Math. Struct. in Comp. Science (1999), *vol.* 9, *pp.* 403–435. Printed in the United Kingdom
© 1999 Cambridge University Press

Coinductive characterizations of applicative structures†

FURIO HONSELL‡ and MARINA LENISA§

‡*Università di Udine, Italy*
Email: honsell@dimi.uniud.it

§*University of Edinburgh, Scotland*
Email: lenisa@dcs.ed.ac.uk

Received 20 October 1998; revised 13 April 1999

We discuss *new* ways of characterizing, as *maximal fixed points* of monotone operators, observational congruences on λ-terms and, more generally, equivalences on applicative structures. These characterizations naturally induce *new* forms of *coinduction principles* for reasoning on program equivalences, which are not based on Abramsky's *applicative bisimulation*. We discuss, in particular, what we call the *cartesian* coinduction principle, which arises when we exploit the elementary observation that functional behaviours can be expressed as *cartesian graphs*. Using the paradigm of final semantics, the soundness of this principle over an applicative structure can be expressed easily by saying that the applicative structure can be construed as a *strongly extensional coalgebra* for the functor $(\mathscr{P}(_ \times _)) \oplus (\mathscr{P}(_ \times _))$. In this paper we present two general methods for showing the soundness of this principle. The first applies to *approximable applicative structures* – many CPO λ-models in the literature and the corresponding quotient models of indexed terms turn out to be approximable applicative structures. The second method is based on Howe's congruence candidates, which applies to many observational equivalences.
Structures satisfying cartesian coinduction are precisely those applicative structures that can be modelled using the standard *set-theoretic* application in non-wellfounded theories of sets, where the Foundation Axiom is replaced by the Axiom X_1 of Forti and Honsell.

1. Introduction

Brute force induction on the number of computation steps is a rather complex and opaque way of reasoning about the operational behaviour of programs. As it is well known, proof principles that allow us to factor out, or modularize, such inductive arguments are therefore extremely valuable. Recently, much attention has been devoted to the possibility of characterizing observational equivalences of programs as maximal fixed points of suitable operators, thus obtaining *coinduction proof principles* for reasoning on program equivalence – see, for example, Abramsky and Ong (1993), Egidi *et al.* (1992), Fiore (1996),

† Work supported by Esprit Working Group 'Types' 21900 and MURST'97 Cofin. 'Tecniche Formali...' grant.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Gordon (1995), Honsell and Lenisa (1995), Howe (1996), Lenisa (1996), Mason *et al.* (to appear), Pitts (1996a), Rutten and de Vink (1997), Lenisa (1997a), Lenisa (1997b) and Lenisa (1998). As far as functional languages or λ-calculi are concerned, however, almost all coinduction principles considered so far in the literature have had the same *applicative* pattern, which is based on Abramsky's notion of *applicative bisimulation*. All principles, apart from that given in Honsell and Lenisa (1995, Section 3), arise from monotone operators of the same shape, along the lines of Abramsky and Ong (1993) and Egidi *et al.* (1992), and exploit the fact that the observational behaviour of functional programs essentially only needs to be tested in *applicative contexts* (Lenisa 1998).

In Honsell and Lenisa (1995, Section 3), the authors considered for the first time another kind of monotone operator, thereby introducing a new form of *coinduction principle* for establishing observational equivalences between terms of λ-calculi, which they proved to be sound for the *call-by-value* λ-calculus. For reasons that will become clear in a few paragraphs, in this paper, we shall call this principle *cartesian* coinduction, and we shall use the term *cartesian* for both those applicative structures and those λ-theories for which it is sound. In Lenisa (1998), cartesian coinduction was also proved sound for the *lazy* λ-calculus.

A first objective of this paper is to show that the 'unorthodox' move of Honsell and Lenisa (1995) can be widely generalized. We present, in fact, plenty of *new* meaningful monotone operators, besides the traditional 'applicative' one, to be used for manufacturing coinduction principles for applicative structures.

The main purpose of this paper, however, is to investigate the general status and theory of the operator introduced in Honsell and Lenisa (1995), which has remained largely unexplored since then, and to answer some of the many open problems raised there. In particular, we will show that there are rich classes of cartesian applicative structures, and plenty of cartesian λ-theories.

Using concepts from *final semantics* (Aczel 1988; Rutten and Turi 1993; Rutten 1996; Turi 1996; Lenisa 1998), one can easily express the difference between the traditional *applicative* coinduction and *cartesian* coinduction. Both kinds of coinduction correspond to the *categorical coinduction principles* that arise when the set of closed λ-terms, Λ^0, (or more in general an applicative structure), is endowed with a *coalgebra* structure for a suitable functor. Applicative coinduction principles for λ-theories are obtained when the functor $F(_) = (\Lambda^0 \to _) \oplus (\Lambda^0 \to _)$ is used[†]. This approach is investigated in full generality in Honsell and Lenisa (1995). Cartesian coinduction, on the contrary, arises when the functor $G(_) = \mathscr{P}(_ \times _) \oplus \mathscr{P}(_ \times _)$ is used. In both cases the direct sum of two copies of the same structure is taken so as to distinguish between *observable values* and *non-values*. Notice how the functor F enforces the view by which elements are functions defined over the set of closed λ- terms. Also, the functor G supports the view of objects as functions, but this time functions are represented as their *cartesian graphs*, whence the name. The nature of the functor G is more general, it does not depend explicitly on any applicative structure specified in advance as F does, namely Λ^0. This allows for a more uniform treatment.

[†] $A \oplus B$ denotes the *disjoint union* of A and B, that is, the set $\{a \cup \{u\} \mid a \in A\} \cup \{b \cup \{v\} \mid b \in B\}$, for suitable elements $u, v \notin \bigcup A \cup \bigcup B$.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

We shall discuss two techniques for showing cartesianity of structures. The first is semantical and applies to ordered λ-models, such as CPO's or quotients of interiors of CPO-models. The second technique is syntactical and uses the generalizations of Howe's technique (Howe 1989) of 'congruence candidates' as carried out in Lenisa (1998). This latter technique applies to term models of λ-theories determined by observing termination under various reduction strategies.

The existence of cartesian applicative structures is closely related to the existence of *set-theoretical* applicative structures in non-wellfounded theories of sets, where the Foundation Axiom is replaced by the Antifoundation Axiom X_1 of Forti and Honsell (1983). Set-theoretical applicative structures are applicative structures whose points contain the set-theoretic description of their functional behaviour, so that application can be rendered by the usual set- theoretic application, that is, for any given points d, d', e in the model, $d \bullet d' = e$ if and only if $(d', e) \in d$. The results in this paper allow us to show that there are plenty of 'well-behaved' set-theoretical applicative structures.

Finally, we would like to point out that this paper can also be viewed as yet another chapter in the general programme of investigating the denotational semantics of λ-calculi, some of whose earlier chapters are Barendregt (1984), Coppo *et al.* (1987), Egidi *et al.* (1992), Honsell and Ronchi (1992), Abramsky and Ong (1993), Honsell and Lenisa (1993), Honsell and Lenisa (1995), and Honsell and Lenisa (1998).

The paper is organized as follows. In Section 2 we discuss, in general, the problem of coinductively characterizing congruences over applicative structures. In particular, we present a number of monotone operators that can be utilized in coinduction principles. We conclude the section by introducing special classes of 'enriched' applicative structures. In Section 3 we introduce the basic ideas of final semantics and give the categorical accounts of the coinduction principles introduced in the previous section. In Section 4 we discuss λ-congruences and λ-models. In particular, we present the six λ-theories that have been most extensively studied in the literature, and which we shall deal with explicitly. In Section 5 we present the main theorems of this paper concerning the existence of cartesian applicative structures. In Section 6 we describe a syntactical technique based on the notion of congruence candidate for establishing cartesianity of λ-theories. In Section 7 we discuss set-theoretical applicative structures in non-wellfounded Set Theories, and their connection with cartesian structures. In Section 8 we give concluding remarks and describe some open problems. In Appendix A we give information about non-wellfounded sets and in Appendix B we recall some facts about final coalgebras.

We assume that the reader is familiar with basic concepts and results in λ-calculus, final semantics, and non-wellfounded Set Theory – the reader may consult Barendregt (1984), Rutten and Turi (1993) and Lenisa (1998), and Forti and Honsell (1983), respectively, for more details.

2. The 'coinductive characterization' problem

The basic notions we shall be concerned with in this paper are those of *applicative structure*, and of *congruence* over an applicative structure:

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Definition 2.1. (Applicative structure) An *applicative structure* $\mathscr{D}$ is a pair $(D, \bullet_D)$, where $\bullet_D : D \times D \to D$; $\bullet_D$ is called *application* – it is often written infix.

Definition 2.2. Let $\mathscr{D}$ be an applicative structure. An equivalence relation $\approx \subseteq D \times D$ is a *congruence* if

$$\forall d_1, d_2, e_1, e_2 \in D.\ (d_1 \approx d_2 \ \wedge\ e_1 \approx e_2 \ \Rightarrow\ d_1 \bullet_D e_1 \approx d_2 \bullet_D e_2).$$

Notation. Let $\mathscr{D}$ be an applicative structure. For $d, e \in D$, we shall often write de, instead of $d \bullet_D e$. We shall denote a, possibly empty, sequence of elements $d_1 \ldots d_n \in D^n$, for $n \geqslant 0$, by $\vec{d}$. Moreover, we shall abbreviate $(\ldots(d \bullet_D d_1)\ldots) \bullet_D d_n$, for $n \geqslant 0$, by $d\vec{d}$.

The most obvious, and finest, congruence over an applicative structure $\mathscr{D}$ is *equality*, $=_D$. Many interesting applicative structures arising in Computer Science, however, are (observational) quotients of syntactical objects. In this case it is convenient to focus on the congruence inducing the quotient.

Given a congruence $\approx_D$ over an applicative structure $\mathscr{D}$, it is natural for us to ask for logical characterizations of it. Such characterizations will be the more useful when they induce proof principles for proving term congruence. By far, one of the most important examples of such characterizations is the one that describes $\approx_D$ as the *maximal fixed point* of a *monotone operator* on relations. As is well known, this immediately induces a *coinduction principle* for establishing that two points are $\approx_D$-congruent.

Definition 2.3. (Ξ-coinduction principle) Let $\mathscr{D}$ be an applicative structure, and let $\Xi : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ be a monotone operator with respect to $\subseteq$. Then

(i) $\mathscr{R} \in \mathscr{P}(D \times D)$ is a Ξ- *bisimulation* if $\mathscr{R} \subseteq \Xi(\mathscr{R})$;

(ii) The Ξ-*coinduction principle* is the following rule:

$$\frac{(d, e) \in \mathscr{R} \quad \mathscr{R} \subseteq \Xi(\mathscr{R})}{d \approx^{\Xi} e}\ ,$$

where $\approx^{\Xi}$ denotes the *maximal fixed point* of Ξ.

The 'coinductive characterization problem' for a congruence $\approx_D$, over an applicative structure $\mathscr{D}$, is the problem of finding appropriate operators whose maximal fixed point is $\approx_D$. The solution to this problem is not immediate. For instance, the natural operator for which congruences are fixed points is $\Theta_D : \mathscr{P}(D \times D) \to \mathscr{P}(D \times D)$, where $\Theta_D(\mathscr{R}) = \{(d, e) \mid \forall f, g.\ f \mathscr{R} g \Rightarrow df \mathscr{R} eg\}$. But Θ_D is not monotone, thus, in order to reason coinductively on applicative structures, alternative operators to it have to be found. Following the seminal work of Abramsky (Abramsky 1989; Abramsky and Ong 1993), all investigations on applicative structures (except for Honsell and Lenisa (1995) and Lenisa (1998)) have focussed mainly on characterizing congruences as maximal fixed points of the class of monotone operators given by the following definition.

Definition 2.4. Let $\mathscr{D}$ be an applicative structure and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation. The operator $\Phi_{\mathsf{Eq}_D} : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ is defined as follows:

$$\Phi_{\mathsf{Eq}_D}(\mathscr{R}) = \{(d, e) \mid d\, \mathsf{Eq}_D\, e \ \wedge\ \forall f \in D.\ df\ \mathscr{R}\ ef\}\ .$$

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

It is easy to show that the maximal fixed point of the operator Φ_{Eq_D}, namely $\approx^{\Phi_{\mathsf{Eq}_D}}$, is the applicative relation $\approx^{app}_{\mathsf{Eq}_D}$ given by the following definition.

Definition 2.5. (Applicative relation) Let $\mathscr{D}$ be an applicative structure and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation. The *applicative relation* $\approx^{app}_{\mathsf{Eq}_D} \subseteq D \times D$ is defined by

$$d \approx^{app}_{\mathsf{Eq}_D} e \iff \forall \vec{f}.\ d\vec{f}\ \mathsf{Eq}_D\ e\vec{f}.$$

Many examples of coinductive characterizations of congruences, in terms of the operators Φ_{Eq_D} above, over both syntactical and semantical structures have been extensively discussed in Egidi *et al.* (1992), Abramsky and Ong (1993), Howe (1996), Lenisa (1997a), Lenisa (1997b) and Pitts (1996a).

Besides the Φ_{Eq_D}'s, one can consider many other operators, which, in our opinion, yield interesting characterizations and are worthy of being investigated. The most intriguing example is that of Ψ_{Eq_D}, the operator introduced in Honsell and Lenisa (1995). This operator arises when we capitalize on the fact that the functional behaviour of elements in applicative structures can be described by a *cartesian graph*, that is, a subcollection of the cartesian product $D \times D$.

