Infinitary Howe's Method

Paul Blain Levy [1]

University of Birmingham, U.K.

Abstract

Howe's method is a well-known technique for proving that various kinds of applicative bisimilarity (or similarity) on a functional language are congruences (or precongruences). It proceeds by constructing an extension of the given relation that has certain special properties.

The method can be used for deterministic and for erratically nondeterministic languages, but in the latter case it has a strange limitation: it requires the language's syntax to be finitary. That excludes, for example, languages with countable sum types, and has repeatedly caused problems in the literature.

In this paper, we give a variation on Howe's method, called "infinitary Howe's method", that avoids this problem. The method involves defining two extensions of the original relation by mutual coinduction. Both extensions possess the key properties of Howe's extension, but it is their intersection that is compatible.

In the first part of the paper, we see how this works for a call-by-value language with countable sum types. In the second part, we see that the method continues to work when we make the syntax non-well-founded. More precisely, we show, using a mixed inductive/coinductive argument, that the various forms of applicative similarity and bisimilarity are preserved by any substituting context.

Key words: Howe's method, applicative bisimulation, nondeterminism, coinductive, infinitary syntax, call-by-value

1 Introduction

1.1 Applicative Simulation On Deterministic Languages

The notions of *applicative* simulation and bisimulation on a deterministic λ-calculus were introduced in [1]. These mimic the notions of simulation and bisimulation from concurrency theory. A closed term is seen rather like a process that evaluates to a λ-abstraction, and then waits to be supplied with an operand. As with other forms of simulation/bisimulation, it is necessary, for

[1] Email: pbl@cs.bham.ac.uk

This paper is electronically published in
Electronic Notes in Theoretical Computer Science
URL: www.elsevier.nl/locate/entcs

these to be useful, to prove that the greatest such (called applicative *similarity* and *bisimilarity*) are precongruences, when extended to non-closed terms. This was proved in [1] by denotational means.

Howe [6] introduced a purely operational technique for proving that applicative similarity is a precongruence, known as "Howe's method". The technique consists of extending similarity to a relation that is obviously compatible[2], and possesses some special properties that cause it to be a simulation. Hence it coincides with similarity.

That paper did not provide a proof that applicative *bisimilarity* is a congruence. But in the deterministic setting, it is not necessary to prove this directly. What one can do instead is to first show that observational preorder and equivalence are, respectively, simulations and bisimulations. It follows that

- applicative similarity and observational preorder coincide
- applicative bisimilarity, mutual applicative similarity and observational equivalence all coincide.

This line of reasoning is presented in [5,14].

In a nondeterministic[3] setting, however, all these coincidences fail: applicative bisimilarity is strictly finer than mutual applicative similarity, which in turn is strictly finer than observational equivalence [8,13]. (This is to simplify matters somewhat, as there are various kinds of applicative similarity and bisimilarity, and of observational equivalence.) Moreover, it is at least arguable that applicative bisimilarity is a more natural equivalence on a nondeterministic functional language than observational equivalence. So the question of proving applicative bisimilarity to be a congruence becomes important.

In a second paper [7], Howe solved this problem by proving that the *transitive closure* of the Howe extension is symmetric. A generalization of this method was given in [13] to prove that *refinement similarity*—a variant of bisimilarity that is not symmetric—is a precongruence. The argument uses the following "cuboid lemma". (R^* is the reflexive transitive closure of R.)

Proposition 1.1 *Let R_i be a reflexive binary relation on A_i for $i \in I$. If I is finite, then*

$$\prod_{i\in I}(R_i^*) = (\prod_{i\in I} R_i)^*$$

as relations on $\prod_{i\in I} A_i$.

The $\subseteq$ direction (which is the one that requires I to be finite) says that, given a cuboid in a finite number of dimensions, there is a finite path from one vertex to the opposite vertex

---

[2] A *compatible* relation is one that is preserved by every term constructor in the language.

[3] More precisely, in an *erratically* nondeterministic setting. Howe's method (and the variant in this paper) cannot be applied a language with McCarthy's amb [9], nor to many calculi of concurrency.

Now suppose $\mathcal{S}$ is the compatible closure of a preorder. Clearly it is reflexive. It might not be transitive, but its transitive closure $\mathcal{S}^*$ is preserved by any term constructor θ by Prop. 1.1, setting I to be the arity of θ. Thus $\mathcal{S}^*$ is compatible. This is the essence of the argument, both Howe's version and Pitcher's.

But this has a curious limitation: it can only work when every term constructor is finitary. That is a strange restriction, because it is entirely syntactic. From a semantic viewpoint, one often wants to study languages with, e.g., countable sum types. It is, therefore, unsurprising that this limitation has repeatedly caused problems in the literature.

- In [13], a nondeterministic language with countable sum types and countable product types is studied. As explained on page 142, Howe's method cannot prove that bisimilarity is a congruence in general—only for a restricted class of fragments.
- Later in [13], refinement similarity is studied. As explained on page 150, Howe's method cannot prove that it is a precongruence in general—only for an even more restricted class of fragments.
- Independently, in [12], a nondeterministic language HOPLA with countable sum types is studied; but it cannot be shown that applicative bisimilarity is a congruence (page 8, property (vi)).

The contribution of this paper is to give a variant of Howe's method called "infinitary Howe's method", which can be used to prove congruence of bisimilarity (and precongruence of refinement bisimilarity) for nondeterministic languages with infinitary syntax. It consists of defining *two* extensions of the original relation—the "forward and backward extensions"—by mutual coinduction. (For a finitary language, these are, respectively, the Howe extension and its dual.) Each of these possesses the same special properties enjoyed by Howe's extension that are used to show simulation. The forward and backward extensions are not compatible—but their intersection is, and this is sufficient.

1.2 Non-Well-Founded Syntax

Having shown that infinitary Howe's method is applicable to a language with infinitely wide syntax, we then apply it to a harder situation: a language with non-well-founded syntax. The difficulty here is the need to show that bisimilarity is preserved by non-well-founded contexts, but we see that the method accomplishes this.

Our account relies on a relational calculus that was developed in [5,8]. Because of the complex mixing of induction and coinduction, it would be difficult to spell out the argument without using the calculus.

1.3 Structure Of Paper

In this paper, an increasing sequence of three languages are studied:

- $\mathcal{L}_0$, whose term syntax is finitary
- $\mathcal{L}_1$, whose term syntax is infinitely wide
- $\mathcal{L}_2$, whose term syntax is non-well-founded.

These are call-by-value languages with countable nondeterminism.

Having defined $\mathcal{L}_0$ and $\mathcal{L}_1$, and the various forms of applicative similarity and bisimilarity, we review Howe's method, and recall how it proves that

(i) various forms of applicative similarity are precongruences on $\mathcal{L}_0$ and $\mathcal{L}_1$

(ii) various forms of applicative bisimilarity and refinement similarity are precongruences on $\mathcal{L}_0$.

We divide this review into two parts. One part, which we designate the "core", is common to Howe's method and infinitary Howe's method. It describes the properties of Howe's extension, and shows how they imply that the extension is a simulation. In the second part, we give the specific construction of Howe's extension and shows how it has the required properties, proving (i). We also see how to use Prop. 1.1 to prove (ii).

We then describe infinitary Howe's method, constructing the forward and backwards extensions and showing they have the desired properties to show that the various forms of applicative bisimilarity and refinement similarity are precongruences on $\mathcal{L}_1$.

Finally, we proceed to $\mathcal{L}_2$, which we define in Sect. 7.1. After considering what it means for a relation to be closed under all contexts (Sect. 7.2), we show in Sect. 7.3 that the intersection of the forwards and backwards extensions satisfies this property. Hence the various forms of applicative similarity and bisimilarity all have this property.

2 A Call-By-Value Calculus

We define languages $\mathcal{L}_0$ and $\mathcal{L}_1$. The types of $\mathcal{L}_0$ are given as follows:

$$\textbf{coinductive definition} \qquad A ::= \; \sum_{i \in I} A_i \;\mid\; A \to A$$

where I ranges over finite sets. (Product types could be included without difficulty.) We write 0 for the empty sum type, and `nat` for the unique type A such that $A = (0 \to 0) + A$.

The types of $\mathcal{L}_1$ are the same, except that I ranges over countable sets.

