howe’s method for higher-order languages*, chapter 5. Number 52 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2011.

- [38] Jan Reiterman. A left adjoint construction related to free triples. *Journal of Pure and Applied Algebra*, 10:57–71, 1977.
- [39] Davide Sangiorgi. The lazy lambda calculus in a concurrency scenario. *Information and Computation*, 111:120–153, 1994.
- [40] Davide Sangiorgi, Naoki Kobayashi, and Eijiro Sumii. Environmental bisimulations for higher-order languages. *ACM Trans. Program. Lang. Syst.*, 33(1):5:1–5:69, 2011.
- [41] Davide Sangiorgi and David Walker. *The π -calculus – A Theory of Mobile Processes*. Cambridge University Press, 2001.
- [42] Sam Staton. General structural operational semantics through categorical logic. In *Proc. 23rd Symposium on Logic in Computer Science*, pages 166–177, 2008.
- [43] Kornel Szlachányi. Skew-monoidal categories and bialgebroids. *Advances in Mathematics*, 231:1694–1730, 2012.
- [44] Daniele Turi and Gordon D. Plotkin. Towards a mathematical operational semantics. In *Proc. 12th Symposium on Logic in Computer Science*, pages 280–291, 1997.
- [45] Mark Weber. Generic morphisms, parametric representations and weakly cartesian monads. *Theory and Applications of Categories*, 13:191–234, 2004.
- [46] Mark Weber. Familial 2-functors and parametric right adjoints. *Theory and Applications of Categories*, 18(22):665–732, 2007.