Definition 2.6. Let $\mathscr{D}$ be an applicative structure and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation. The operator $\Psi_{\mathsf{Eq}_D} : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ is defined as follows:

$$\Psi_{\mathsf{Eq}_D}(\mathscr{R}) = \{(d,e) \mid d\ \mathsf{Eq}_D\ e\ \wedge \begin{pmatrix} \forall f \in D\ \exists g \in D.\ f\ \mathscr{R}\ g & \wedge & df\ \mathscr{R}\ eg \\ \wedge & & \\ \forall f \in D\ \exists g \in D.\ f\ \mathscr{R}\ g & \wedge & ef\ \mathscr{R}\ dg \end{pmatrix}\} .$$

We use $\approx^{\Psi_{\mathsf{Eq}_D}}$ to denote the greatest Ψ_{Eq_D}-bisimulation.

Variants of the above operators can be obtained when we consider that, in view of currying, points of applicative structures also exhibit the behaviour of n-ary functions.

Definition 2.7. Let $\mathscr{D}$ be an applicative structure, let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation, and let $n > 0$.

(i) Let $\Phi^n_{\mathsf{Eq}_D} : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ be the operator defined as follows:

$$\Phi^n_{\mathsf{Eq}_D}(\mathscr{R}) = \{(d,e) \mid d\,\mathsf{Eq}_D\,e\ \wedge\ \forall \vec{f} \in D^n.\ d\vec{f}\ \mathscr{R}\ e\vec{f}\}\ .$$

(ii) Let $\Psi^n_{\mathsf{Eq}_D} : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ be the operator defined as follows:

$$\Psi^n_{\mathsf{Eq}_D}(\mathscr{R}) = \{(d,e) \mid d\ \mathsf{Eq}_D\ e\ \wedge \begin{pmatrix} \forall \vec{f} \in D^n\ \exists \vec{g} \in D^n.\ \forall i. f_i\ \mathscr{R}\ g_i\ \wedge\ d\vec{f}\ \mathscr{R}\ e\vec{g} \\ \wedge \\ \forall \vec{f} \in D^n\ \exists \vec{g} \in D^n.\ \forall i. f_i\ \mathscr{R}\ g_i\ \wedge\ e\vec{f}\ \mathscr{R}\ d\vec{g} \end{pmatrix}\} .$$

More operators arise if we take the existential quantification over n in the above definition.

We can also combine Φ's and Ψ's, as in the following definition.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Definition 2.8. Let $\mathscr{D}$ be an applicative structure and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation. Let $Z_{\mathsf{Eq}_D} : \mathscr{P}(D \times D) \longrightarrow \mathscr{P}(D \times D)$ be the operator defined as follows:

$$Z_{\mathsf{Eq}_D}(\mathscr{R}) = \{(d,e) \mid d\ \mathsf{Eq}_D\ e\ \wedge \left(\begin{array}{l} \forall f \in D\ \exists g \in D.\ f\ \mathscr{R}\ g\ \wedge\ df \approx^{app}_{\mathsf{Eq}_D} eg \\ \wedge \\ \forall f \in D\ \exists g \in D.\ f\ \mathscr{R}\ g\ \wedge\ ef \approx^{app}_{\mathsf{Eq}_D} dg \end{array} \right)\} .$$

We shall not present other operators, but simply point out that using the above templates, one can easily devise further combined operators.

Notice that all the operators introduced above are monotone, hence each of them induces a coinduction principle.

For simplicity, we shall refer to Φ_{Eq_D}-coinduction as Eq_D-*applicative coinduction*. For the reasons we mentioned earlier, and which will become clearer in Section 3, we shall refer to Ψ_{Eq_D}-coinduction as Eq_D-*cartesian coinduction*.

The above coinduction principles can be used for reasoning on the kinds of congruences on applicative structures given by the following definition.

Definition 2.9. (Ξ_{Eq_D}-coinductive congruence) Let $\approx_D$ be a congruence on an applicative structure $\mathscr{D}$, and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation. Then $\approx_D$ is a Ξ_{Eq_D}-*coinductive congruence*, or, equivalently, $\approx_D$ *satisfies* Ξ_{Eq_D}-*coinduction*, if $\approx_D$ coincides with $\approx^{\Xi_{\mathsf{Eq}_D}}$.

Again, for simplicitly, we shall refer to Ψ_{Eq_D}-coinductive congruences as Eq_D-*cartesian congruences*. Rather than saying that $=_D$ is Ξ_{Eq_D}-coinductive, we shall simply say that the applicative structure $\mathscr{D}$ is Ξ_{Eq_D}-coinductive.

An elementary result concerning the Ξ_{Eq_D}-coinduction principles for $\Xi = \Phi, \Psi, \Phi^n, \Psi^n$, Z is given by the following proposition.

Proposition 2.10. Let $(D, \bullet_D)$ be an applicative structure, $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation, and $\mathscr{R} \subseteq D \times D$ be a reflexive relation:

— if $\mathscr{R}$ is a Φ_{Eq_D}-bisimulation, it is also a Z_{Eq_D}-bisimulation;
— if $\mathscr{R}$ is a Φ_{Eq_D}-bisimulation, it is also a Ψ_{Eq_D}-bisimulation;
— if $\mathscr{R}$ is a Z_{Eq_D}-bisimulation, $(R \cup \approx^{\Psi_{\mathsf{Eq}_D}})$ is also a Ψ_{Eq_D}-bisimulation;
— if n divides m and $\mathscr{R}$ is a $\Phi^n_{\mathsf{Eq}_D}$- bisimulation, it is also a $\Phi^m_{\mathsf{Eq}_D}$-bisimulation;
— if n divides m and $\mathscr{R}$ is a $\Psi^n_{\mathsf{Eq}_D}$- bisimulation, it is also a $\Psi^m_{\mathsf{Eq}_D}$-bisimulation.

It is surprisingly difficult to extend results for one of the above operators to the others. This paper is devoted essentially to showing that large classes of applicative structures satisfying the Eq_D-applicative coinduction principle also satisfy the Eq_D-cartesian coinduction principle.

2.1. *Enriched applicative structures*

In this paper we shall be mainly concerned with applicative structures that have some 'extra' structure. In particular, we shall consider 'order enriched' and 'approximable' applicative structures. First we introduce ordered applicative structures.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Definition 2.11. (Ordered applicative structure) An *ordered applicative structure*, $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D)$, is a triple such that

1 $(D, \sqsubseteq_D)$ is a non-trivial partial order;
2 $\bullet_D : D \times D \to D$ is continuous, that is, for all $\{d_n\}_{n\in\omega}$, $\{e_n\}_{n\in\omega}$ ω-chains, if there exist $\bigsqcup_n d_n$ and $\bigsqcup_n e_n$, then $\bigsqcup_n d_n \bullet_D \bigsqcup_n e_n =_D \bigsqcup_n d_n \bullet_D e_n$.

In dealing with ordered applicative structures, it is natural to ask for coinductive characterizations of the order relation itself, rather than just $=_D$. We now give an important definition.

Definition 2.12. (Φ_{pEq_D}-coinductive applicative structure) Let $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D)$ be an ordered applicative structure, and let $\mathsf{pEq}_D \subseteq D \times D$ be a pre-equivalence relation. The structure $\mathscr{D}$ is pEq_D-*coinductive* if the order relation $\sqsubseteq_D$ satisfies the following pEq_D-applicative coinduction principle:

$$d \sqsubseteq_D e \iff \exists \mathscr{R} \subseteq D \times D.\ (\mathscr{R} \subseteq \Phi_{\mathsf{pEq}_D}(\mathscr{R}) \ \wedge\ d \mathscr{R} e)\ ,$$

where
$\Phi_{\mathsf{pEq}_D} : \mathscr{P}(D \times D) \to \mathscr{P}(D \times D)$
$\Phi_{\mathsf{pEq}_D}(\mathscr{R}) = \{(d, e) \mid d\, \mathsf{pEq}_D\, e \ \wedge\ \forall f \in D.\ df\ \mathscr{R}\ ef\}$.

Coinductive characterizations of inequality appear to be especially useful in the case of the operator Φ_{Eq_D}, since we have that $\approx^{app}_{\mathsf{Eq}_D} = \sqsubseteq^{app}_{\mathsf{pEq}_D} \cap (\sqsubseteq^{app}_{\mathsf{pEq}_D})^{-1}$, where $\mathsf{Eq}_D = \mathsf{pEq}_D \cap (\mathsf{pEq}_D)^{-1}$ and $\sqsubseteq^{app}_{\mathsf{pEq}_D}$ is the maximal fixed point of Φ_{pEq_D}. For many other operators, and notably for Ψ_{Eq_D}, such a neat correspondence between preorder and equality does not arise. In fact, it is not the case in general that $\approx^{\Psi_{\mathsf{Eq}_D}} = \sqsubseteq^{\Psi_{\mathsf{pEq}_D}} \cap (\sqsubseteq^{\Psi_{\mathsf{pEq}_D}})^{-1}$, where $\sqsubseteq^{\Psi_{\mathsf{pEq}_D}}$ denotes the maximal fixpoint of the operator $\Psi_{\mathsf{pEq}_D} : \mathscr{P}(D \times D) \to \mathscr{P}(D \times D)$ defined by

$$\Psi_{\mathsf{pEq}_D}(\mathscr{R}) = \{(d, e) \mid d\, \mathsf{pEq}_D\, e \ \wedge\ \forall f \in D. \exists g \in D.\ (f \mathscr{R} g \ \wedge\ df \mathscr{R} eg)\}\ .$$

We conclude this subsection by introducing another important class of enriched applicative structures: the *approximable applicative structures* (aas). These are ordered applicative structures $(D, \sqsubseteq_D, \bullet_D)$, together with a countable system of elements of D, $\{p_n\}_{n\in\omega}$, representing projection functions, which approximate the elements of D. Clearly all 'inverse limit' λ-models are aas's.

Definition 2.13. (Approximable applicative structure) An *approximable applicative structure (aas)* is a structure $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ such that

1 $(D, \sqsubseteq_D, \bullet_D)$ is an ordered applicative structure;
2 $\{p_n\}_{n\in\omega}$ is a countable set of elements of D whose applicative behaviour is that of a complete system of *projection* functions $\{\pi_n : D \to D\}_{n\in\omega}$, that is, $\forall n \geqslant 0.\ \forall d \in D.\ p_n d =_D \pi_n(d)$, such that
   — $\forall n, m \geqslant 0.\ p_m(p_n d) =_D p_{min\{m,n\}} d$;
   — $\forall n > 0.\ \forall d, e \in D.\ (p_n d)e =_D p_{n-1}(d(p_{n-1}e))$;
   — $\forall d \in D.\ d =_D \bigsqcup_n p_n d$;
   — $\exists n, m \geqslant 0.\ \exists d, e \in D.\ n \neq m \ \wedge\ (p_n d)e \neq (p_m d)e$.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

The latter condition in the above definition may seem odd. If we drop it, then, trivially, all applicative structures can be turned into aas's, by taking as projections any set of elements $\{p_n\}_{n\in\omega}$ satisfying $\forall d,e.\ (p_n d)e =_D de$. In fact, one can see that: either all projections satisfy this latter condition, or the last condition in the definition above is met.

Notation. When clear from the context, for $d \in D$, we will use d_n to denote $p_n \bullet_D d$, and D_n to denote the set $\{p_n \bullet_D d \mid d \in D\}$.

Correspondingly, we also introduce the following notion.

Definition 2.14. (Φ_{pEq_D}-coinductive approximable applicative structure)
A Φ_{pEq_D}-*coinductive approximable applicative structure (*pEq_D-*caas)* is an approximable applicative structure $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ such that the ordered applicative structure $(D, \sqsubseteq_D, \bullet_D)$ is a Φ_{pEq_D}-coinductive applicative structure.

3. The final perspective

Following the Final Semantics Paradigm, introduced in Aczel (1988), and further developed in Aczel and Mendler (1989), Rutten and Turi (1993), Rutten (1996), Turi (1996) and Lenisa (1998), coinduction principles can be given a very neat *categorical explanation*. It is interesting to note that many of the monotone operators described in Section 2 were actually suggested by this categorical analysis.

We work in Set^*, the category whose objects are sets of a universe of the *non- wellfounded* set theory $\mathsf{ZF}^\circ\mathsf{X}_1$, and whose morphisms are the set theoretic functions. Set-theoretic and categorical concepts are defined in Appendices A and B. The general pattern of the Final Semantics justification of a coinduction principle over an applicative structure $\mathscr{D}$, induced by the monotone operator Ξ, is as follows. We endow $\mathscr{D}$ with the structure of a H_Ξ-*coalgebra*, (D, α_Ξ), for a suitable endofunctor H_Ξ (see Theorem 3.2 below), that preserves weak pullbacks and has *final* coalgebra. Then the *unique* mapping into the final coalgebra induces an equivalence on $\mathscr{D}$ that is the union of all *categorical* H_Ξ-*bisimulations*. Full definitions appear in Appendix B. If the functor H_Ξ and the H_Ξ- coalgebra structure have been given appropriately, we have the following crucial theorem.

Theorem 3.1. $\mathscr{R}$ is a categorical H_Ξ-bisimulation on the coalgebra (D, α_Ξ) if and only if $\mathscr{R}$ is a Ξ-bisimulation.

In Honsell and Lenisa (1995), the authors succeeded in providing a final justification of the *applicative coinduction* principle (Φ_{Eq_D}-coinduction principle) for various λ-theories over the applicative structure consisting of the set of closed λ-terms.

Generalizing Honsell and Lenisa (1995), we can establish the following correspondence between monotone operators and functors in Set^*.