A *context* is a sequence $\mathtt{x}_0 : A_0, \ldots, \mathtt{x}_{n-1} : A_{n-1}$ of distinct identifiers with associated types. A *renaming* $\Gamma \xrightarrow{q} \Delta$ is a function taking each identifier $(\mathtt{x} : A) \in \Gamma$ to an identifier $(q(\mathtt{y})) \in \Delta$.

The calculus, as in [8], distinguishes values from ordinary terms (it is not clear how to make Howe's method work without this distinction). So there are two judgements: $\Gamma \vdash M : B$ means that M is a term of type B, and $\Gamma \vdash^{\mathsf{v}} V : B$ means that V is a value of type B. This style of call-by-value

λ-calculus is called *fine-grain*. The syntax is defined inductively in Fig. 1.

$$\frac{}{\Gamma \vdash ? : \mathtt{nat}}$$

$$\frac{}{\Gamma, \mathtt{x} : A, \Gamma' \vdash^{\mathsf{v}} \mathtt{x} : A} \qquad \frac{\Gamma \vdash^{\mathsf{v}} V : A \quad \Gamma, \mathtt{x} : A \vdash M : B}{\Gamma \vdash \mathtt{let}\ V\ \mathtt{be}\ \mathtt{x}.\ M : B}$$

$$\frac{\Gamma \vdash^{\mathsf{v}} V : A}{\Gamma \vdash \mathtt{return}\ V : A} \qquad \frac{\Gamma \vdash M : A \quad \Gamma, \mathtt{x} : A \vdash N : B}{\Gamma \vdash M\ \mathtt{to}\ \mathtt{x}.\ N : B}$$

$$\frac{\Gamma \vdash^{\mathsf{v}} V : A_{\hat{\imath}}}{\Gamma \vdash^{\mathsf{v}} \langle \hat{\imath}, V \rangle : \sum_{i \in I} A_i}\ \hat{\imath} \in I \qquad \frac{\Gamma \vdash^{\mathsf{v}} V : \sum_{i \in I} A_i \quad \Gamma, \mathtt{x} : A_i \vdash M_i : B\ \ (\forall i \in I)}{\Gamma \vdash \mathtt{pm}\ V\ \mathtt{as}\ \{\langle i, \mathtt{x} \rangle . M_i\}_{i \in I} : B}$$

$$\frac{\Gamma, \mathtt{x} : A \vdash M : B}{\Gamma \vdash^{\mathsf{v}} \lambda \mathtt{x}.M : A \to B} \qquad \frac{\Gamma \vdash^{\mathsf{v}} V : A \to B \quad \Gamma \vdash^{\mathsf{v}} W : A}{\Gamma \vdash VW : B}$$

Fig. 1. Syntax Of Fine-Grain CBV With Countable Nondeterminism

We write M `to` $\mathtt{x}.\ N$ for the sequenced computation that first executes M, and when, this returns a value V proceeds to execute N with $\mathtt{x}$ bound to V. This was written in Moggi's syntax using `let`, but we reserve `let` for mere binding. The keyword `pm` stands for "pattern-match". For each $n \in \mathbb{N}$, the closed value $\underline{n}$ of type `nat` is defined in the obvious way.

Any term or value is uniquely of the form $\theta\{M_i\}_{i \in I}$, where

- θ is a term constructor of arity I (a finite set for $\mathcal{L}_0$, a countable set for $\mathcal{L}_1$)
- $\{M_i\}_{i \in I}$ are the immediate subterms of M, which may be terms or values.

In particular, each identifier is a term constructor of arity 0.

Let Γ and Δ be contexts.

- A renaming $\Gamma \xrightarrow{q} \Delta$ can be applied to any term $\Gamma \vdash M : B$ to obtain a term $\Delta \vdash q^{\dagger} M : B$, and likewise to a value.
- A *substitution* $\Gamma \xrightarrow{\overrightarrow{V/\mathtt{x}}} \Delta$ is a function taking each identifier $(\mathtt{x} : A) \in \Gamma$ to a value $\Delta \vdash V_{\mathtt{x}} : A$. We can apply this to a term $\Gamma \vdash M : B$ to obtain a term $\Delta \vdash M[\overrightarrow{V/\mathtt{x}}] : B$, and likewise to a value.

These operations are defined inductively [3,4].

The operational behaviour of a closed term M of type A is given in three parts [11,8]:

- a relation $M \Downarrow V$, where V a closed value of type A, meaning that M may return V

- a predicate $M \Uparrow$, meaning that M may diverge
- a relation $M \Downarrow_{\Box} \mathcal{V}$, where $\mathcal{V}$ is a set of closed values of type A, meaning that M must return something, and $\mathcal{V}$ is the set of possibilities.

These relations are defined in Fig. 2. They are related by the following result.

Proposition 2.1 *Let M be a closed term of type A, and $\mathcal{V}$ a set of closed values of type A. Then $M \Downarrow_{\Box} \mathcal{V}$ iff $M \not\Uparrow$ and $\mathcal{V} = \{V \mid M \Downarrow V\}$.*

3 Relations

3.1 Basic Constructions

Because this paper uses a lot of reasoning about relations, we gather together the basic properties here.

Definition 3.1 (i) A *closed relation* R associates to each type A a binary relation on the closed terms inhabiting it, and a binary relation on the closed values inhabiting it.

(ii) An *open relation* $\mathcal{R}$ associates to each sequent $\Gamma \vdash A$ a binary relation on the terms inhabiting it, and to each value sequent $\Gamma \vdash^{\mathsf{v}} A$ a binary relation on the values inhabiting it, all preserved by $q^{\dagger}$ for any renaming $\Gamma \xrightarrow{q} \Delta$.

(iii) We write id for the identity relation on terms and values, and idf for the identity relation restricted to identifiers. (These are both open relations.)

(iv) We write ; for relational composition, in diagrammatic order.

(v) If $\mathcal{R}$ is an open relation, we write $\mathcal{R}_0$ for the restriction of $\mathcal{R}$ to closed terms and closed values.

(vi) Let R be a closed relation. We define R° (the *open extension* of $\mathcal{R}$) to be the open relation that relates two terms $\Gamma \vdash M, N : B$ when $M[\overrightarrow{V/\mathtt{x}}] \, R \, N[\overrightarrow{V/\mathtt{x}}]$ for any substitution $\overrightarrow{V/\mathtt{x}}$ from Γ to the empty context.

Notice that the poset of closed relations and the poset of open relations each forms a complete lattice under inclusion. Therefore, when we define an open relation using monotone functions, least prefixed points and greatest postfixed points, we do not need to prove the renaming condition for the resulting relation—it is automatic.

Definition 3.2 Let $\mathcal{R}$ and $\mathcal{S}$ be open relations. We define $\mathcal{R}[\mathcal{S}]$ (the *substitution of $\mathcal{S}$ into $\mathcal{R}$*) to be the open relation consisting of the pairs of terms $\Delta \vdash M[\overrightarrow{V/\mathtt{x}}], N[\overrightarrow{W/\mathtt{x}}] : B$ for every pair of terms $\Gamma \vdash M', N' : B$ and pair of substitutions $\Gamma \xrightarrow{\overrightarrow{V/\mathtt{x}}} \Delta$ and $\Gamma \xrightarrow{\overrightarrow{W/\mathtt{x}}} \Delta$ such that $M' \, \mathcal{R} \, N'$ and $V_{\mathtt{x}} \, \mathcal{S} \, V'_{\mathtt{x}}$ for each $(\mathtt{x} : A) \in \Gamma$.