Theorem 3.2. Let $\mathscr{D} = (D, \bullet_D)$ be an applicative structure and let $\mathsf{Eq}_D \subseteq D \times D$ be an equivalence relation on D such that $|D/\mathsf{Eq}_D| \leqslant \kappa$, that is, the cardinality of the set of equivalence classes of Eq_D is less than or equal to κ.

Let Ξ be one of the operators $\{\Phi_{\mathsf{Eq}_D}, \Psi_{\mathsf{Eq}_D}, \Phi^n_{\mathsf{Eq}_D}, \Psi^n_{\mathsf{Eq}_D}, Z_{\mathsf{Eq}_D}\}$. Ξ-bisimulations are

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

categorical H_Ξ-bisimulations on the H_Ξ-coalgebra (D, α_Ξ) for the endofunctor H_κ^Ξ in Set^* given by the following table:

$$\begin{array}{lll}
\Xi \equiv \Phi_{\mathsf{Eq}_D} & : & H_\kappa^\Xi(X) = \bigoplus_\kappa (D \to X) \\
& & H_\kappa^\Xi(f) = \bigoplus_\kappa (id_D \to f) \\
& & \alpha_\Xi(d) = \{i_{[d]}\} \cup \lambda e \in D.\ de, \\
& & \text{where } i_{[d]} \text{ is the tag of the } \mathsf{Eq}_D\text{-equivalence class of } d.
\end{array}$$

$$\begin{array}{lll}
\Xi \equiv \Psi_{\mathsf{Eq}_D} & : & H_\kappa^\Xi(X) = \bigoplus_\kappa \mathscr{P}_{<\aleph_1}(X \times X) \\
& & H_\kappa^\Xi(f) = \bigoplus_\kappa (f \times f)^+ \\
& & \alpha_\Xi(d) = \{i_{[d]}\} \cup \lambda e \in D.\ de, \\
& & \text{where } i_{[d]} \text{ is the tag of the } \mathsf{Eq}_D\text{-equivalence class of } d, \text{ and,} \\
& & \text{for } f : X \to Y,\ f^+ : \mathscr{P}_{<\aleph_1}(X) \to \mathscr{P}_{<\aleph_1}(Y) \text{ is defined by} \\
& & \forall x \in \mathscr{P}_{<\aleph_1}(X).\ f^+(x) = \{f(z) \mid z \in x\}.
\end{array}$$

$$\begin{array}{lll}
\Xi \equiv \Phi^n_{\mathsf{Eq}_D} & : & H_\kappa^\Xi(X) = \bigoplus_k (D^n \to X), \\
& & H_\kappa^\Xi(f) = \bigoplus_k (id_D^n \to f) \\
& & \alpha_\Xi(d) = \{i_{[d]}\} \cup \lambda e_1 \ldots e_n \in D.\ (\ldots(de_1)\ldots e_n), \\
& & \text{where } i_{[d]} \text{ is the tag of the } \mathsf{Eq}_D\text{-equivalence class of } d.
\end{array}$$

$$\begin{array}{lll}
\Xi \equiv \Psi^n_{\mathsf{Eq}_D} & : & H_\kappa^\Xi(X) = \bigoplus_\kappa \mathscr{P}_{<\aleph_1}(X^n \times X) \\
& & H_\kappa^\Xi(f) = \bigoplus_\kappa (f^n \times f)^+ \\
& & \alpha_\Xi(d) = \{i_{[d]}\} \cup \lambda e_1 \ldots e_n \in D.\ (\ldots(de_1)\ldots e_n), \\
& & \text{where } i_{[d]} \text{ is the tag of the } \mathsf{Eq}_D\text{-equivalence class of } d.
\end{array}$$

$$\begin{array}{lll}
\Xi \equiv Z_{\mathsf{Eq}_D} & : & H_\kappa^\Xi(X) = \bigoplus_\kappa \mathscr{P}_{<\aleph_1}(X \times (D/\approx^{app}_{\mathsf{Eq}_D})) \\
& & H_\kappa^\Xi(f) = \bigoplus_\kappa (f \times id_{(D/\approx^{app}_{\mathsf{Eq}_D})})^+ \\
& & \alpha_\Xi(d) = \{i_{[d]}\} \cup \lambda e \in D.\ [de]_{\approx^{app}_{\mathsf{Eq}_D}}, \\
& & \text{where } i_{[d]} \text{ is the tag of the } \mathsf{Eq}_D\text{-equivalence class of } d \text{ and} \\
& & [de]_{\approx^{app}_{\mathsf{Eq}_D}} \text{ is the equivalence class of } de \text{ modulo } \approx^{app}_{\mathsf{Eq}_D}.
\end{array}$$

For A, B sets, the notation $\bigoplus_{|A|} B$ stands for the disjoint sum of $|A|$ copies of B, that is, $\{\{\alpha\} \cup b \mid \alpha \in |A| \ \wedge\ b \in B\}$, assuming $|A| \cap TC(B) = \varnothing$. The proof of the above theorem is standard, see Aczel (1988), Rutten and Turi (1993), and Lenisa (1998).

It is interesting to note that, in each case, the structure of the functor corresponding to the monotone operator reflects the way in which we construe objects of an applicative structure: as a function on a constant set, as the graph of a self-function, as an n-ary function, as the graph of an n-ary self-function, or as the graph of a function that takes values over a suitable constant set. In each case, however, we need $|D/\mathsf{Eq}_D|$ copies of the same domain, one for each equivalence class of the equivalence Eq_D we use as start-up.

The categorical account also allows us to give a clear characterization of applicative structures that satisfy Ξ-coinduction principles. In particular, for

$$\Xi \in \{\Phi_{\mathsf{Eq}_D}, \Psi_{\mathsf{Eq}_D}, \Phi^n_{\mathsf{Eq}_D}, \Psi^n_{\mathsf{Eq}_D}, Z_{\mathsf{Eq}_D}\},$$

we have the following result.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Corollary 3.3. Let $\mathscr{D}$ be an applicative structure, such that $|D/\mathsf{Eq}_D| \leqslant \kappa$. Then the H_κ^Ξ-coalgebra (D, α_Ξ) is strongly extensional if and only if $\mathscr{D}$ is a Ξ-coinductive applicative structure.

4. Theories and models of the λ-calculus

In this section we discuss λ-theories, the very important class of congruence relations over the paradigm applicative structure consisting of closed λ-terms.

We start by giving basic definitions and general results on λ-theories and λ-models. In Subsection 4.1 we present the specific λ-theories we shall be mainly concerned with.

Let Λ^C denote the class of λ-terms built over the set of basic constants C, that is,

$$(\Lambda\ni)\ M\ ::=\ x\ \mid\ c\ \mid\ MM\ \mid\ \lambda x.M\ ,$$

where $x \in Var$, $c \in C$.
Let $(\Lambda^C)^0$ denote the class of *closed* λ- terms built over the set of basic constants C.

Definition 4.1.

— A *λ-pre-equivalence* is a reflexive and transitive relation on $\Lambda^C \times \Lambda^C$.
— A *λ-equivalence* is a symmetric λ-pre-equivalence.
— A *λ-precongruence*, $\leqslant$, is a λ-pre-equivalence that is a congruence with respect to application and λ-abstraction, that is, for all $M, N, M', N' \in \Lambda^C$,

$$M \leqslant N\ \wedge\ M' \leqslant N'\ \Rightarrow\ MM' \leqslant NN'\ ,\ \text{and}$$

$$M \leqslant N\ \Rightarrow\ \lambda x.M \leqslant \lambda x.N\ .$$

— A *λ-congruence* is a λ-equivalence which is a congruence.
— A *λ-theory* is the restriction of a λ-congruence to $(\Lambda^C)^0$.

We focus on λ-(pre)congruences that arise from relations $\mathsf{Eq} \subseteq (\Lambda^C)^0 \times (\Lambda^C)^0$ in the following sense.

Definition 4.2. Let $\mathsf{Eq} \subseteq (\Lambda^C)^0 \times (\Lambda^C)^0$. The *contextual relation* $\approx^{\mathsf{Eq}} \subseteq \Lambda^C \times \Lambda^C$ is defined as follows:

$$M \approx^{\mathsf{Eq}} N\ \iff\ \forall C[\,].\ (C[M], C[N] \in (\Lambda^C)^0 \Rightarrow C[M]\ \mathsf{Eq}\ C[N])\ .$$

Notice that, if Eq is a (pre)equivalence, then the relation $\approx^{\mathsf{Eq}}$ is a (pre)congruence.

All λ-congruences are induced by an equivalence relation with just *two* equivalence classes, as given by the following proposition.

Proposition 4.3. Any λ-congruence $\approx \subseteq \Lambda^C \times \Lambda^C$ is induced by a suitable set $\mathscr{V} \subseteq (\Lambda^C)^0$ in the following sense:

$$M \approx N\ \iff\ \forall C[\,].\ (C[M], C[N] \in (\Lambda^C)^0 \Longrightarrow (C[M] \in \mathscr{V} \iff C[N] \in \mathscr{V}))\ .$$

Proof. Just take $\mathscr{V} = \{PQ \mid P \approx Q\}$. □

Question 4.4. Is there an analogue of Proposition 4.3 for λ-precongruences?

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

The relationship between λ-theories and β-reduction is formalized and clarified by the following proposition, whose proof is straightforward.

Proposition 4.5. Let $\approx^{\mathsf{Eq}}$ be the congruence induced by the equivalence relation $\mathsf{Eq} \subseteq (\Lambda^C)^0 \times (\Lambda^C)^0$. Then the notion of β-reduction $\rightarrow_{\beta_r}$[†] is correct with respect to $\approx^{\mathsf{Eq}}$, that is, $=_{\beta_r} \subseteq \approx^{\mathsf{Eq}}$, if and only if Eq is closed under β_r-conversion, that is, $(=_{\beta_r} \cap ((\Lambda^C)^0 \times (\Lambda^C)^0)) \subseteq \mathsf{Eq}$.

Finally we recall some useful semantical definitions.

Definition 4.6. (Ordered λ-model) An *ordered λ-model* is a quadruple $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, [\![\ ]\!]^D)$, where

1 $(D, \sqsubseteq_D, \bullet_D)$ is an ordered applicative structure;
2 $[\![\ ]\!]^D : \Lambda \times Env \to D$, for $Env = [Var \to D_{var}]$,

$$D_{var} = \begin{cases} D \setminus \{\bot\} & \text{if } \bot \in D \text{ is the least element of } D \text{ and } \forall d \in D.\ d \bullet_D \bot = \bot, \\ D & \text{otherwise,} \end{cases}$$

is the *interpretation function*;
3 $[\![x]\!]^D_\rho = \rho(x)$;
4 $[\![c]\!]^D_\rho = d_c$, for some $d_c \in D$;
5 $[\![MN]\!]^D_\rho = [\![M]\!]^D_\rho \bullet_D [\![N]\!]^D_\rho$;
6 $\forall d \in D_{var}.\ [\![\lambda x.M]\!]^D_\rho \bullet_D d = [\![M]\!]^D_{\rho[d/x]}$;
7 $\forall d \in D_{var}.\ [\![M]\!]^D_{\rho[d/x]} = [\![N]\!]^D_{\rho[d/x]} \Rightarrow [\![\lambda x.M]\!]^D_\rho = [\![\lambda x.N]\!]^D_\rho$;
8 $\forall \rho, \rho' : Var \to D_{var}\ (\forall x \in FV(M).\ \rho(x) = \rho'(x) \Rightarrow [\![M]\!]^D_\rho = [\![M]\!]^D_{\rho'})$.

One can see immediately that, given a λ-model $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, [\![\ ]\!]^D)$, the theory $\mathscr{T}^{\mathscr{D}}$ induced by it, that is, the set of pairs of terms that have the same interpretation in a λ-model, is a λ-congruence. Correspondingly, the set of pairs of terms whose first component has an interpretation that is less than that of the second component, is a λ-precongruence. A λ-model $(D, \sqsubseteq_D, \bullet^D, [\![\ ]\!])$ is *computationally adequate* with respect to a λ-precongruence $\leqslant$ if

$$\forall M, N \in \Lambda^0.\ [\![M]\!]^D \sqsubseteq_D [\![N]\!]^D \implies M \leqslant N.$$

A λ-model is *fully abstract* if the above implication is a bi-implication.

In this paper we shall be concerned with two kinds of λ-models: *finitary λ-models* and *term models*.

As far as finitary λ-models we shall focus on those given by the following definition.

Definition 4.7. (λ-aas, λ-caas)

— An aas $(D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ is a *λ-aas* if it is a λ-model.
— A structure $(D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ is a $\Phi_{\mathsf{pEq_D}}\lambda$*-caas* if it is a $\Phi_{\mathsf{pEq_D}}$-caas which is a λ-model.

[†] A notion of *β-reduction*, $\rightarrow_{\beta_r}$, is the λ-precongruence generated by a set of pairs (redex, β-reduct). The λ-congruence generated by the symmetric closure of a β-reduction $\rightarrow_{\beta_r}$ is the $=_{\beta_r}$*-conversion*.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Many CPO λ-models in the literature turn out to be λ-aas. Other interesting examples of λ-aas are models obtained by quotienting sets of indexed terms (see Section 5.1 for examples).

Term models are induced by λ-precongruences and are defined as follows.

Definition 4.8. (Term model) Let $\leqslant$ be a λ-precongruence and let T be the quotient $\Lambda^0/(\leqslant \cap(\leqslant)^{-1})$, that is, the set of $(\leqslant \cap(\leqslant)^{-1})$-equivalence classes of closed terms. The term model $\mathscr{T}_{\leqslant}$ is the applicative structure $(T, \bullet)$, where, for all $M, N \in (\Lambda^C)^0$, $[M] \bullet [N] = [MN]$. The interpretation function $[\![\ ]\!] : \Lambda \times Env \to T$, for $Env = [Var \to T_{var}]$ and

$$T_{var} = \begin{cases} T \setminus \bot & \text{if } \bot \in T \text{ is the least element of } T \text{ and } \forall M.\ [M] \bullet \bot = \bot \\ T & \text{otherwise,} \end{cases}$$

is defined as follows:

$$[\![M]\!]_\rho = [\rho(M)]\ ,$$

where $FV(M) \subseteq \{x_1, \ldots, x_n\}$ and $\rho(M) = M[\rho(x_1)/x_1, \ldots, \rho(x_n)/x_n]$.

The equivalence determined by the interpretation function into $\mathscr{T}_{\leqslant}$ is $\leqslant \cap(\leqslant)^{-1}$, of course. Notice that, since we only consider *closed* terms, it is not always the case that a term model is an ordered λ-model, since Clause 7 of Definition 4.6 may fail. In any case, this will hold for all the term models we shall deal with explicitly in this paper. Our notion of term model is sligthly non-standard, so that it also accomodates 'restricted calculi' such as Plotkin's call-by-value λ_v-calculus. In these cases $\bot$ is intended to denote *non-values*. All the precongruences that we shall explicitly mention in this paper satisfy this condition.

4.1. *Examples of λ-theories*

In this paper we are mainly concerned with λ-precongruences that arise from reduction strategies. Namely, we take a term to approximate another if we cannot *observe* that, for a given *closing context*, the strategy halts successfully when one is used to fill the hole, but does not halt when the other is used. A *reduction strategy* is a procedure for determining, for each λ-term, a specific β-redex in it to contract. A (possibly non-deterministic) strategy can be formalized as a relation $\to_\sigma \subseteq \Lambda \times \Lambda\ (\Lambda^0 \times \Lambda^0)$ such that if $(M, N) \in \to_\sigma$ (also written infix as $M \to_\sigma N$), then N is a possible result of applying $\to_\sigma$ to M. We denote by $\to_\sigma^*$ the reflexive and transitive closure of $\to_\sigma$. The set of terms that do not belong to the domain of $\to_\sigma$ are partitioned into two disjoint sets: the set of *σ-values*, denoted by Val_σ, and the set of *σ-deadlocks*. Given $\to_\sigma$, we can define the *evaluation relation* $\Downarrow_\sigma \subseteq \Lambda \times \Lambda\ (\Lambda^0 \times \Lambda^0)$, such that $M \Downarrow_\sigma N$ holds if and only if there exists a (possible empty) reduction path from M to a σ-value N. If there exists N such that $M \Downarrow_\sigma N$, then $\to_\sigma$ *halts successfully* on M and M converges ($M \Downarrow_\sigma$), otherwise $\to_\sigma$ does not terminate on M, or reaches a *deadlock* from M, and M diverges ($M \Downarrow\!\!\!\!/_\sigma$). Each reduction strategy induces an *operational semantics*, in that we can imagine a machine that evaluates terms by implementing the given strategy. The *observational pre-equivalence* arises when we consider programs as *black boxes* and only observe their 'halting properties'.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Definition 4.9. Let $\rightarrow_\sigma$ be a reduction strategy and let $M, N \in \Lambda$.

— The *observational precongruence* $\leqslant_\sigma$ is defined by
$M \leqslant_\sigma N \iff \forall C[\,].(C[M], C[N] \in \Lambda^0 \Rightarrow (C[M] \Downarrow_\sigma \Rightarrow C[N] \Downarrow_\sigma))$.
— The *observational* λ-theory $\approx_\sigma$ is the congruence defined by
$M \approx_\sigma N \iff \forall C[\,].(C[M], C[N] \in \Lambda^0 \Rightarrow (C[M] \Downarrow_\sigma \Leftrightarrow C[N] \Downarrow_\sigma))$.

It is immediate to check that the σ-observational precongruence, $\leqslant_\sigma$, is the λ-precongruence induced by the pre-equivalence relation $\mathsf{pEq}_\sigma = \{(M,N) \mid M \Downarrow_\sigma \Rightarrow N \Downarrow_\sigma\}$. And, similarly, the σ-observational congruence, $\approx_\sigma$, is the λ-theory induced by the equivalence relation $\mathsf{Eq}_\sigma = \{(M,N) \in \Lambda^0 \times \Lambda^0 \mid M \Downarrow_\sigma \Leftrightarrow N \Downarrow_\sigma\}$. Notice also that $\approx_\sigma = \leqslant_\sigma \cap (\leqslant_\sigma)^{-1}$.

There is no loss of generality in considering σ-observational congruences, rather than λ-theories. In fact, by Proposition 4.3, any λ-theory can be viewed as an observational congruence induced by a trivially empty strategy, whose values are the terms in the set $\mathscr{V}$ arising in the proof of Proposition 4.3.

As we remarked earlier, λ-theories can be viewed as congruences on the applicative structure consisting of closed λ-terms. Hence we shall say that a λ-theory $\approx_\sigma$ is Ξ_{Eq_σ}-coinductive if $\approx_\sigma$ coincides with $\approx^{\Xi_{\mathsf{Eq}_\sigma}}$, that is, the maximal fixed-point of the operator Ξ_{Eq_σ}. Hence, those theories that are $\Phi_{\mathsf{Eq}_\sigma}$-coinductive or $\Psi_{\mathsf{Eq}_\sigma}$-coinductive, we shall call Eq_σ-*applicative* or Eq_σ-*cartesian*, respectively.

Now we present six examples of σ-observational precongruences, and the corresponding *computationally adequate finitary denotational* λ- *models*, that have been extensively studied in the literature (Plotkin 1977; Barendregt 1984; Coppo *et al.* 1987; Honsell and Ronchi della Rocca 1992; Egidi *et al.* 1992; Abramsky and Ong 1993; Honsell and Lenisa 1999).

4.1.1. $\leqslant_l$. The precongruence $\leqslant_l$ is induced by the *lazy call-by-name* strategy $\rightarrow_l \subseteq \Lambda^0 \times \Lambda^0$, which reduces the *leftmost* β-redex not appearing in a λ-abstraction. $Val_l = \{\lambda x.M \mid M \in \Lambda\} \cap \Lambda^0$. The evaluation $\Downarrow_l$ is the least binary relation over $\Lambda^0 \times Val_l$ satisfying the rules:

$$\frac{}{\lambda x.M \Downarrow_l \lambda x.M} \qquad \frac{M \Downarrow_l \lambda x.P \quad P[N/x] \Downarrow_l Q}{MN \Downarrow_l Q}$$

Classical β-reduction is correct with respect to $\approx_l$ (Abramsky and Ong 1993) – this is the reduction strategy of lazy functional languages.

A computationally adequate ordered λ-model for $\leqslant_l$ is the model $\mathscr{D}^l$, studied in Abramsky and Ong (1993). The model $\mathscr{D}^l$ is the quadruple $(D^l, \sqsubseteq^l, \bullet_D^l, [\![\ ]\!]^l)$, where D^l is the inverse limit initial solution of the equation $D \simeq [D \rightarrow D]_\perp$ in the category $CPO_\perp$, and $\bullet_D^l$ and $[\![\ ]\!]^l$ are defined using the canonical isomorphisms given by the inverse limit construction. Computational adequacy follows from the following theorem.

Theorem 4.10. (Computational adequacy of $\mathscr{D}^l$)

$$M \Downarrow_l \iff \forall \rho.\ [\![M]\!]^l_\rho \sqsupseteq_l \boldsymbol{\lambda} d \in D^l.\ \perp\ .$$

4.1.2. $\leqslant_v$. The precongruence $\leqslant_v$ is determined by Plotkin's *lazy call-by-value* strategy $\rightarrow_v \subseteq \Lambda^0 \times \Lambda^0$, which reduces the *leftmost* β-redex, not appearing within a λ-abstraction, whose argument is a λ-abstraction. $Val_v = \{\lambda x.M \mid M \in \Lambda\} \cap \Lambda^0$. The evaluation $\Downarrow_v$ is the

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

least binary relation over $\Lambda^0 \times Val_v$ satisfying the following rules:

$$\frac{}{\lambda x.M \Downarrow_v \lambda x.M} \qquad \frac{M \Downarrow_v \lambda x.P \quad N \Downarrow_v Q \quad P[Q/x] \Downarrow_v U}{MN \Downarrow_v U}$$

A notion of β-reduction that is correct with respect to $\approx_v$ is Plotkin's $\rightarrow_{\beta_v} \subseteq \Lambda \times \Lambda$, that is,

$$(\lambda x.M)N \rightarrow_{\beta_v} M[N/x], \quad \text{if } N \text{ is a variable or an abstraction.}$$

Notice that the β_v-reduction is far from being the largest notion of β-reduction correct with respect to $\approx_v$. For example, we can extend this notion by allowing the reduction whenever N β_v-reduces to a variable or an abstraction. The characterization of *call-by-value*-redexes, that is, $\approx_v \cap =_\beta$, is given by

$$\{\langle(\lambda x.M)N, M[N/x]\rangle \mid \forall \rho \in [Var \rightarrow Val_v].\ N^\rho \Uparrow_v \Longrightarrow (M[N/x])^\rho \Uparrow_v\},$$

where, for $M \in \Lambda$ such that $FV(M) \subseteq \{x_1, \ldots, x_n\}$, and $\rho \in [Var \rightarrow Val_v]$, M^ρ denotes the closed term $M[\rho(x_1)/x_1, \ldots, \rho(x_n)/x_n]$.

The reduction strategy $\rightarrow_v$ is the one implemented by the $SECD$ machine of Landin and used in ML.

A computationally adequate ordered λ-model for $\leqslant_v$ is the model $\mathscr{D}^v$, studied in Egidi *et al.* (1992). The model $\mathscr{D}^v$ is the quadruple $(D^v, \sqsubseteq^v, \bullet_D^v, [\![\ ]\!]^v)$, where D^v is the inverse limit initial solution of the equation $D \simeq [D \rightarrow_\perp D]_\perp$ in the category $CPO_\perp$, and $\bullet_D^v$ and $[\![\ ]\!]^v$ are defined using the canonical isomorphisms, which are given by the inverse limit construction. We recall that $[D \rightarrow_\perp D]$ denotes the cpo of *strict* continuous functions. Computational adequacy follows from the following theorem.

Theorem 4.11. (Computational adequacy of $\mathscr{D}^v$)

$$M \Downarrow_v \iff \forall \rho.\ [\![M]\!]^v_\rho \sqsupseteq_v \boldsymbol{\lambda} d \in D^v.\ \perp \ .$$

4.1.3. $\leqslant_h$. The precongruence $\leqslant_h$ is determined by the *head call-by-name* strategy $\rightarrow_h \subseteq \Lambda \times \Lambda$, which reduces the *leftmost* β-redex, if the term is not in head normal form. Val_h is the set of λ-terms in head normal form. The evaluation $\Downarrow_h$ is the least binary relation over $\Lambda \times Val_h$ satisfying the following rules, for $n \geqslant 0$:

$$\frac{}{xM_1 \ldots M_n \Downarrow_h xM_1 \ldots M_n} \qquad \frac{M \Downarrow_h N}{\lambda x.M \Downarrow_h \lambda x.N} \qquad \frac{M[N/x]M_1 \ldots M_n \Downarrow_h P}{(\lambda x.M)NM_1 \ldots M_n \Downarrow_h P}$$

β-reduction is correct with respect to $\approx_h$ (see for example, Barendregt (1984)).

In the next definition we introduce an alternative axiomatization of the notion of $\rightarrow_h$-convergence on closed terms, $\Downarrow_h^0$, which will be useful in Section 6. Notice the somewhat 'lazy' flavour of the abstraction rule in this proof system.

Definition 4.12. Let $\Downarrow_h^o$ be the least binary relation over $\Lambda^0 \times \Lambda^0$ satisfying the following rules, for $n \geqslant 0$:

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

$$\frac{M \in Val_h}{M \Downarrow_h^o M} \qquad \frac{\lambda x.M \notin Val_h \quad M[P/x] \Downarrow_h^o N}{\lambda x.M \Downarrow_h^o \lambda x.M} \qquad \frac{M[N/x]M_1 \ldots M_n \Downarrow_h^o P}{(\lambda x.M)NM_1 \ldots M_n \Downarrow_h^o P}$$

The following theorem clarifies the meaning of the $\Downarrow_h^o$.

Theorem 4.13. For all $M \in \Lambda^0$,

(i) $\exists N \in \Lambda^0.\ M \Downarrow_h^o N \ \Rightarrow\ M \Downarrow_h$;
(ii) $\forall N \in \Lambda^0.\ M \Downarrow_h^o N \ \Rightarrow\ M \rightarrow_h^* N$;
(iii) $M \Downarrow_h \Rightarrow M \Downarrow_h^o$.

Proof. (i) and (ii) are proved by induction on the length of the derivation of $M \Downarrow_h^o N$.