May Convergence (inductive definition)

$$\frac{}{?\Downarrow \underline{n}}\; n \in \mathbb{N} \qquad \frac{M[W/\mathtt{x}] \Downarrow V}{\mathtt{let}\ W\ \mathtt{be}\ \mathtt{x}.\ M \Downarrow V}$$

$$\frac{}{\mathtt{return}\ V \Downarrow V} \qquad \frac{M \Downarrow W \quad N[W/\mathtt{x}] \Downarrow V}{M\ \mathtt{to}\ \mathtt{x}.\ N \Downarrow V}$$

$$\frac{M[W/\mathtt{x}] \Downarrow V}{(\lambda \mathtt{x}.M)W \Downarrow V} \qquad \frac{M_{\hat{\imath}}[W/\mathtt{x}] \Downarrow V}{\mathtt{pm}\ \langle \hat{\imath}, W\rangle\ \mathtt{as}\ \{\langle i, \mathtt{x}\rangle.M_i\}_{i\in I} \Downarrow V}\; \hat{\imath} \in I$$

Divergence (coinductive definition)

$$\frac{M[W/\mathtt{x}] \Uparrow}{\mathtt{let}\ W\ \mathtt{be}\ \mathtt{x}.\ M \Uparrow}$$

$$\frac{M \Uparrow}{M\ \mathtt{to}\ \mathtt{x}.\ N \Uparrow} \qquad \frac{M \Downarrow V \quad N[V/\mathtt{x}] \Uparrow}{M\ \mathtt{to}\ \mathtt{x}.\ N \Uparrow}$$

$$\frac{M[W/\mathtt{x}] \Uparrow}{(\lambda \mathtt{x}.M)W \Uparrow} \qquad \frac{M_{\hat{\imath}}[W/\mathtt{x}] \Uparrow}{\mathtt{pm}\ \langle \hat{\imath}, W\rangle\ \mathtt{as}\ \{\langle i, \mathtt{x}\rangle.M_i\}_{i\in I} \Uparrow}\; \hat{\imath} \in I$$

Must convergence (inductive definition)

$$\frac{}{?\Downarrow_{\Box} \{\underline{n} \,|\, n \in \mathbb{N}\}} \qquad \frac{M[W/\mathtt{x}] \Downarrow_{\Box} \mathcal{V}}{\mathtt{let}\ W\ \mathtt{be}\ \mathtt{x}.\ M \Downarrow_{\Box} \mathcal{V}}$$

$$\frac{}{\mathtt{return}\ V \Downarrow_{\Box} \{V\}} \qquad \frac{M \Downarrow_{\Box} \mathcal{W} \quad N[W/\mathtt{x}] \Downarrow_{\Box} \mathcal{V}_W \ \ (\forall W \in \mathcal{W})}{M\ \mathtt{to}\ \mathtt{x}.\ N \Downarrow_{\Box} \bigcup_{W\in\mathcal{W}} \mathcal{V}_W}$$

$$\frac{M[W/\mathtt{x}] \Downarrow_{\Box} \mathcal{V}}{(\lambda \mathtt{x}.M)W \Downarrow_{\Box} \mathcal{V}} \qquad \frac{M_{\hat{\imath}}[W/\mathtt{x}] \Downarrow_{\Box} \mathcal{V}}{\mathtt{pm}\ \langle \hat{\imath}, W\rangle\ \mathtt{as}\ \{\langle i, \mathtt{x}\rangle.M_i\}_{i\in I} \Downarrow_{\Box} \mathcal{V}}\; \hat{\imath} \in I$$

Fig. 2. Operational Semantics

Definition 3.3 Let $\mathcal{R}$ be an open relation.

(i) We define $\widehat{\mathcal{R}}$ (the *compatible refinement* of $\mathcal{R}$) to be the open relation that relates two terms $\theta\{M_i\}_{i\in I}$ and $\phi\{N_j\}_{j\in J}$ when $\theta = \phi$ (hence $I = J$), and $M_i \,\mathcal{R}\, N_i$ for each $i \in I$.

(ii) We define $\mathcal{R}^{\mathsf{fin}}$ the same way, except that I must be finite. (For $\mathcal{L}_0$, this coincides with $\widehat{\mathcal{R}}$.

(iii) We define $\widetilde{\mathcal{R}}$ the same way, except that θ must not be an identifier.

Proposition 3.4 (i) *All the operations given above are monotone.*

(ii) *The complete lattice of open relations forms an ordered monoid under the binary operation* $-[-]$, *with unit given by* idf.

(iii)

$$
\begin{aligned}
\mathsf{id}[\mathsf{id}] &= \mathsf{id} && (1)\\
\mathcal{R}^{\mathsf{op}}[\mathcal{S}^{\mathsf{op}}] &= (\mathcal{R}[\mathcal{S}])^{\mathsf{op}} && (2)\\
(\mathcal{R};\mathcal{R}')[\mathcal{S};\mathcal{S}'] &= (\mathcal{R}[\mathcal{S}]);(\mathcal{R}'[\mathcal{S}']) && (3)\\
(\bigcup_{i\in I} R_i)[\mathcal{S}] &= \bigcup_{i\in I}(\mathcal{R}_i[\mathcal{S}]) && (4)\\
(\mathcal{R}[\mathcal{S}])^* &\subseteq \mathcal{R}^*[\mathcal{S}^*] && (5)\\
R^{\circ}[\mathsf{id}] &= R^{\circ} && (6)\\
{R^{\circ}}_0 &= R && (7)\\
\mathcal{S} &\subseteq {(\mathcal{S}[\mathsf{id}])_0}^{\circ} && (8)\\
\widehat{\mathcal{R}} &= \widetilde{\mathcal{R}} \cup \mathsf{idf} \quad \text{and} \quad \widetilde{\mathcal{R}} \cap \mathsf{idf} = \emptyset && (9)\\
\mathsf{idf} \subseteq \mathcal{R}^{\mathsf{fin}} &\subseteq \widehat{\mathcal{R}} && (10)\\
\widetilde{\mathcal{R}}[\mathcal{S}] &\subseteq \widetilde{\mathcal{R}[\mathcal{S}]} && (11)\\
\widehat{\mathsf{id}} &= \mathsf{id} && (12)
\end{aligned}
$$

(iv) *If* $\mathcal{R}$ *and* $\mathcal{S}$ *are reflexive open relations then*

$$
\begin{aligned}
&\mathcal{R}^*[\mathcal{S}^*] = (\mathcal{R}[\mathcal{S}])^* && (13)\\
&\mathcal{R}^{*\mathsf{fin}} = \mathcal{R}^{\mathsf{fin}*} && (14)
\end{aligned}
$$

Proof. (11) follows from the renaming assumption on $\mathcal{R}$. (iv) follows from Prop. 1.1. The rest is trivial. □

Definition 3.5 Let $\mathcal{S}$ be an open relation.

(i) $\mathcal{S}$ is *substitutive* when $\mathsf{idf} \subseteq \mathcal{S}$ and $\mathcal{S}[\mathcal{S}] \subseteq \mathcal{S}$.

(ii) $\mathcal{S}$ is *compatible* when $\widehat{\mathcal{S}} \subseteq \mathcal{S}$.

(iii) $\mathcal{S}$ is *finitely compatible* when $\mathcal{S}^{\mathsf{fin}} \subseteq \mathcal{S}$. (For $\mathcal{L}_0$, this coincides with compatibility.)

Proposition 3.6 (not valid in Sect. 7) *Let* $\mathcal{R}$ *be an open relation.*

(i) *If* $\mathcal{R}$ *is compatible, then it is reflexive.*

(ii) *There is a unique open relation* $\mathcal{S}$ *such that* $\mathcal{S} = \mathcal{R} \cup \widehat{\mathcal{S}}$, *and it is the least compatible open relation containing* $\mathcal{R}$.

We write $\mathcal{R}^{\mathsf{C}}$, the *compatible closure* of $\mathcal{R}$, for the open relation described in Prop. 3.6(ii).

Proposition 3.7 *[5] Let f be a monotone endofunction on a lattice A, and let $x \in A$.*

(i) *(strong induction) Suppose f has least prefixed point a. Then $f(x \wedge a) \wedge a \leqslant x$ implies $a \leqslant x$.*

(ii) *(strong coinduction) Suppose f has greatest postfixed point b. Then $x \leqslant f(x \vee b) \vee b$ implies $x \leqslant b$.*

4 Applicative Similarity

Definition 4.1 A closed relation R *respects values* when

- $V\,R\,V' : A \to B$ implies $VW\,R\,V'W : B$ for every closed value $W : A$
- $\langle \hat{\imath}, V\rangle\,R\,\langle \hat{\imath}', V'\rangle : \sum_{i\in I} A_i$ implies $\hat{\imath} = \hat{\imath}'$ and $V\,R\,V' : A_{\hat{\imath}}$

In [8,13], three variants of applicative simulation are studied, corresponding to lower, upper and convex powerdomains. Here, we introduce a fourth variant called "smash", intermediate between upper and convex.

Definition 4.2 Let R be a closed relation.

(i) We say that R is a *lower simulation* when it respects values, and $M\,R\,M'$ and $M \Downarrow V$ implies $M' \Downarrow V'$ for some V' such that $V\,R\,V'$.