The proof of (iii) follows from the fact that, for all $M \in \Lambda$ such that $FV(M) \subseteq \{x_1, \ldots, x_n\}$, $M \Downarrow_h \Rightarrow\ \exists N, P_1, \ldots, P_n \in \Lambda^0.\ M[P_1/x_1, \ldots, P_n/x_n] \Downarrow_h^o N$. This latter fact is proved by induction on the length of the derivation of $M \Downarrow_h$. □

A fully abstract ordered λ-model for $\leqslant_h$ is given the well-known Scott $\mathscr{D}_\infty$ model, which, by uniformity, we will call $\mathscr{D}^h$. The model $\mathscr{D}^h$ is given by the quadruple

$$(D^h, \sqsubseteq^h, \bullet_{D^h}, [\![\ ]\!]^h),$$

where D^h is the inverse limit solution of the equation $D \simeq [D \rightarrow D]$ in the category CPO, starting from the initial domain $D_0^h = \{\bot, \top\}$ and using the initial projection $j_{\top,\bot}^h : D_1^h \rightarrow D_0^h$ defined by $j_{\top,\bot}^h(d) = \bot$, if $d \neq \lambda d \in D_0^h.\top$, $j_{\top,\bot}^h(\lambda d \in D_0^h.\top) = \top$. The definition of application and intepretation are given using the canonical isomorphisms given by the inverse limit construction. Computational adequacy follows from the following theorem.

Theorem 4.14. (Computational adequacy of $\mathscr{D}^h$)

$$M \Downarrow_h \iff \exists \rho.\ [\![M]\!]_\rho^h \neq \bot\ .$$

4.1.4. $\leqslant_o$. The precongruence $\leqslant_o$ is determined by the non-deterministic strategy $\rightarrow_o \subseteq (\Lambda^{\{\Omega\}})^0 \times (\Lambda^{\{\Omega\}})^0$ (Honsell and Ronchi della Rocca 1992). This strategy rewrites λ-terms that contain occurrences of the constant Ω by reducing any β-redex. $Val_o = \Lambda^0$. Normal forms that are not in Val_o are the $\rightarrow_o$-deadlock terms. The evaluation relation $\Downarrow_o$ is the least binary relation over $(\Lambda^{\{\Omega\}})^0 \times Val_o$ satisfying the following rules:

$$\frac{M \in Val_o}{M \Downarrow_o M} \qquad \frac{C[(\lambda x.M)N] \notin Val_o \quad C[M[N/x]] \Downarrow_o P}{C[(\lambda x.M)N] \Downarrow_o P}$$

β-reduction is trivially correct with respect to $\approx_o$.

A computationally adequate ordered λ-model for $\leqslant_o$ is the model $\mathscr{D}^o$, introduced in Honsell and Ronchi (1992). The model $\mathscr{D}^o$ is the quadruple $(D^o, \sqsubseteq^o, \bullet_{D^o}, [\![\ ]\!]^o)$, where D^o is Park's inverse limit solution of the equation $D \simeq [D \rightarrow D]$ in the category CPO, obtained starting from the domain $D_0^o = \{\bot, v\}$ and the initial projection $j_{1,0}^o : D_1^o \rightarrow D_0^o$ defined by $j_{1,0}^o(\bot) = \bot$, $j_{1,0}^o(d) = v$, for $d \neq_{D^o} \bot$. Application and abstraction in $\mathscr{D}^o$ are defined using the canonical isomorphism given by the inverse limit construction. Computational adequacy of $\mathscr{D}^o$ follows from the following theorem.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Theorem 4.15. (Computational adequacy of $\mathscr{D}^o$)

$$M \Downarrow_o \iff \exists \rho.\ [\![M]\!]^o_\rho \neq \bot \ .$$

4.1.5. $\leqslant_n$. The precongruence $\leqslant_n$ is determined by the *normalizing* strategy $\rightarrow_n \subseteq \Lambda \times \Lambda$, which reduces the leftmost β-redex. Val_n is the set of λ-terms in normal form. The evaluation $\Downarrow_n$ is the least binary relation over $\Lambda \times Val_n$ satisfying the following rules, for $n \geqslant 0$:

$$\frac{M_1 \Downarrow_n M_1' \ \dots \ M_n \Downarrow_n M_n'}{xM_1 \dots M_n \Downarrow_n xM_1' \dots M_n'} \qquad \frac{M \Downarrow_n N}{\lambda x.M \Downarrow_n \lambda x.N} \qquad \frac{M[N/x]M_1 \dots M_n \Downarrow_n P}{(\lambda x.M)NM_1 \dots M_n \Downarrow_n P}$$

β-reduction is correct with respect to $\approx_n$.

A computationally adequate ordered λ-model for $\leqslant_n$ is the model $\mathscr{D}^n$, studied in Coppo *et al.* (1987). The model $\mathscr{D}^n$ is the quadruple $(D^n, \sqsubseteq^n, \bullet_{D^n}, [\![\ ]\!]^n)$, where D^n is the inverse limit solution of the equation $D \simeq [D \to D]$ in the category CPO, which is obtained by starting with the initial domain $D_0^n = \{\bot, 0, 1\}$, with $0 \sqsubseteq_n 1$, and the initial projection $j^n_{1,0} : D_1^n \to D_0^n$ defined by $j^n_{1,0}(\bot) = \bot$, $j^n_{1,0}(d) = 0$, if $d \neq \bot, \boldsymbol{\lambda} d \in D_0^n.1$, $j_{1,0}(\boldsymbol{\lambda} d \in D_0^n.1) = 1$. Application and interpretation in D^n are defined using the canonical isomorphism given by the inverse limit construction. Computational adequacy of $\mathscr{D}^n$ follows from the following theorem.

Theorem 4.16. (Computational adequacy of $\mathscr{D}^n$)

$$M \Downarrow_n \iff \exists \rho. [\![M]\!]^n_\rho \sqsupseteq^n 0.$$

4.1.6. $\leqslant_p$. The precongruence $\leqslant_p$ is determined by any *perpetual strategy*, such as Barendregt's *perpetual* strategy $\rightarrow_p \subseteq \Lambda \times \Lambda$, which reduces the leftmost β-redex not in the operator of a redex, which is either an $I\beta$-redex[†] or a $K\beta$-redex[‡] whose argument is a normal form. Val_p is the set of λ-terms in normal form. One can easily show that the evaluation $\Downarrow_p$ is the least binary relation over $\Lambda \times Val_p$ satisfying the following rules for $n \geqslant 0$:

$$\frac{M_1 \Downarrow_p M_1' \ \dots \ M_n \Downarrow_p M_n'}{xM_1 \dots M_n \Downarrow_p xM_1' \dots M_n'} \qquad \frac{M \Downarrow_p N}{\lambda x.M \Downarrow_p \lambda x.N}$$

$$\frac{N \Downarrow_p \quad M[N/x]M_1 \dots M_n \Downarrow_p V}{(\lambda x.M)NM_1 \dots M_n \Downarrow_p V}$$

The following notion of β-reduction, $\rightarrow_{\beta_{KN}}$, defined by

$$(\lambda x.M)N \rightarrow_{\beta_{KN}} M[N/x], \qquad \begin{array}{l}\text{if } (\lambda x.M)N \text{ is either an } I\beta\text{-redex} \\ \text{or a } K\beta\text{-redex with } N \in \Lambda^0 \text{ and } N \Downarrow_p,\end{array}$$

[†] $(\lambda x.M)N$ is an $I\beta$-redex if $x \in FV(M)$.
[‡] $(\lambda x.M)N$ is a $K\beta$-redex if $x \notin FV(M)$.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

is correct with respect to $\leqslant_p$. The characterization of *perpetual*-redexes, that is, $\approx_p \cap =_\beta$, is given by

$$\{\langle (\lambda x.M)N, M[N/x]\rangle \mid \forall \rho \in [Var \to Val_p].\ N^\rho \Uparrow_p \Longrightarrow (M[N/x])^\rho \Uparrow_p\}\ ,$$

where, for $M \in \Lambda$ such that $FV(M) \subseteq \{x_1, \ldots, x_n\}$, and $\rho \in [Var \to Val_p]$, M^ρ denotes the closed term $M[\rho(x_1)/x_1, \ldots, \rho(x_n)/x_n]$. See Honsell and Lenisa (1998) for more information.

Notice that $\approx_p$ is not very well behaved, we do not have, for instance, the property that if $M \to_p N$ then $M \approx_p N$. Consider for example $(\lambda x.(\lambda xy.x)xx)$ and $\lambda x.x$.

As we did in the case of $\to_h$, we can introduce an alternative 'lazy' axiomatization of $\Downarrow_p$, to be used in Section 6.

Definition 4.17. Let $\Downarrow_p^o$ be the least binary relation over $\Lambda^0 \times \Lambda^0$ satisfying the following rules for $n \geqslant 0$:

$$\frac{M \in Val_p}{M \Downarrow_p^o M} \qquad \frac{\lambda x.M \notin Val_p \quad M[P/x] \Downarrow_p^o N}{\lambda x.M \Downarrow_p^o \lambda x.M} \qquad \frac{N \Downarrow_p^o \quad M[N/x]M_1 \ldots M_n \Downarrow_p^o V}{(\lambda x.M)NM_1 \ldots M_n \Downarrow_p^o V}$$

The proof of the following theorem is similar to the proof of Theorem 4.13.

Theorem 4.18. For all $M \in \Lambda^0$,

(i) $\exists N \in \Lambda^0.\ M \Downarrow_p^o N \ \Rightarrow\ M \Downarrow_p$;
(ii) $\forall N \in \Lambda^0.\ M \Downarrow_p^o N \ \Rightarrow\ M \to_p^* N$;
(iii) $M \Downarrow_p \Rightarrow M \Downarrow_p^o$.

A computationally adequate model for $\leqslant_p$ is the model $\mathscr{D}^p$, studied in Honsell and Lenisa (1998). The model $\mathscr{D}^p$ is the quadruple $(D^p, \sqsubseteq^p, \bullet_{D^p}, [\![\ ]\!]^p)$, where D^p is the inverse limit solution of the equation $D \simeq [D \to_\perp D]$ in the category $CPO_\perp$, which is obtained by starting from the initial domain $D_0^p = \{\perp, 0, 1\}$, with $0 \sqsubseteq_p 1$, and the initial projection $j_{1,0}^p : D_1^p \to D_0^p$ defined by $j_{1,0}^p(\perp) = \perp$, $j_{1,0}^p(d) = 0$, if $d \neq_{D^p} \perp$, $d \neq_{D^p} \lambda d \in D_0^p$. if $d =_{D^p} \perp$ then $\perp$ else 1 and $j_{1,0}(\lambda d \in D_0^p.$ if $d =_{D^p} \perp$ then $\perp$ else $1) = 1$. Application and interpretation are defined using the canonical isomorphism given by the inverse limit construction. Computational adequacy follows from the following theorem.

Theorem 4.19. (Computational adequacy of $\mathscr{D}^p$)

$$M \Downarrow_p \iff \exists \rho.\ [\![M]\!]_\rho^p \neq \perp\ .\ [\![M]\!]^p \sqsupseteq^p 0\ .$$

5. Cartesian applicative structures

As we pointed out in the Introduction and Section 2, most of the existing literature on coinduction principles for applicative structures has focussed on Abramsky's notion of *applicative bisimulation*, and hence, in our terminology, only on the operator Φ_{Eq_D}. The most important results that have been obtained in this direction are that all the six observational λ-theories presented in Section 4.1 can be characterized using the $\Phi_{\mathsf{Eq}_\sigma}$-coinduction principle – see Egidi *et al.* (1992), Abramsky and Ong (1993), Howe (1996),

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Pitts (1996b), Lenisa (1997a), Lenisa (1997b) and Lenisa (1998). We can rephrase these results by saying that the term models $\mathscr{T}_{\leqslant_\sigma}$, which are indeed λ-models (Lenisa 1998, Theorem 7.6.4), are $\Phi_{\mathsf{Eq}_\sigma}$-coinductive applicative structures, and hence, by Corollary 3.3, they can be viewed as strongly extensional $H_2^{\Phi_{\mathsf{Eq}_\sigma}}$-coalgebras.

The only exceptions to the hegemony of applicative coinduction principles in the literature appear in Honsell and Lenisa (1995) and Lenisa (1998). In the former, the authors introduced the coinduction operator Ψ_{Eq_D} and they proved, using a semantical technique, that $\mathscr{T}_{\leqslant_v}$ is an Eq_v-cartesian applicative structure. Lenisa (1998) proved, using a syntactical technique, that both $\mathscr{T}_{\leqslant_v}$ and $\mathscr{T}_{\leqslant_l}$ are cartesian.

However, general theorems that allow us to extend the results for Φ_{Eq_D} to the other coinduction operators presented in Section 2 are far from obvious. In this section we show that large classes of applicative structures satisfying a Φ_{Eq_D}-coinduction principle, are also Eq_D-cartesian. The technique presented in this section is semantical, in Section 6 we shall discuss and extend the purely syntactical technique of Lenisa (1998).

The semantical technique, called *Finitary Method*, applies to the *approximable applicative structures* introduced in Definition 2.13. The main result is that all $\Phi_{\mathsf{pEq}^0{}_D}$-coinductive approximable applicative structures (see Defintion 2.14) are $\mathsf{Eq}^0{}_D$-cartesian. Where pEq^0_D and Eq^0_D are defined as follows.

Definition 5.1. Let $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ be an aas.

1 Let $\mathsf{pEq}^0_D \subseteq D \times D$ be defined by $\{(d,e) \in D \times D \mid d_0 \sqsubseteq_D e_0\}$;
2 Let $\mathsf{Eq}^0_D \subseteq D \times D$ be defined by $\{(d,e) \in D \times D \mid d_0 =_D e_0\}$.

In particular, the Finitary Method can be used to prove $\Phi_{\mathsf{Eq}^0{}_D}$- cartesianity of many λ-aas.

Notice that the equality relation $=_D$ on a pEq^0_D-caas immediately satisfies the $\Phi_{\mathsf{Eq}^0_D}$-applicative coinduction principle.

In the following proposition, we provide an alternative characterization of the partial order relation on a pEq^0_D-caas.

Proposition 5.2. Let $(D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ be a pEq^0_D-caas. Then

$$d \sqsubseteq_D e \iff \forall n \geqslant 0.\ \forall \vec{f} \in D^n.\ (d\vec{f})_0 \sqsubseteq_D (e\vec{f})_0 \ .$$

Proof. The implication ($\Leftarrow$) follows from the fact that $\{(d,e) \mid \forall \overrightarrow{f}.\ (d\overrightarrow{f})_0 \sqsubseteq_D (e\overrightarrow{f})_0\}$ is a $\Phi_{\mathsf{pEq}^0_D}$-bisimulation. The converse, that is, ($\Rightarrow$), follows immediately from the fact that $d \sqsubseteq_D e \implies d\overrightarrow{f} \sqsubseteq_D e\overrightarrow{f}$. □

Finally, we can give one of the crucial theorems of this paper.

Theorem 5.3. Let $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega})$ be a pEq^0_D-caas. Then $\mathscr{D}$ is a Eq^0_D-cartesian applicative structure, that is,

$$\approx^{\Psi_{\mathsf{Eq}^0_D}} = =_D \ .$$

Proof. The inclusion $=_D \subseteq \approx^{\Psi_{\mathsf{Eq}^0_D}}$ is immediate, since a $\Phi_{\mathsf{Eq}^0_D}$-bisimulation is a $\Psi_{\mathsf{Eq}^0_D}$-bisimulation. In order to show the reverse inclusion (that is, $\approx^{\Psi_{\mathsf{Eq}^0_D}} \subseteq =_D$), we proceed as

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

follows. Assume the contrary, that is, $\exists d, e.\ d \approx^{\Psi_{\mathsf{Eq}^0_D}} e \wedge d \not\sqsubseteq_D e$. Suppose n is the least natural such that $d \approx^{\Psi_{\mathsf{Eq}^0_D}} e$ but $d_n \not\sqsubseteq_D e$. Then $n \geqslant 1$, by definition of $\approx^{\Psi_{\mathsf{Eq}^0_D}}$. Hence, by Proposition 5.2, there exist $f^1, \ldots, f^m \in D$, such that $(d_n f^1 \ldots f^m)_0 \not\sqsubseteq_D (e f^1 \ldots f^m)_0$. But, since $d \approx^{\Psi_{\mathsf{Eq}^0_D}} e$, there exist $g^1, \ldots, g^m$ such that $g^i \approx^{\Psi_{\mathsf{Eq}^0_D}} f^i$, for all i, and $d g^1 \ldots g^m \approx^{\Psi_{\mathsf{Eq}^0_D}} e f^1 \ldots f^m$, and hence $(d g^1 \ldots g^m)_0 =_D (e f^1 \ldots f^m)_0$, and $(d_n f^1 \ldots f^m)_0 \not\sqsubseteq_D (d g^1 \ldots g^m)_0$. Assume for simplicity that $m \leqslant n$ (the other case is dealt with similarly). By the induction hypothesis, for all $i = 1, \ldots, m$, $(f^i)_{n-i} \sqsubseteq g^i$, hence we have $(d_n f^1 \ldots f^m)_0 =_D (d_n (f^1)_{n-1} \ldots (f^m)_{n-m})_0 \sqsubseteq_D (d g^1 \ldots g^m)_0$, which contradicts $(d_n f^1 \ldots f^m)_0 \not\sqsubseteq_D (d g^1 \ldots g^m)_0$. □

Using the categorical framework of Final Semantics, the above theorem states that pEq^0_D-caas's can be construed as strongly extensional coalgebras for the functor $H^{\Psi_{\mathsf{Eq}^0_D}}_{|D_0|}$.

5.1. *Examples of cartesian approximable applicative structures*

In this section, we present many examples of $\mathsf{pEq}^0_D\lambda$- caas's.

5.1.1. *CPO λ-models.* Many 'inverse limit' CPO λ-models studied in the literature – for example, those mentioned in Section 4.1 – are $\mathsf{pEq}^0_D\lambda$-caas. Therefore, by Theorem 5.3, they are Eq^0_D-cartesian applicative structures. Notice that, for $\sigma = \{l, v\}$, we have to take as D^σ_0 the 1-projection domain of D^σ, that is, the two-element CPO $\{\bot, \lambda d.\ \bot\}$.

5.1.2. *Quotient λ-models.* Another interesting class of $\mathsf{pEq}^0_D\lambda$-caas arises from quotienting the *applicative and projective closure* of the *interior* of a suitable λ-aas. We recall that $\mathscr{I}^D$, the interior of a model $\mathscr{D}$, is the subset consisting of the denotations of $(\Lambda^C)^0$. The *applicative and projective closure* of the *interior* of a λ- model $\mathscr{D}$ can be defined easily using *indexed terms*, that is, those terms generated by extending the set of constants with symbols to denote projections.

Definition 5.4. The set of indexed λ-terms Λ^{C^+} is defined as follows:

$$A \ ::= \ x \mid AA \mid \lambda x.A \mid c \mid c^{p_n} ,$$

where $x \in Var$, $c \in C$.

The set of closed indexed terms will be denoted by $(\Lambda^{C^+})^0$.

In the following we shall assume that $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega}, [\![\ ]\!])$ is a λ-aas modelling the set of λ-terms Λ^C. We can easily extend it canonically to a λ-aas $\mathscr{D}^+$, which models Λ^{C^+}, by extending $[\![\ ]\!]^D$ on the new constants as follows: for all $n \in \omega$, $[\![c^{p_n}]\!]^{D^+}_\rho = p_n$. Then the applicative and projective closure of $\mathscr{I}^D$ is $\mathscr{I}^{D^+}$.

In the rest of this section, we will show how to quotient $\mathscr{I}^{D^+}$ by a suitable equivalence relation $\equiv^\omega$, and we will discuss conditions under which this quotient structure is a $\mathsf{pEq}^0\lambda$-caas. In effect, we start by introducing an appropriate relation, which will be shown to be a pre-equivalence relation in Proposition 5.9 below. Notice that, for all $n \geqslant 0$, we have that $\forall d.\ p_n d =_D ([\![\lambda x.x]\!]^D_\rho)_{n+1} d$.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Definition 5.5. Let $\sqsubseteq^{\omega}_{D} \subseteq \mathscr{I}^{D^+} \times \mathscr{I}^{D^+}$ be the relation defined by

$$a \sqsubseteq^{\omega}_{D} b \iff \forall n \geqslant 0.\ a_n \sqsubseteq^{n}_{D} b_n\ ,$$

where the relations $\sqsubseteq^{n}_{D} \subseteq (\mathscr{I}^{D^+} \cap D_n) \times (\mathscr{I}^{D^+} \cap D_n)$ are inductively defined as follows:

— $a \sqsubseteq^{0}_{D} b \iff a \sqsubseteq_{D} b$.

— $a \sqsubseteq^{n+1}_{D} b \iff (a_0 \sqsubseteq^{0}_{D} b_0 \ \wedge\ \forall c \sqsubseteq^{n}_{D} d.\ ac \sqsubseteq^{n}_{D} bd)$.

Let $\equiv^{\omega}_{D}$ be the relation $\sqsubseteq^{\omega}_{D} \cap (\sqsubseteq^{\omega}_{D})^{-1}$.

The notion $\sqsubseteq^{\omega}_{D}$ above generalizes the notion of *call-by-value applicative bisimulation* introduced in Egidi *et al.* (1992), and the corresponding notion used in Abramsky and Ong (1993) for the lazy λ-calculus. In Egidi *et al.* (1992), this relation was used to show that the theory $\approx_v$ is Eq_v-applicative. Notice that it is not at all obvious that the relation $\sqsubseteq^{\omega}_{D}$ is a pre-equivalence and a congruence with respect to application. The proof of this (Proposition 5.9) and, in particular, of the fact that $\sqsubseteq^{\omega}_{D}$ is reflexive is rather involved. It generalizes the one carried out in Egidi *et al.* (1992) for *call-by-value applicative bisimulation*. Once we have this, we can define a quotient λ-model by endowing $Q^{+}_{D} = (\mathscr{I}^{D^+} / \equiv^{\omega}_{D})$ with the structure of a λ-aas (Corollary 5.10). Then, we will show that the relation $\sqsubseteq^{\omega}_{D}$ satisfies a $\Phi_{\mathsf{pEq}^0_{Q^+_D}}$-coinduction principle. Therefore the quotient structure Q^{+}_{D} is a $\mathsf{pEq}^0_{Q^+_D}\lambda$-caas and hence, by Theorem 5.3, it is $\mathsf{pEq}^0_{Q^+_D}$-cartesian.

We will start by isolating a crucial property of the relations $\sqsubseteq^{n}_{D}$.

Lemma 5.6. $\forall n \geqslant 0,\ \forall a, b \in \mathscr{I}^{D^+}$,

(i) $a \sqsubseteq^{n+1}_{D} b \implies a_n \sqsubseteq^{n}_{D} b_n$;

(ii) $a \sqsubseteq^{n}_{D} b \implies a \sqsubseteq^{n+1}_{D} b$.

Proof. The proof of (i) and (ii) is by mutual induction on n.

Base case: $a \sqsubseteq^{1}_{D} b \implies a_0 \sqsubseteq^{0}_{D} b_0$, by definition; $a \sqsubseteq^{0}_{D} b \implies a \sqsubseteq^{1}_{D} b$, by monotonicity of $\bullet_D$.

Induction step: Assume $a \sqsubseteq^{n+1}_{D} b$. For (i), we have only to show that $a_n \sqsubseteq^{n}_{D} b_n$, that is, $\forall c \sqsubseteq^{n-1}_{D} d.\ a_n c \sqsubseteq^{n-1}_{D} b_n d$. By the induction hypothesis, $c \sqsubseteq^{n-1}_{D} d \implies c \sqsubseteq^{n}_{D} d$, and hence $ac \sqsubseteq^{n}_{D} bd$, by hypothesis. Again by the induction hypothesis, $ac \sqsubseteq^{n}_{D} bd \implies (ac)_{n-1} \sqsubseteq^{n-1}_{D} (bd)_{n-1}$, that is (by properties of projections), $a_n c \sqsubseteq^{n-1}_{D} b_n d$. To establish (ii), we only have to show that $a \sqsubseteq^{n+2}_{D} b$, that is, $\forall c \sqsubseteq^{n+1}_{D} d.\ ac \sqsubseteq^{n+1}_{D} bd$. If $c \sqsubseteq^{n+1}_{D} d$, then, by (i), $c_n \sqsubseteq^{n}_{D} d_n$. Therefore, by hypothesis, $ac_n \sqsubseteq^{n}_{D} bd_n$. By the induction hypothesis, $ac_n \sqsubseteq^{n+1}_{D} bd_n$, that is, since $a, b \in \mathscr{I}^{D^+} \cap D_{n+1}$, we have $ac \sqsubseteq^{n+1}_{D} bd$. □

Lemma 5.7. For all $a, b \in \mathscr{I}^{D^+}$,

$$a \sqsubseteq^{\omega}_{D} b \iff a_0 \sqsubseteq^{\omega}_{D} b_0 \ \wedge\ \forall c, d \in \mathscr{I}^{D^+}.\ (c \sqsubseteq^{\omega}_{D} d \ \Rightarrow\ ac \sqsubseteq^{\omega}_{D} bd)\ .$$

Proof. ($\Rightarrow$) This follows from the fact that in finitary models $ab =_D \bigsqcup_{n \in \mathbf{N}} a_n b$, and from the fact that $\sqsubseteq^{\omega}_{D}$ is inclusive, that is, if $\forall n.\ a_n \sqsubseteq^{\omega}_{D} b_n$, then $a \sqsubseteq^{\omega}_{D} b$.

($\Leftarrow$) First of all, notice that, using Lemma 5.6, one can easily show that $\forall a, b \in \mathscr{I}^{D^+}_{n}.\ a \sqsubseteq^{n}_{D} b \iff a \sqsubseteq^{\omega}_{D} b$. Using this fact, one can easily show that $\forall n \geqslant 0.\ a_{n+1} \sqsubseteq^{n+1}_{D} b_{n+1}$. The thesis then follows using the hypothesis $a_1 \sqsubseteq^{1}_{D} b_1 \implies a_0 \sqsubseteq^{0}_{D} b_0$. □

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Lemma 5.8. Let $D_{var} \cap \mathscr{I}^{D^+} \neq \varnothing$. Then, for all $M \in \Lambda^{C^+}$, for all $\rho, \rho' : Var \to (D_{var} \cap \mathscr{I}^{D^+})$,

$$\rho \sqsubseteq_D^\omega \rho' \implies [\![M]\!]_\rho^D \sqsubseteq_D^\omega [\![M]\!]_{\rho'}^D ,$$

where $\rho \sqsubseteq_D^\omega \rho'$ denotes the fact that $\forall x \in Var.\ \rho(x) \sqsubseteq_D^\omega \rho'(x)$

Proof. The proof is by structural induction. In particular, for the application case, use Lemma 5.7($\Rightarrow$), and for the abstraction case, use Lemma 5.7($\Leftarrow$). □

Proposition 5.9. Let $D_{var} \cap \mathscr{I}^{D^+} \neq \varnothing$. Then, the relation $\sqsubseteq_D^\omega$ is a pre-equivalence, which is a congruence with respect to application.