(ii) We say that R is an *upper simulation* when it respects values, and $M\,R\,M'$ and $M \Downarrow_{\Box} \mathcal{V}$ implies $M' \Downarrow_{\Box} \mathcal{V}'$ where $\forall V' \in \mathcal{V}'.\ \exists V \in \mathcal{V}.\ V\,R\,V'$.

(iii) We say that R is a *smash simulation* when it respects values, and $M\,R\,M'$ and $M \Downarrow_{\Box} \mathcal{V}$ implies $M' \Downarrow_{\Box} \mathcal{V}'$ where $\forall V' \in \mathcal{V}'.\ \exists V \in \mathcal{V}.\ V\,R\,V'$ and $\forall V \in \mathcal{V}.\ \exists V' \in \mathcal{V}'.\ V\,R\,V'$.

(iv) We say that R is a *convex simulation* (aka *partial bisimulation* [2]) when it is both a lower simulation and an upper simulation (hence also a smash simulation).

(v) We say that R is a lower/upper/smash/convex *opsimulation* when R^{op} is a lower/upper/smash/convex simulation.

We define lower/upper/smash/convex *similarity* to be the largest closed relation that is a lower/upper/smash/convex simulation.

We can define a host of closed relations by combining simulations and opsimulations. The following two examples suffice for our purposes.

Definition 4.3 (i) *Lower bisimilarity* is the largest closed relation that is both a lower simulation and a lower opsimulation.

(ii) *Refinement similarity* is the largest closed relation that is both a lower simulation and an upper opsimulation.

All of these relations are clearly preorders. Our aim is to show that their open extensions (which are also preorders) are precongruences.

5 Howe's Method

5.1 The Core of the Method

Let R be a closed relation that is a preorder. In this section we review Howe's method for proving that the R° is a precongruence. It centres on finding an open relation satisfying the following.

Definition 5.1 Let R be a closed preorder and let $\mathcal{S}$ be an open relation. $\mathcal{S}$ is *Howe-suitable* over R when

- $\mathcal{S}$ is reflexive, substitutive and finitely compatible
- $\mathcal{S}; R^\circ \subseteq \mathcal{S}$
- If $\theta\{M_i\}_{i\in I}\, \mathcal{S}\, N$ then there exists $\{M'_i\}_{i\in I}$ such that $M_i\, \mathcal{S}\, M'_i$ for each $i \in I$ and $\theta\{M'_i\}_{i\in I}\, R^\circ\, N$. In short, $\mathcal{S} \subseteq \widehat{\mathcal{S}}; R^\circ$.

Dually, $\mathcal{S}$ is *op-Howe-suitable* over R when it is reflexive, substitutive and finitely compatible, and $R^\circ; \mathcal{S} \subseteq \mathcal{S} \subseteq R^\circ; \widehat{\mathcal{S}}$

Def. 5.1 is significant because of the following theorems.

Proposition 5.2 *Let $\mathcal{S}$ be an open relation that is Howe-suitable or op-Howe-suitable over the closed preorder R.*

(i) $R^\circ \subseteq \mathcal{S}$

(ii) *If $\mathcal{S}_0 \subseteq R$ (e.g. if $(\mathcal{S}^*)_0 \subseteq R$), then $R^\circ = \mathcal{S} = \mathcal{S}^*$.*

(iii) *If R respects values, then so does $\mathcal{S}_0$, and hence so does $\mathcal{S}^*_0$.*

Proof.

(i) $R^\circ = \mathsf{id}; R^\circ \subseteq \mathcal{S}; R^\circ \subseteq \mathcal{S}$

(ii) By (8).

(iii) Suppose $\langle \hat{\imath}, V\rangle\, \mathcal{S}_0\, \langle \hat{\imath}', V'\rangle$. Then there exists V'' such that $V\, \mathcal{S}_0\, V''$ and $\langle \hat{\imath}, V''\rangle\, R\, \langle \hat{\imath}', V'\rangle$. Because R respects values, $\hat{\imath} = \hat{\imath}'$ and $V''\, R\, V'$ so $V \mathcal{S} V'$. The other requirement holds because $\mathcal{S}$ is reflexive and finitely compatible.

□

Proposition 5.3 (Howe simulation theorem) (i) *Let $\mathcal{S}$ be an open relation that is Howe-suitable over the closed preorder R. If R is a lower/upper/smash/convex simulation, then so is $\mathcal{S}_0$, and hence so is $\mathcal{S}^*_0$.*

(ii) *Dually, let $\mathcal{S}$ be an open relation that is Howe-suitable over R. If R is a lower/upper/smash/convex opsimulation, then so is $\mathcal{S}_0$, and hence so is $\mathcal{S}^*_0$.*

Proof.

(i) Suppose that R is a lower simulation. We have to show that $M\, \mathcal{S}_0\, N$ and $M \Downarrow V$ implies $M' \Downarrow V'$ for some V' such that $V\, \mathcal{S}_0\, V'$. We proceed by induction on $M \Downarrow V$.

Suppose that $M = (\lambda \mathtt{x}.M')W$. Then we have $M'[W/\mathtt{x}] \Downarrow V$, and there exists P and W' such that $\lambda \mathtt{x}.M' \mathcal{S} P$ and $W \mathcal{S} W'$ and $PW' R N$. From $\lambda \mathtt{x}.M' \mathcal{S} P$, there exists M'' such that $M' \mathcal{S} M''$ and $\lambda \mathtt{x}.M'' R P$. Since $M' \mathcal{S} M''$ and $W \mathcal{S} W'$, we obtain $M'[W/\mathtt{x}] \mathcal{S} M''[W'/\mathtt{x}]$, so, by inductive hypothesis, there exists V'' such that $V \mathcal{S} V''$ and $M''[W'/\mathtt{x}] \Downarrow V''$, so $(\lambda \mathtt{x}.M'')W' \Downarrow V''$. We have $(\lambda \mathtt{x}.M'')W' R PW' R N$ (because R respects values), and R is a lower simulation, so there exists V' such that $N \Downarrow V'$ and $V'' R V'$. Hence $V \mathcal{S} V'$.

The other cases are similar but easier.

The result for upper simulations and for smash simulations is proved similarly, and the result for convex simulations is then immediate.

□

5.2 *Howe's Extension*

Howe's extension is defined as follows.

Proposition 5.4 *Let R be a closed relation. Then there is a unique relation $\mathcal{S}$ such that $\mathcal{S} = \widehat{\mathcal{S}}; R^\circ$, which we call $R^\bullet$. Dually, there exists a unique relation $\mathcal{S}$ such that $\mathcal{S} = R^\circ; \widehat{\mathcal{S}}$, which we call $R^\S$.*

Proof. If $\mathcal{S}$ and $\mathcal{S}'$ are two such, we prove that $M \mathcal{S} N$ implies $M \mathcal{S}' N$ by induction on N. □

As a unique fixpoint, $R^\bullet$ can be defined either inductively or coinductively. Here is the inductive definition, written out explicitly: $R^\bullet$ is the least relation $\mathcal{S}$ such that, if $M_i \mathcal{S} M'_i$ for all $i \in I$, and $\theta\{M'_i\}_{i \in I} R^\circ N$, then $\theta\{M_i\}_{i \in I} \mathcal{S} N$.

Proposition 5.5 *Let R be a closed preorder.*

(i) *$R^\bullet$ is Howe-suitable over R, and $R^\S$ is op-Howe-suitable over R.*

(ii) *$R^\bullet$ and $R^\S$ are compatible.*

Proof. This can be proved from either the inductive definition or the coinductive definition of $R^\bullet$. Here is the inductive version.

(ii) is trivial, and all the requirements of Howe-suitability of $R^\bullet$ other than substitutivity follow immediately. For substitutivity, we have to prove that if $\Gamma \vdash M R^\bullet N : B$ and $V(\mathtt{x}) R^\bullet V'(\mathtt{x})$ for each $(\mathtt{x} : A) \in \Gamma$ then $M[\overrightarrow{V/\mathtt{x}}] R^\bullet N[\overrightarrow{V'/\mathtt{x}}]$. We proceed by induction on $M R^\bullet N$ (using the inductive definition), treating separately the case that M is an identifier and the case that it is not. □

Now, if we write R for lower similarity, then, by Prop. 5.3(i), $R^\bullet{}_0$ is a lower simulation, hence contained in R, so by Prop. 5.2(ii), R° is equal to $R^\bullet$, which is compatible. So R° is compatible. By the same argument, the open extensions of upper, smash and convex similarity are all compatible. Dually, we can use the op-Howe extension to show (directly) that the open extensions of lower, upper, smash and convex opsimilarity are compatible.

Next, we treat lower bisimilarity and refinement similarity, following [7,13].

Proposition 5.6 (i) *If $\mathcal{R}$ is an open preorder on $\mathcal{L}_0$, then $\mathcal{R}^{\mathsf{C}*}$ is compatible.*

(ii) *If R is a closed preorder on $\mathcal{L}_0$, then*

$$R^{\bullet *} = R^{\circ \mathsf{C}*} = R^{\S *}$$

Moreover, $R^{\bullet}$ is the only relation Howe-suitable over R, and $R^{\S}$ is the only relation op-Howe-suitable over R.

(iii) *If R is a closed equivalence relation on $\mathcal{L}_0$, then $R^{\bullet *}$ is symmetric.*

Proof.

(i) This follows from Prop. 1.1.

(ii) $R^{\bullet} \supseteq R^{\circ \mathsf{C}}$ because $R^{\bullet}$ is compatible and contains R°. $R^{\bullet} \subseteq R^{\circ \mathsf{C}}$ by the inductive definition of $R^{\bullet}$, using (i). The second equation is dual to the first. If $\mathcal{S}$ is Howe-suitable over R, then $\mathcal{S} = \widehat{\mathcal{S}}; \mathcal{R}$, so $\mathcal{S} = R^{\bullet}$ by Prop. 5.4.

(iii) In general, $R^{\S} = R^{\mathsf{op}\bullet\mathsf{op}}$, so this follows from (ii).

□

So if R is refinement similarity on $\mathcal{L}_0$, then $R^{\bullet *}$ is both a lower simulation and, being $R^{\S *}$, an upper opsimulation. So it is contained in R and we obtain $R = R^{\bullet}$, so R is compatible. Similarly for lower bisimilarity.

However, this method does not work for $\mathcal{L}_1$, so we turn to infinitary Howe's method, which does.

6 Infinitary Howe's Method

We come now to the key construction: the *forwards extension* of R, written $R^{\rightarrow}$, and the *backwards extension* of R, written $R^{\leftarrow}$.

Definition 6.1 Let R be a closed preorder. We define $(R^{\rightarrow}, R^{\leftarrow})$ to be the greatest pair of open relations $(\mathcal{S}, \mathcal{T})$ such that

- if $M = \theta\{M_i\}_{i \in I} \,\mathcal{S}\, N$, then there exists $\{M'_i\}_{i \in I}$ such that $M_i \,\mathcal{S}\, M'_i$ for all $i \in I$ and $\theta\{M'_i\}_{i \in I} \,R^{\circ}\, N$, and $M \,\mathcal{T}^*\, N$
- if $N \,\mathcal{T}\, M = \theta\{M_i\}_{i \in I}$, then there exists $\{M'_i\}_{i \in I}$ such that $N \,R^{\circ}\, \theta\{M'_i\}_{i \in I}$ and $M'_i \,\mathcal{T}\, M_i$ for all $i \in I$, and $N \,\mathcal{S}^*\, M$.

In short, $(R^{\rightarrow}, R^{\leftarrow})$ is

$$\nu(\mathcal{S}, \mathcal{T}).((\widehat{\mathcal{S}}; R^{\circ}) \cap \mathcal{T}^*, (R^{\circ}; \widehat{\mathcal{T}}) \cap \mathcal{S}^*)$$

In two special cases, we can simplify this definition.

Proposition 6.2 (i) *If R is a closed preorder on $\mathcal{L}_0$, then $R^{\rightarrow} = R^{\bullet}$ and $R^{\leftarrow} = R^{\S}$.*

(ii) *If R is a closed equivalence relation, then $R^{\leftarrow} = R^{\rightarrow^{\mathsf{op}}}$, and $R^{\rightarrow}$ is the greatest open relation $\mathcal{S}$ such that if $M = \theta\{M_i\}_{i\in I}\,\mathcal{S}\,N$, then there exists $\{M_i'\}_{i\in I}$ such that $M_i\,\mathcal{S}\,M_i'$ for all $i \in I$ and $\theta\{M_i'\}_{i\in I}\,R^{\circ}\,N$, and $N\,\mathcal{S}^*\,M$. In short, $R^{\rightarrow}$ is $\nu\mathcal{S}.((\widehat{\mathcal{S}};R^{\circ}) \cap \mathcal{S}^{\mathsf{op}*})$.*

Proof. Plain coinduction in both cases, using Prop. 5.6(iii) in (i). □

In general, to prove $\mathcal{S} \subseteq R^{\rightarrow}$ and $\mathcal{T} \subseteq R^{\leftarrow}$ using Prop. 3.7(ii), it suffices to prove

$$\mathcal{S} \subseteq (\widehat{\mathcal{S} \cup R^{\rightarrow}}; R^{\circ}) \cup R^{\rightarrow} \tag{15}$$
$$\mathcal{S} \subseteq (\mathcal{T} \cup R^{\leftarrow})^* \cup R^{\rightarrow} \tag{16}$$
$$\mathcal{T} \subseteq (R^{\circ}; \widehat{\mathcal{T} \cup R^{\leftarrow}}) \cup R^{\leftarrow} \tag{17}$$
$$\mathcal{T} \subseteq (\mathcal{S} \cup R^{\rightarrow})^* \cup R^{\rightarrow} \tag{18}$$

When (15)–(18) are satisfied, we say that the pair $(\mathcal{S}, \mathcal{T})$ is *good*. In all our examples, the proof of (17)–(18) is dual to that of (15)–(16), so we omit it.

Proposition 6.3 *Let R be a preorder on closed terms.*

(i) $R^{\rightarrow *} = R^{\leftarrow *}$

(ii) *$R^{\rightarrow}$ is Howe-suitable for R, and $R^{\leftarrow}$ is op-Howe-suitable for R.*

(iii) *$R^{\rightarrow} \cap R^{\leftarrow}$ is compatible.*

Proof.

(i) $R^{\rightarrow} \subseteq R^{\leftarrow *}$, so $R^{\rightarrow *} \subseteq R^{\leftarrow *}$. By the same argument, $R^{\leftarrow *} \subseteq R^{\rightarrow *}$.

(ii) We firstly show (R°, R°) to be good, which implies that $R^{\rightarrow}$ and $R^{\leftarrow}$ are reflexive. To prove (15) $R^{\circ} = \mathsf{id}; R^{\circ} = \widehat{\mathsf{id}}; R^{\circ} \subseteq$ RHS. (16) is trivial.

Next we show that $(R^{\rightarrow}; R^{\circ}, R^{\circ}; R^{\leftarrow})$ is good. To prove (15), $R^{\rightarrow}; R^{\circ} \subseteq (\widehat{R^{\rightarrow}}; R^{\circ}); R^{\circ} = \widehat{R^{\rightarrow}; \mathsf{id}}; R^{\circ} \subseteq \widehat{R^{\rightarrow}; R^{\circ}}; R^{\circ} \subseteq$ RHS. To prove (16), $R^{\rightarrow}; R^{\circ} \subseteq R^{\leftarrow *}; R^{\circ} = (\mathsf{id}; R^{\leftarrow})^*; (R^{\circ}; \mathsf{id}) \subseteq (R^{\circ}; R^{\leftarrow})^*; (R^{\circ}; R^{\leftarrow}) \subseteq$ RHS.