Proof. Reflexivity follows from Lemma 5.8, since the interpretation of closed terms in $(\Lambda^{C^+})^0$ does not depend on the environment, and, by assumption, there exist environments $\rho, \rho' : Var \to (D_{var} \cap \mathscr{I}^{D^+})$ such that $\rho \sqsubseteq_D^\omega \rho'$. Transitivity follows by showing, by induction on n, that, the relations $\sqsubseteq_D^n$ are all transitive. The fact that $\sqsubseteq_D^\omega$ is a congruence with respect to application follows from Lemma 5.7($\Rightarrow$). □

Finally we can define a λ-caas out of the quotient Q_D^+.

Corollary 5.10. Let $\mathscr{D} = (D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega}, [\![\ ]\!])$ be a λ-aas. If $(\mathscr{I}^{D^+} / \equiv_D^\omega)$ is non-trivial, then the structure $(Q_D^+, \widetilde{\sqsubseteq}_D^\omega, \bullet, \{[p_n]_{\equiv_D^\omega}\}_{n\in\omega}, [\![\ ]\!])$ is a λ-aas, where:

— $Q_D^+ \equiv (\mathscr{I}^{D^+} / \equiv_D^\omega)$;
— $\widetilde{\sqsubseteq}_D^\omega$ is the partial order induced on the quotient by $\sqsubseteq_D^\omega$;
— $[d]_{\equiv_D^\omega} \bullet [e]_{\equiv_D^\omega} = [de]_{\equiv_D^\omega}$, where $[d]_{\equiv_D^\omega}$ denotes the equivalence class of d modulo $\equiv_D^\omega$;
— for all $\rho : Var \to (Q_D^+)_{var}$, $[\![M]\!]_\rho = [M[\tilde{\rho}(x_1)/x_1, \ldots, \tilde{\rho}(x_n)/x_n]]_{\equiv_D^\omega}$, where $FV(M) \subseteq \{x_1, \ldots, x_n\}$, and $\tilde{\rho}(x_i)$ is a representative of the equivalence class $\rho(x_i)$.

We can show that all non-trivial quotient structures are $\Phi_{pEq^0_{Q_D^+}}$-coinductive. To this end, we need to give an alternative *applicative* characterization of the relation $\sqsubseteq_D^\omega$. But first we need the following lemma, which is easily proved by induction on n, using Lemma 5.6.

Lemma 5.11. For all $n \geqslant 0$,

$$a \sqsubseteq_D^n b \implies \forall \vec{a}.\ (a\vec{a})_0 \sqsubseteq_D (b\vec{a})_0 .$$

Proposition 5.12.

$$a \sqsubseteq_D^\omega b \iff \forall \vec{a}.\ (a\vec{a})_0 \sqsubseteq_D (b\vec{a})_0 .$$

Proof. ($\Rightarrow$) This follows by contradiction, using Lemma 5.11 and properties of application in an aas.
($\Leftarrow$) Suppose, for a contradiction, that $\exists n.\ a_n \not\sqsubseteq_D^n b_n$. Then $n > 0$. By definition of $\sqsubseteq_D^n$, there exist $a_{n-1}^1, \ldots, a_0^n$, $b_{n-1}^1, \ldots, b_0^n$ such that $\forall i = 1, \ldots, n.\ a_{n-i}^i \sqsubseteq_D^{n-i} b_{n-i}^i \ \wedge\ (a_n a_{n-1}^1 \ldots a_0^n)_0 \not\sqsubseteq_D (b_n b_{n-1}^1 \ldots b_0^n)_0$. Moreover, by Lemma 5.6, $a_{n-i}^i \sqsubseteq_D^\omega b_{n-i}^i$. Then, since $\sqsubseteq_D^\omega$ is a congruence, we have $(b_n a_{n-1}^1 \ldots a_0^n)_0 \sqsubseteq_D^\omega (b_n b_{n-1}^1 \ldots b_0^n)_0$, and hence *a fortiori* $(aa_{n-1}^1 \ldots a_0^n)_0 \not\sqsubseteq_D (ba_{n-1}^1 \ldots a_0^n)_0$, which is a contradiction by Lemma 5.11. □

The proof of the following theorem can be easily derived from the proposition above.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Theorem 5.13. Let $(D, \sqsubseteq_D, \bullet_D, \{p_n\}_{n\in\omega}, [\![\ ]\!]^D)$ be a λ-aas. If Q_D^+ is non-trivial, then the structure $(Q_D^+, \widetilde{\sqsubseteq}_D^{\omega}, \bullet, \{[p_n]_{\approx_D^{\omega}}\}_{n\in\omega}, [\![\ ]\!])$, defined in Corollary 5.10, is a $\mathsf{pEq}^0_{Q_D^+}\lambda$-caas.

Using Theorem 5.3, we immediately get the following corollary.

Corollary 5.14. Let $\mathscr{D}$ be a λ-aas. If Q_D^+ is non-trivial, then the approximable applicative structure $(Q_D^+, \widetilde{\sqsubseteq}_D^{\omega}, \bullet, \{[p_n]_{\equiv_D^{\omega}}\}_{n\in\omega})$ is a $\mathsf{pEq}^0_{Q_D^+}$-cartesian applicative structure.

Corollary 5.14 can be applied to the quotient λ-aas generated by all the models $\mathscr{D}^h$, $\mathscr{D}^l$, $\mathscr{D}^v$, $\mathscr{D}^o$, $\mathscr{D}^n$, and $\mathscr{D}^p$ of Section 4.1. Again, for $\sigma = \{l, v\}$, we have to take as D_0^σ the 1-projection domain of D^σ, that is, the two-element CPO $\{\bot, \lambda d.\ \bot\}$. Moreover, since projection functions are λ-definable in the models $\mathscr{D}^v$ and $\mathscr{D}^o$, we can also use Corollary 5.14 to show that, for $\sigma = v, o$ the theories $\approx_\sigma$ are Eq_σ-cartesian. In fact, using the Computational Adequacy Theorems 4.11 and 4.15, one can check that $\mathsf{Eq}_\sigma = \mathsf{Eq}^0_{Q_{D^\sigma}^+}$.

6. The congruence candidate method

In this section, we present a purely syntactical technique for establishing Eq_σ-cartesianity of observational λ-theories $\approx_\sigma$. We recall that $\mathsf{Eq}_\sigma = \{(M, N) \in (\Lambda^C)^0 \times (\Lambda^C)^0 \mid M \Downarrow_\sigma \Leftrightarrow N \Downarrow_\sigma\}$. This method, which we call *congruence candidate method*, was introduced in Lenisa (1998). It is inspired by the congruence candidate method used in Howe (1989) and Howe (1996), where it is used to show that $\approx_\sigma$ for lazy strategies is $\Phi_{\mathsf{Eq}_\sigma}$-coinductive. A first extension of Howe's method is in Lenisa (1997b), where it is used to show $\Phi_{\mathsf{Eq}_\sigma}$-coinduction for strategies other than lazy (for example, $\sigma = h, n, p$). The congruence candidate method will be used here for showing that the λ-theories $\approx_\sigma$, for $\sigma = h, l, v, p$, are Eq_σ-cartesian. In general, it can be applied to observational λ-theories, which satisfy the $\Phi_{\mathsf{Eq}_\sigma}$-coinduction and the technical condition $(*)$ of Theorem 6.5 below.

The congruence candidate method makes an essential use of the Eq_σ-cartesian coinduction principle itself. One starts out by defining a *candidate relation* on Λ^C, which is a congruence, and which includes the greatest $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$. This candidate relation is then shown to be a $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. Hence the Eq_σ- cartesian coinduction principle guarantees that $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$ itself is a congruence with respect to application and thus it coincides with $\approx^{\Phi_{\mathsf{Eq}_\sigma}}$. Since this method is very technical we will begin by giving a general outline.

General pattern of the congruence candidate method

— Build a *candidate relation* $\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} \subseteq \Lambda^C \times \Lambda^C$ such that
  1 $\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} \supseteq \approx^{\Psi_{\mathsf{Eq}_\sigma}}$;
  2 $\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}}$ is a congruence with respect to application;
  3 $(\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}})_{|(\Lambda^C)^0}$ is a $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation.

— Use the Eq_σ-cartesian coinduction principle to deduce that $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$ is a congruence with respect to application.

We shall now present the congruence candidate method in general. The following Definitions and Lemmata build up to Theorem 6.5. In the next subsection we shall apply Theorem 6.5 to various strategies.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

In order to discuss uniformly both call-by-value and eager strategies, we introduce the following notation.

Notation. Let $\rightarrow_\sigma$ be a strategy. We put:

$$\Delta_\sigma = \begin{cases} \{M \in (\Lambda^C)^0 \mid M \Downarrow_\sigma\} & \text{if } \forall M, N \in (\Lambda^C)^0.\ N \not\Downarrow_\sigma \Rightarrow MN \not\Downarrow_\sigma, \\ (\Lambda^C)^0 & \text{otherwise.} \end{cases}$$

First, we have to explain how to build the *candidate relation* $\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}}$. These are defined in terms of the extensions to open terms of $\Psi_{\mathsf{Eq}_\sigma}$-bisimulations.

Definition 6.1. Let $\mathscr{R}$ be a $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. Then, for all $M, N \in \Lambda^C$ such that $FV(M,N) \subseteq \{x_1, \ldots, x_k\}$, we define

$$\begin{array}{l} M\ \mathscr{R}^{ext}\ N \iff \\ \forall P_1, \ldots, P_k \in \Delta_\sigma.\ \exists Q_1, \ldots, Q_k \in \Delta_\sigma. \\ (\forall i = 1, \ldots, k.\ P_i\ \mathscr{R}\ Q_i\ \wedge\ M[P_1/x_1 \ldots P_k/x_k]\ \mathscr{R}\ N[Q_1/x_1 \ldots Q_k/x_k]) \wedge \\ \forall Q_1, \ldots, Q_k \in \Delta_\sigma.\ \exists P_1, \ldots, P_k \in \Delta_\sigma. \\ (\forall i = 1, \ldots, k.\ P_i\ \mathscr{R}\ Q_i\ \wedge\ M[P_1/x_1 \ldots P_k/x_k]\ \mathscr{R}\ N[Q_1/x_1 \ldots Q_k/x_k])\ . \end{array}$$

In the following, by abuse of notation, we will simply denote $\mathscr{R}^{ext}$ by $\mathscr{R}$.

Definition 6.2. (Candidate relation) Let $\mathscr{R} \subseteq \Lambda^C \times \Lambda^C$ be a reflexive and transitive $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. Define the *candidate relation* $\widehat{\mathscr{R}} \subseteq \Lambda^C \times \Lambda^C$ by induction on M as follows:

$$\frac{x\ \mathscr{R}\ N}{x\ \widehat{\mathscr{R}}\ N} \qquad \frac{M_1\ \widehat{\mathscr{R}}\ M_1' \quad M_2\ \widehat{\mathscr{R}}\ M_2' \quad M_1'M_2'\ \mathscr{R}\ N}{M_1M_2\ \widehat{\mathscr{R}}\ N}$$

$$\frac{c\ \mathscr{R}\ N}{c\ \widehat{\mathscr{R}}\ N} \qquad \frac{M\ \widehat{\mathscr{R}}\ M' \quad \lambda x.M'\ \mathscr{R}\ N}{\lambda x.M\ \widehat{\mathscr{R}}\ N}$$

Notice that the candidate relation is not simply the contextual closure of $\mathscr{R}$; this subtle definition of $\widehat{\mathscr{R}}$, originally due to Howe, is necessary to guarantee the crucial Substitutivity Lemma 6.4. The following lemma is an easy consequence of the definition of $\widehat{\mathscr{R}}$.

Lemma 6.3. Let $\mathscr{R} \subseteq \Lambda^C \times \Lambda^C$ be a reflexive and transitive $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. Then:

(i) $\widehat{\mathscr{R}}$ is reflexive.
(ii) $\mathscr{R} \subseteq \widehat{\mathscr{R}}$.
(iii) $\widehat{\mathscr{R}}$ is a congruence with respect to application.
(iv) $M\widehat{\mathscr{R}}M'\ \wedge\ M'\mathscr{R}N \implies M\widehat{\mathscr{R}}N$.

Lemma 6.4. (Substitutivity) Let $\mathscr{R} \subseteq \Lambda^C \times \Lambda^C$ be a reflexive and transitive $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. For all $M, M' \in \Lambda^C$, $N, N' \in \Delta_\sigma$,

$$M\widehat{\mathscr{R}}M'\ \wedge\ N\widehat{\mathscr{R}}N' \implies M[N/x]\widehat{\mathscr{R}}M'[N'/x]\ .$$

Proof. The proof is by induction on the structure of M.

- $M \equiv x$: $\quad \dfrac{x\ \mathscr{R}\ M'}{x\ \widehat{\mathscr{R}}\ M'}$

$x\mathscr{R}M' \implies \exists P \in \Delta_\sigma.\ N'\mathscr{R}P\ \wedge\ P\mathscr{R}M'[N'/x]$, by definition of $\mathscr{R}$, and hence, by transitivity

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

of $\mathscr{R}$, $N'\mathscr{R}M'[N'/x]$

$$N\widehat{\mathscr{R}}N' \ \wedge \ N'\mathscr{R}M'[N'/x] \implies N\widehat{\mathscr{R}}M'[N'/x],$$

from (iv) of Lemma 6.3.

- $M \equiv c$: $\dfrac{c \ \mathscr{R} \ M'}{c \ \widehat{\mathscr{R}} \ M'}$ this case is immediate.