Next we show that $(R^{\rightarrow \mathsf{fin}}, R^{\leftarrow \mathsf{fin}})$ is good. To prove (15) for this pair,

$$R^{\rightarrow \mathsf{fin}} \subseteq \widehat{R^{\rightarrow}} \subseteq (\widehat{R^{\rightarrow \mathsf{fin}} \cup R^{\rightarrow}}; R^{\circ}) \cup R^{\rightarrow}$$

To prove (16) for this pair,

$$R^{\rightarrow \mathsf{fin}} \subseteq R^{\leftarrow * \mathsf{fin}} \subseteq R^{\leftarrow \mathsf{fin} *} \subseteq (R^{\leftarrow \mathsf{fin}} \cup R^{\leftarrow})^* \cup R^{\rightarrow}$$

To prove $R^{\rightarrow}$ and $R^{\leftarrow}$ substitutive, we show that $(R^{\rightarrow}[R^{\rightarrow}], R^{\leftarrow}[R^{\leftarrow}])$ is good. To prove (15) for this pair,

$$
\begin{aligned}
R^{\rightarrow}[R^{\rightarrow}] \quad & (\widehat{R^{\rightarrow}}; R^{\circ})[R^{\rightarrow}; \mathsf{id}] \\
& \subseteq (\widehat{R^{\rightarrow}}[R^{\rightarrow}]); (R^{\circ}[\mathsf{id}]) \\
& \subseteq ((\widetilde{R^{\rightarrow}} \cup \mathsf{idf})[R^{\rightarrow}]); R^{\circ} \\
& \subseteq \widetilde{R^{\rightarrow}}[R^{\rightarrow}]; R^{\circ} \cup \mathsf{idf}[R^{\rightarrow}]; R^{\circ} \\
& \subseteq \widetilde{R^{\rightarrow}[R^{\rightarrow}]}; R^{\circ} \cup R^{\rightarrow}; R^{\circ} \\
& \subseteq \widetilde{R^{\rightarrow}[R^{\rightarrow}]}; R^{\circ} \cup R^{\rightarrow}
\end{aligned}
$$

which $\subseteq$ the RHS. To prove (16) for this pair,

$$R^{\rightarrow}[R^{\rightarrow}] \subseteq R^{\leftarrow *}[R^{\leftarrow *}] \subseteq (R^{\leftarrow}[R^{\leftarrow}])^{*}$$

(iii) It is easily shown that $(\widehat{R^{\rightarrow} \cap R^{\leftarrow}}, \widehat{R^{\rightarrow} \cap R^{\leftarrow}})$ is good. A stronger result is proved in detail below (Prop. 7.6).

$\square$

To illustrate how we can use this, let R be refinement similarity. Then by Prop. 5.3, $R^{\rightarrow *}_{\;0}$ is a lower simulation and $R^{\leftarrow *}_{\;0}$ is an upper opsimulation, but they are the same, hence contained in R. By Prop. 5.2(ii)

$$R^{\circ} = R^{\rightarrow} = R^{\leftarrow} = R^{\rightarrow} \cap R^{\leftarrow}$$

so R° is compatible.

7 Non-Well-Founded Syntax

7.1 Adapting The Well-Founded Account

We now come to $\mathcal{L}_2$, in which the term syntax is non-well-founded. The syntax of types is the same as that of $\mathcal{L}_1$ (so there are countable sum types). To define the term syntax, we might be tempted to make all the rules of Fig. 1 coinductive, but that would give us "infinite values" such as $\langle i_0, \langle i_1, \langle i_2, \ldots \rangle\rangle\rangle$, which ought not to exist [4]. We therefore need to ensure that values are given inductively and terms are given coinductively.

Write valseq for the set of value sequents $\Gamma \vdash^{\mathsf{v}} B$ and $\mathsf{termseq}$ for the set of term sequents $\Gamma \vdash B$. For any $\mathsf{termseq}$-indexed set X, we define the valseq-indexed set $\mathsf{val}(X)$ inductively by the rules in Fig. 3. Then we define a $\mathsf{termseq}$-indexed set P coinductively in Fig. 4. Finally, we write

- $\Gamma \vdash M : A$ for $M \in P\ (\Gamma \vdash A)$
- $\Gamma \vdash^{\mathsf{v}} V : A$ for $V \in \mathsf{val}(P)\ (\Gamma \vdash^{\mathsf{v}} A)$.

Renaming and substitution are defined coinductively, as in [10]

[4] In λ-calculus with sum types and non-well-founded syntax, this is indeed a term, but under the call-by-value evaluation strategy, it diverges.

$$\frac{}{\mathtt{x} \in \mathsf{val}(X)\ (\Gamma \vdash^{\mathsf{v}} A)}\ (\mathtt{x} : A) \in \Gamma \qquad \frac{M \in X\ (\Gamma, \mathtt{x} : A \vdash B)}{\lambda \mathtt{x}.M \in \mathsf{val}(X)\ (\Gamma \vdash^{\mathsf{v}} A \to B)}$$

$$\frac{V \in \mathsf{val}(X)\ (\Gamma \vdash^{\mathsf{v}} A_{\hat{\imath}})}{\langle \hat{\imath}, V \rangle \in \mathsf{val}(X)\ (\Gamma \vdash^{\mathsf{v}} \sum_{i \in I} A_i)}\ \hat{\imath} \in I$$

Fig. 3. Values—inductive definition of $\mathsf{val}(X)$, a set indexed by value sequents

$$\frac{}{? \in P(\Gamma \vdash \mathtt{nat})} \qquad \frac{V \in \mathsf{val}(P)(\Gamma \vdash^{\mathsf{v}} A) \quad M \in P\ (\Gamma, \mathtt{x} : A \vdash B)}{\mathtt{let}\ V\ \mathtt{be}\ \mathtt{x}.\ M \in P\ (\Gamma \vdash B)}$$

$$\frac{V \in \mathsf{val}(P)(\Gamma \vdash^{\mathsf{v}} A)}{\mathtt{return}\ V \in P(\Gamma \vdash^{\mathsf{v}} A)} \qquad \frac{M \in P\ (\Gamma \vdash A) \quad N \in P\ (\Gamma, \mathtt{x} : A \vdash B)}{M\ \mathtt{to}\ \mathtt{x}.\ N \in P\ (\Gamma \vdash B)}$$

$$\frac{V \in P(\Gamma \vdash^{\mathsf{v}} A \to B) \quad W \in P(\Gamma \vdash^{\mathsf{v}} A)}{VW \in P(\Gamma \vdash B)}$$

$$\frac{V \in \mathsf{val}(P)\ (\Gamma \vdash^{\mathsf{v}} \sum_{i \in I} A_i) \quad M_i \in P\ (\Gamma, \mathtt{x} : A_i \vdash B)\ (\forall i \in I)}{\mathtt{pm}\ V\ \mathtt{as}\ \{(i, \mathtt{x}).M_i\}_{i \in I} \in P\ (\Gamma \vdash B)}$$

Fig. 4. Terms—coinductive definition of P, a set indexed by term sequents

The operational semantics is defined by Fig. 2 just as before, and the various notions of applicative similarity are defined exactly as in Sect. 4. We use infinitary Howe's method to prove that the open extension of each one is compatible just as in Sect. 5.1 and Sect. 6.

As for Prop. 5.5, listing the properties of the Howe extension, to make this valid, we define $R^{\bullet}$ to be the greatest fixpoint $\nu\mathcal{S}.(\widehat{\mathcal{S}}; R^{\circ})$. The least fixpoint would not even be reflexive.

7.2 Closure Under Contexts

Definition 7.1 A *closure operator* on a poset set A is a monotone endofunction f on A such that $x \leqslant fx = f(fx)$ for all $x \in A$. Those elements x such that $fx \leqslant x$ (i.e. $fx = x$) are said to be *f-closed.*

By standard order theory, the compatible relations are the closed elements of the endofunction mapping an open relation $\mathcal{R}$ to the least compatible relation containing it, viz. $\mu\mathcal{S}.(\mathcal{R} \cup \widehat{\mathcal{S}})$. This latter relation can be thought of as the closure of $\mathcal{R}$ under all *well-founded* contexts (which may have countably many holes, each occurring countably many times). But we would like to know that relations are closed under *all* contexts. So we proceed as follows.

Definition 7.2 Let $\mathcal{R}$ be a relation. Its *closure under binding contexts* $\mathcal{R}^{\mathsf{C}}$ is the relation $\nu\mathcal{S}.(\mathcal{R} \cup \widehat{\mathcal{S}})$.

Proposition 7.3 (i) $-^{\mathsf{C}}$ *is a closure operator.*
(ii) $\mathcal{R}^{\mathsf{C}}$ *is reflexive and compatible.*

Proof.

(i) $\mathcal{R} \subseteq \mathcal{R} \cup \widehat{\mathcal{R}}$, so $\mathcal{R} \subseteq \mathcal{R}^{\mathsf{C}}$. And

$$\mathcal{R}^{\mathsf{C}^{\mathsf{C}}} = \mathcal{R}^{\mathsf{C}} \cup \widehat{\mathcal{R}^{\mathsf{C}^{\mathsf{C}}}} = \mathcal{R} \cup \widehat{\mathcal{R}^{\mathsf{C}}} \cup \widehat{\mathcal{R}^{\mathsf{C}^{\mathsf{C}}}} \subseteq \mathcal{R} \cup \widehat{\mathcal{R}^{\mathsf{C}^{\mathsf{C}}}}$$

so $\mathcal{R}^{\mathsf{C}^{\mathsf{C}}} \subseteq \mathcal{R}^{\mathsf{C}}$ by plain coinduction.

(ii) $\mathsf{id} = \widehat{\mathsf{id}} \subseteq \mathcal{R} \cup \widehat{\mathsf{id}}$, so $\mathsf{id} \subseteq \mathcal{R}^{\mathsf{C}}$. Compatibility follows Lambek's Lemma.

□

Binding contexts are so named, because they bind the identifiers in the plugged terms. A more general kind of context is called a *substituting* context, which may subsitute given values for identifiers in the plugged terms. A first suggestion for closing $\mathcal{R}$ under subsituting contexts is the relation

$$\mathcal{Q} = \nu\mathcal{S}.\ (\mathcal{R}[\mathcal{S}] \cup \widehat{\mathcal{S}}) \tag{19}$$

A pair of terms is in this relation iff it is at the root of a proof tree in which certain nodes are compatibility nodes

$$\frac{M_i \,\mathcal{Q}\, M_i' \ \ (\forall i \in I)}{\theta\{M_i\}_{i\in I} \,\mathcal{Q}\, \theta\{M_i'\}_{i\in I}}$$

where α is a term constructor of arity I, and the other nodes are substitution nodes

$$\frac{V_0 \,\mathcal{Q}\, V_0' \ \ \cdots \ \ V_{n-1} \,\mathcal{Q}\, V_{n-1}'}{M[V_0/\mathtt{x}_0,\ldots,V_{n-1}/\mathtt{x}_1] \,\mathcal{Q}\, M'[V_0'/\mathtt{x}_0,\ldots,V_{n-1}'/\mathtt{x}_1]}$$

where M, M' are in context $\mathtt{x}_0 : A_0, \ldots, \mathtt{x}_{n-1} : A_{n-1}$ (either terms of the same type, or values of the same type.)

The problem with (19) is that it is the universal relation. Instead we need to constrain the proof trees so that, moving along a branch away from the root, there are only finitely many consecutive substitution nodes, so that one eventually hits a compatibility node. We make this precise in the following way.

Definition 7.4 Let $\mathcal{R}$ be a relation. Its *closure under substituting contexts* $\mathcal{R}^{\mathsf{SC}}$ is the relation $\nu\mathcal{S}.\ \mu\mathcal{T}.\ (\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{S}})$.

Proposition 7.5 (i) $\mathcal{R}^{\mathsf{C}} \subseteq \mathcal{R}^{\mathsf{SC}}$.

(ii) $-^{\mathsf{SC}}$ *is a closure operator.*

(iii) $\mathcal{R}^{\mathsf{SC}}[\mathcal{R}^{\mathsf{SC}}] \subseteq \mathcal{R}^{\mathsf{SC}}$

Proof.

(i) We reason

$$\begin{aligned}
\mathcal{R}^{\mathsf{C}} &= \mathcal{R} \cup \widehat{\mathcal{R}^{\mathsf{C}}} \\
&= \mathcal{R}[\mathsf{idf}] \cup \widehat{\mathcal{R}^{\mathsf{C}}} \\
&\subseteq \mathcal{R}[\widehat{\mathcal{R}^{\mathsf{C}}}] \cup \widehat{\mathcal{R}^{\mathsf{C}}} \\
&\subseteq \mathcal{R}[\mathcal{R}[\mu\mathcal{T}.\ (\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{C}}})] \cup \widehat{\mathcal{R}^{\mathsf{C}}}] \cup \widehat{\mathcal{R}^{\mathsf{C}}} \\
&= \mathcal{R}[\mu\mathcal{T}.\ (\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{C}}})] \cup \widehat{\mathcal{R}^{\mathsf{C}}} \\
&= \mu\mathcal{T}.\ (\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{C}}})
\end{aligned}$$

Hence $\mathcal{R}^{\mathsf{C}} \subseteq \nu\mathcal{S}.\mu\mathcal{T}.\ (\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{S}})$.

(ii) Clearly $\mathcal{R} \subseteq \mathcal{R}^{\mathsf{C}} \subseteq \mathcal{R}^{\mathsf{SC}}$. We note that

$$\mathcal{R}^{\mathsf{SC}} = \mu\mathcal{T}.(\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{SC}}}) \tag{20}$$

To show $\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}} \subseteq \mathcal{R}^{\mathsf{SC}}$, it suffices by plain coinduction to show

$$\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}} \subseteq \mu\mathcal{T}.(\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}})$$

and we abbreviate the RHS by $\mathcal{T}'$. Since the LHS is $\mu\mathcal{T}.(\mathcal{R}^{\mathsf{SC}}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}})$, it suffices, by Prop. 3.7(i), to show

$$\mathcal{R}^{\mathsf{SC}}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}} \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}} \subseteq \mathcal{T}' \tag{21}$$

It is clear that $\widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}}$ is contained in the RHS of (21), so it suffices to prove

$$\mathcal{R}^{\mathsf{SC}}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \subseteq \mathcal{T}'$$

This is equivalent to saying that $\mathcal{R}^{\mathsf{SC}}$ is contained in the relation $\mathcal{Q}$ defined as follows. Two computations $\Delta \vdash^{\mathsf{c}} M, M' : \underline{B}$ are related by $\mathcal{Q}$ when for any context morphisms $\Gamma \xrightarrow{k,k'} \Delta$ related by $\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}$, the computations $\Gamma \vdash^{\mathsf{c}} k^*M, k'^*M' : \underline{B}$ are related by $\mathcal{T}'$, and likewise for values.

Since $\mathcal{R}^{\mathsf{SC}}$ is $\mu\mathcal{T}.(\mathcal{R}[\mathcal{T}] \cup \widehat{\mathcal{R}^{\mathsf{SC}}})$, it suffices, by plain induction, to show

$$\mathcal{R}[\mathcal{Q}] \cup \widehat{\mathcal{R}^{\mathsf{SC}}} \subseteq \mathcal{Q}$$

$\mathcal{R}[\mathcal{Q}] \subseteq \mathcal{Q}$ is given by

$$\mathcal{R}[\mathcal{Q}][\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \subseteq \mathcal{R}[\mathcal{Q}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}]] \subseteq \mathcal{R}[\mathcal{T}'] \subseteq \mathcal{R}[\mathcal{T}'] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}} = \mathcal{T}'$$

For $\widehat{\mathcal{R}^{\mathsf{SC}}} \subseteq \mathcal{Q}$, we have to show $\widehat{\mathcal{R}^{\mathsf{SC}}}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \subseteq \mathcal{T}'$. This is given by

$$
\begin{aligned}
\widetilde{\mathcal{R}^{\mathsf{SC}}}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] &= (\widetilde{\mathcal{R}^{\mathsf{SC}}} \cup \mathsf{idf})[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \\
&= \widetilde{\mathcal{R}^{\mathsf{SC}}}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \cup \mathsf{idf}[\mathcal{T}' \cap \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \\
&\subseteq \widetilde{\mathcal{R}^{\mathsf{SC}}}[\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \cup \mathsf{idf}[\mathcal{T}'] \\
&\subseteq \widetilde{\mathcal{R}^{\mathsf{SC}}[\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}]} \cup \mathcal{T}' \\
&\subseteq \widehat{\mathcal{R}^{\mathsf{SC}}[\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}}} \cup (\mathcal{T}' \cup \mathcal{T}') \\
&= \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}} \cup \mathcal{T}' \cup (\mathcal{R}[\mathcal{T}'] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}}) \\
&= \mathcal{T}' \cup (\mathcal{R}[\mathcal{T}'] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}}) = \mathcal{T}' \cup \mathcal{T}' = \mathcal{T}'
\end{aligned}
$$

(iii) $$\mathcal{R}^{\mathsf{SC}}[\mathcal{R}^{\mathsf{SC}}] \subseteq \mathcal{R}^{\mathsf{SC}}[\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}] \cup \widehat{\mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}}} = \mathcal{R}^{\mathsf{SC}^{\mathsf{SC}}} = \mathcal{R}^{\mathsf{SC}}$$

□

A relation $\mathcal{R}$ is *closed under substituting contexts* when $\mathcal{R}^{\mathsf{SC}} \subseteq \mathcal{R}$. By Prop. 7.5, every such relation is closed under contexts, compatible, reflexive and substitutive.

7.3 Applicative Similarity Is Closed Under Substituting Contexts

To adapt our proofs of compatibility to proofs of closure under substituting contexts, we strengthen Prop. 5.5(ii) and Prop. 6.3(iii) as follows.

Proposition 7.6 *Let R be a closed relation. Then $R^{\bullet}$ and $R^{\S}$ and $R^{\rightarrow} \cap R^{\leftarrow}$ are closed under substituting contexts.*

Proof. We just give the proof for $R^{\rightarrow} \cap R^{\leftarrow}$, which we abbreviate as R^{g}. To show $R^{\mathsf{g}\mathsf{SC}} \subseteq R^{\mathsf{g}}$, it suffices to show that the pair $(R^{\rightarrow}[R^{\mathsf{g}\mathsf{SC}}], R^{\leftarrow}[R^{\mathsf{g}\mathsf{SC}}])$ is good, because that implies

$$R^{\mathsf{g}\mathsf{SC}} = \mathsf{idf}[R^{\mathsf{g}\mathsf{SC}}] \subseteq R^{\rightarrow}[R^{\mathsf{g}\mathsf{SC}}] \cap R^{\leftarrow}[R^{\mathsf{g}\mathsf{SC}}] \subseteq R^{\rightarrow} \cap R^{\leftarrow}$$

First we prove (15) for this pair. If we can show

$$R^{\mathsf{g}\mathsf{SC}} \subseteq \widehat{R^{\rightarrow}[R^{\mathsf{g}\mathsf{SC}}]}; R^{\circ} \tag{22}$$

then we can deduce (15), by

$$
\begin{aligned}
R^{\to}[R^{\mathsf{gSC}}] &\subseteq (\widehat{R^{\to}}; R^{\circ})[R^{\mathsf{gSC}}; \mathsf{id}] \\
&\subseteq (\widehat{R^{\to}}[R^{\mathsf{gSC}}]); (R^{\circ}[\mathsf{id}]) \\
&\subseteq (\widetilde{R^{\to}} \cup \mathsf{idf})[R^{\mathsf{gSC}}]; R^{\circ} \\
&= (\widetilde{R^{\to}}[R^{\mathsf{gSC}}]; R^{\circ}) \cup (\mathsf{idf}[R^{\mathsf{gSC}}]; R^{\circ}) \\
&\subseteq (\widetilde{R^{\to}[(R^{\mathsf{gSC}})]}; R^{\circ}) \cup (R^{\mathsf{gSC}}; R^{\circ}) \\
&\subseteq (\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cup (\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}; R^{\circ}) \\
&= \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ} \\
&\subseteq \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ} \cup R^{\to}
\end{aligned}
$$

To prove (22), since R^{gSC} can be expressed as $\mu\mathcal{T}.(R^{\mathsf{g}}[\mathcal{T}] \cup \widehat{R^{\mathsf{gSC}}})$, by Prop. 3.7(i) it suffices to show

$$
(R^{\mathsf{g}}[(\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}] \cup \widehat{R^{\mathsf{gSC}}}) \cap R^{\mathsf{gSC}} \subseteq \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}
$$

which is equivalent to

$$
R^{\mathsf{g}}[(\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}] \cap R^{\mathsf{gSC}} \subseteq \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ} \tag{23}
$$

$$
\widehat{R^{\mathsf{gSC}}} \cap R^{\mathsf{gSC}} \subseteq \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ} \tag{24}
$$

For (23), the LHS is contained in

$$
\begin{aligned}
& R^{\to}[(\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}] \\
\subseteq\; & (\widehat{R^{\to}}; R^{\circ})[((\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}); \mathsf{id}] \\
\subseteq\; & (\widehat{R^{\to}}[(\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}]); (R^{\circ}[\mathsf{id}]) \\
\subseteq\; & ((\widetilde{R^{\to}} \cup \mathsf{idf})[(\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cap R^{\mathsf{gSC}}]); R^{\circ} \\
\subseteq\; & ((\widetilde{R^{\to}}[R^{\mathsf{gSC}}]) \cup (\mathsf{idf}[\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}])); R^{\circ} \\
\subseteq\; & (\widetilde{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}) \cup ((\widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}); R^{\circ})
\end{aligned}
$$

which is contained in the RHS of (23). For (24), we reason

$$
\widehat{R^{\mathsf{gSC}}} \cap R^{\mathsf{gSC}} \subseteq \widehat{\mathsf{idf}[R^{\mathsf{gSC}}]}; \mathsf{idf} \subseteq \widehat{R^{\to}[R^{\mathsf{gSC}}]}; R^{\circ}
$$

That completes the proof of (15). To prove (16) for this pair, we reason

$$
\begin{aligned}
R^{\to}[R^{\mathsf{gSC}}] &\subseteq (R^{\leftarrow *}; \mathsf{id})[\mathsf{id}; R^{\mathsf{gSC}}] \\
&\subseteq R^{\leftarrow *}[\mathsf{id}]; \mathsf{id}[R^{\mathsf{gSC}}] \\
&\subseteq (R^{\leftarrow}[\mathsf{id}])^{*}; (R^{\leftarrow}[R^{\mathsf{gSC}}]) \\
&\subseteq (R^{\leftarrow}[R^{\mathsf{gSC}}])^{*}; (R^{\leftarrow}[R^{\mathsf{gSC}}]) \\
&\subseteq (R^{\leftarrow}[R^{\mathsf{gSC}}])^{*} \cup R^{\to}
\end{aligned}
$$

□

8 Adapting The Method

To apply infinitary Howe's method to any calculus with binding, all that is needed is an analogue of Prop. 5.3. The rest of the theory is purely syntactic.

Acknowledgements I thank Soren Lassen for his comments on this paper.

References

[1] Abramsky, S., *The lazy λ-calculus*, in: *Research topics in Functional Programming*, Addison Wesley, 1990 pp. 65–117.

[2] Abramsky, S., *A domain equation for bisimulation*, Information and Computation **92** (1991).

[3] Altenkirch, T. and B. Reus, *Monadic presentations of λ-terms using generalized inductive types*, in: J. Flum and M. Rodríguez-Artalejo, editors, *Proc. of 13th Int. Workshop on Computer Sci. Logic, 1999*, LNCS **1683**, Springer, 1999 .

[4] Fiore, M., G. D. Plotkin and D.Turi, *Abstract syntax and variable binding*, in: G. Longo, editor, *Proceedings of the 14th Annual Symposium on Logic in Computer Science (LICS'99)* (1999), pp. 193–202.

[5] Gordon, A. D., *Bisimilarity as a theory of functional programming*, Theor. Comput. Sci. **228** (1999), pp. 5–47.

[6] Howe, D. J., *Equality in lazy computation systems*, in: *Proc. 4th IEEE Symposium on Logic in Computer Science*, 1989, pp. 198–203.

[7] Howe, D. J., *Proving congruence of bisimulation in functional programming languages*, Inf. and Comp. **124** (1996).

[8] Lassen, S., "Relational Reasoning about Functions and Nondeterminism," Ph.D. thesis, Univ. of Aarhus (1998).

[9] Lassen, S. B. and A. K. Moran, *Unique fixed point induction for McCarthy's amb*, in: *Proceedings of the 24th International Symposium on Mathematical Foundations of Computer Science*, "LNCS" **1672** (1999), pp. 198–208.

[10] Matthes and Uustalu, *Substitution in non-wellfounded syntax with variable binding*, TCS: Theoretical Computer Science **327** (2004).

[11] Moran, A., "Natural Semantics for Non-Determinism," Master's thesis, Dept. of Computing Science, Chalmers University of Technology (1994).

[12] Nygaard, M. and G. Winskel, *HOPLA—A higher-order process language*, in: *Proc., 13th Intl. Conf. on Concurrency Theory*, LNCS **2421**, 2002.

[13] Pitcher, C., "Functional Programming and Erratic Non-Determinism," Ph.D. thesis, Oxford Univ. (2001).

[14] Pitts, A. M., *Operationally-based theories of program equivalence*, in: P. Dybjer and A. M. Pitts, editors, *Semantics and Logics of Computation*, Publications of the Newton Institute, Cambridge University Press, 1997 pp. 241–298.