- $M \equiv M_1M_2$: $\exists M_1', M_2'$ such that $\dfrac{M_1 \ \widehat{\mathscr{R}} \ M_1' \quad M_2 \ \widehat{\mathscr{R}} \ M_2' \quad M_1'M_2' \ \mathscr{R} \ M'}{M_1M_2 \ \widehat{\mathscr{R}} \ M'}$

By definition of $\mathscr{R}$, $\exists P \in \Delta_\sigma$ such that $N'\mathscr{R}P$ and $M_1'M_2'[P/x]\mathscr{R}M'[N'/x]$. In particular, from $N\widehat{\mathscr{R}}N'$ and $N'\mathscr{R}P$, we get $N\widehat{\mathscr{R}}P$. By the induction hypothesis, $M_1[N/x]\widehat{\mathscr{R}}M_1'[P/x]$ and $M_2[N/x]\widehat{\mathscr{R}}M_2'[P/x]$. Hence

$$\frac{M_1[N/x] \ \widehat{\mathscr{R}} \ M_1'[P/x] \quad M_2[N/x] \ \widehat{\mathscr{R}} \ M_2'[P/x] \quad M_1'M_2'[P/x] \ \mathscr{R} \ M'[N'/x]}{M_1M_2[N/x] \ \widehat{\mathscr{R}} \ M'[N'/x]} .$$

- $M \equiv \lambda y.M_1$: $\exists M_1'$ such that $\dfrac{M_1 \ \widehat{\mathscr{R}} \ M_1' \quad \lambda y.M_1' \ \mathscr{R} \ M'}{\lambda y.M_1 \ \widehat{\mathscr{R}} \ M'}$

By the definition of $\mathscr{R}$, $\exists P \in \Delta_\sigma$ such that $N'\mathscr{R}P$ and $(\lambda y.M_1')[P/x]\mathscr{R}M'[N'/x]$. In particular, from $N\widehat{\mathscr{R}}N'$ and $N'\mathscr{R}P$, we get $N\widehat{\mathscr{R}}P$. By the induction hypothesis, we have $M_1[N/x]\widehat{\mathscr{R}}M_1'[P/x]$. Hence:

$$\frac{M_1[N/x] \ \widehat{\mathscr{R}} \ M_1'[P/x] \quad (\lambda y.M_1')[P/x] \ \mathscr{R} \ M'[N'/x]}{(\lambda y.M_1)[N/x] \ \widehat{\mathscr{R}} \ M'[N'/x]} .$$

□

Thus, if we take $\mathscr{R}$ to be the equivalence $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$, we get a relation $\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}}$, which, by (ii) of Lemma 6.3, includes $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$. Moreover, by (iii) of the same lemma, it is a congruence with respect to application. In order to show that $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$ is itself a congruence with respect to application, we prove that $(\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}})_{|(\Lambda^C)^0} = (\approx^{\Psi_{\mathsf{Eq}_\sigma}})_{|(\Lambda^C)^0}$. This is done using the $\Psi_{\mathsf{Eq}_\sigma}$-coinduction principle, by proving that $(\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}})_{|(\Lambda^C)^0}$ is a $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation. In order to prove that $(\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}})_{|(\Lambda^C)^0}$ is a $\Psi_{\mathsf{Eq}_\sigma}$-bisimulation, it is sufficient to show that, for all $M, N \in (\Lambda^C)^0$,

$$M\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}}N \ \wedge \ M \Downarrow_\sigma \implies N \Downarrow_\sigma .$$

Hence we can state the following theorem.

Theorem 6.5. If, for all $M, N \in (\Lambda^C)^0$,

$$M\widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}}N \ \wedge \ M \Downarrow_\sigma \quad \implies \quad N \Downarrow_\sigma, \qquad (*)$$

then $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$ is a congruence with respect to application.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

6.1. *The congruence candidate method at work*

The validity of hypothesis (∗) of Theorem 6.5 depends on the particular strategy. In this section we show that hypothesis (∗) holds for all the observational congruences induced by the strategies $\rightarrow_l$, $\rightarrow_v$, $\rightarrow_h$, $\rightarrow_p$. First we need the following two lemmata. Notice that for the 'non-lazy' strategies h and p we refer to the alternative 'lazy' axiomatizations $\Downarrow^0_\sigma$ introduced in Definitions 4.12 and 4.17. This is an essential ingredient for proving Theorem 6.9 below. Notice that, in Lenisa (1997b) and Lenisa (1998) the proof of the fact that the theory $\approx_\sigma$ is Φ_{Eq_D}-coinductive, for $\sigma \in \{h, n, p\}$, is based directly on $\Downarrow_\sigma$, but a straightforward application of a similar argument for cartesianity breaks down. This explains also why proving cartesianity for $\approx_n$ is problematic, since for this theory there is no immediate 'lazy' axiomatization of convergence.

Lemma 6.6. Let $\sigma \in \{l, v, h, p\}$. Then, for all $(\lambda x.M)N \in \Lambda^0$ and $N \in \Delta_\sigma$,

$$(\lambda x.M)N \approx^{\Psi_{\mathsf{Eq}_\sigma}} M[N/x] \ .$$

Proof. It is easy to see that

$$(\lambda x.M)N \approx^{\Phi_{\mathsf{Eq}_\sigma}} M[N/x] \ .$$

The thesis follows from $\approx^{\Phi_{\mathsf{Eq}_\sigma}} \subseteq \approx^{\Psi_{\mathsf{Eq}_\sigma}}$. □

Lemma 6.7.

(i) Let $\sigma \in \{h, p\}$. Then, for all $M, P \in \Lambda^0$

$$M \Downarrow^0_\sigma P \ \Rightarrow \ M \approx^{\Psi_{\mathsf{Eq}_\sigma}} P \ .$$

(ii) Let $\sigma \in \{l, v\}$. Then, for all $M \in \Lambda^0$

$$M \Downarrow_\sigma P \ \Rightarrow \ M \approx^{\Psi_{\mathsf{Eq}_\sigma}} P \ .$$

Proof. (i) It is sufficient to prove that $M \approx^{\Phi_{\mathsf{Eq}_\sigma}} P$, and this follows from the correctness of $\rightarrow_\beta (\rightarrow_{\beta_{KN}})$ reduction.
(ii) The proof is similar to that for item (i). □

First we prove that condition (∗) holds for the 'lazy' strategies.

Theorem 6.8. Let $M, N \in \Lambda^0$, and let $\sigma \in \{l, v\}$. Then

$$M \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} N \ \wedge \ M \Downarrow_\sigma \lambda x.P \implies \exists Q.(N \Downarrow_\sigma \lambda x.Q \ \wedge \ P \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} Q) \ .$$

Proof. The proof proceeds by induction on the derivation of $M \Downarrow_\sigma \lambda x.P$.

- $M \equiv \lambda x.P$: $\exists N'$ such that $\dfrac{P \ \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} N' \quad \lambda x.N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} N}{\lambda x.P \ \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} N}$

From the definition of $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$, it follows that there exists Q such that $N \Downarrow_\sigma \lambda x.Q$. By Lemma 6.6, $N \approx^{\Psi_{\mathsf{Eq}_\sigma}} \lambda x.Q$, and hence, by transitivity of $\approx^{\Psi_{\mathsf{Eq}_\sigma}}$, $\lambda x.N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} \lambda x.Q$. In particular, using Lemma 6.6 again, it is easy to check that $N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} Q$. In fact: in order to show $N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} Q$, we have to show that $\forall P \exists R.\ P \approx^{\Psi_{\mathsf{Eq}_\sigma}} R \ \wedge \ N'[P/x] \approx^{\Psi_{\mathsf{Eq}_\sigma}} Q[R/x]$. But, $\lambda x.N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} \lambda x.Q$ implies $\forall P \exists R.\ P \approx^{\Psi_{\mathsf{Eq}_\sigma}} R \ \wedge \ (\lambda x.N')P \approx^{\Psi_{\mathsf{Eq}_\sigma}} (\lambda x.Q)R$. Then the thesis follows from Lemma

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

6.6. Hence, from $P \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} N'$ and $N' \approx^{\Psi_{\mathsf{Eq}_\sigma}} Q$, using (iv) of Lemma 6.3, we get $P \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} Q$.

- $M \equiv M_1 M_2 : \quad \exists N_1, N_2 \quad$ such that $\quad \dfrac{M_1 \ \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} \ N_1 \quad M_2 \ \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} \ N_2 \quad N_1 N_2 \approx^{\Psi_{\mathsf{Eq}_\sigma}} N}{M_1 M_2 \ \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} \ N}$

We assume without loss of generality that $N_1 N_2 \in \Lambda^0$, otherwise one can apply the Substitutivity Lemma.
We deal with the case $\sigma = l$, the other case is similar. Since $M_1 M_2 \Downarrow_l$, there exist P, P' such that

$$\frac{M_1 \Downarrow_l \lambda x.P' \quad P'[M_2/x] \Downarrow_l \lambda x.P}{M_1 M_2 \Downarrow_l \lambda x.P} \ .$$

By the induction hypothesis, since $M_1 \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} N_1$ and $M_1 \Downarrow_l \lambda x.P'$, there exists Q' such that $N_1 \Downarrow_l \lambda x.Q'$ and $P' \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} Q'$. By the Substitutivity Lemma, $P'[M_2/x] \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} Q'[N_2/x]$. Hence, by the induction hypothesis, there exists Q'' such that $Q'[N_2/x] \Downarrow_l \lambda x.Q''$ and $P \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} Q''$. Moreover, from $N_1 N_2 \approx^{\Psi_{\mathsf{Eq}_l}} N \ \wedge \ N_1 N_2 \Downarrow_l \lambda x.Q''$, using Lemma 6.7 (ii), $\exists Q.\ N \Downarrow_l \lambda x.Q \approx^{\Psi_{\mathsf{Eq}_l}} \lambda x.Q''$. But $\lambda x.Q'' \approx^{\Psi_{\mathsf{Eq}_l}} \lambda x.Q \implies Q'' \approx^{\Psi_{\mathsf{Eq}_l}} Q$, by definition of $\approx^{\Psi_{\mathsf{Eq}_l}}$. Finally, by Lemma 6.3 (iv), $P \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} Q'' \ \wedge \ Q'' \approx^{\Psi_{\mathsf{Eq}_l}} Q \implies P \widehat{\approx}^{\Psi_{\mathsf{Eq}_l}} Q$. □

Next we discuss the 'eager' strategies, notice again the use of $\Downarrow^o_\sigma$.

Theorem 6.9. Let $M, N \in \Lambda^0$, and let $\sigma \in \{h, p\}$. Then

$$M \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} N \ \wedge \ M \Downarrow^o_\sigma \lambda x.P \implies \exists Q.(N \Downarrow^o_\sigma \lambda x.Q \ \wedge \ P \widehat{\approx}^{\Psi_{\mathsf{Eq}_\sigma}} Q) \ .$$

Proof. The proof proceeds by induction on the derivation of $M \Downarrow^o_\sigma \lambda x.P$. For simplicity, we will only work out in detail the case of $\sigma = h$, the other case being similar.

- The only rule applied in the derivation of $M \Downarrow^o_h \lambda x.P$ is the first rule in Definition 4.12. Then M is solvable, that is, $M \equiv \lambda x_1 \dots x_n.x_i M_1 \dots M_k$. Hence we have $\exists N_1, \dots, N_k, N^0, \dots, N^{k-1}, N'_1, \dots, N'_n$ such that

$$\dfrac{\dfrac{\dfrac{x_i \approx^{\Psi_{\mathsf{Eq}_h}} N^0}{x_i \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N^0} \quad M_1 \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N_1 \quad N^0 N_1 \approx^{\Psi_{\mathsf{Eq}_h}} N^1}{x_i M_1 \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N^1}}{\begin{array}{c} \vdots \\ x_i M_1 \dots M_{k-1} \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N^{k-1} \qquad M_k \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N_k \quad N^{k-1} N_k \approx^{\Psi_{\mathsf{Eq}_h}} N'_n \\ \hline x_i M_1 \dots M_k \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N'_n \end{array}}$$

and

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

$$\dfrac{\dfrac{\dfrac{x_iM_1\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'_n \quad \lambda x_n.N'_n\approx^{\Psi_{\mathsf{Eq}_h}}N'_{n-1}}{\lambda x_n.x_iM_1\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'_{n-1}}}{\vdots}}{\dfrac{\lambda x_2\ldots x_n.x_iM_1\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'_1 \qquad \lambda x_1.N'_1\approx^{\Psi_{\mathsf{Eq}_h}}N}{\lambda x_1\ldots x_n.x_iM_1\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N}}$$

In order to show that $\exists Q.\ N\Downarrow_h^o\lambda x_1.Q\ \wedge\ Q\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}\lambda x_2\ldots x_n.x_iM_1\ldots M_k$, it is sufficient to prove that $\lambda x_1.N'_1\Downarrow_h^o$. Then the thesis follows using Theorem 4.13 and Lemma 6.3 (iv), and using the fact that $\lambda x_2\ldots x_n.x_iM_1\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'_1$. Then we prove $\lambda x_1.N'_1\Downarrow_h^o$.
$x_i\approx^{\Psi_{\mathsf{Eq}_h}}N^0\implies x_iN_1\approx^{\Psi_{\mathsf{Eq}_h}}N^0N_1$
$x_iN_1\approx^{\Psi_{\mathsf{Eq}_h}}N^0N_1\ \wedge\ N^0N_1\approx^{\Psi_{\mathsf{Eq}_h}}N^1\implies x_iN_1\approx^{\Psi_{\mathsf{Eq}_h}}N^1$
$\vdots$
$x_iN_1\ldots N_k\approx^{\Psi_{\mathsf{Eq}_h}}N^{k-1}N_k\ \wedge\ N^{k-1}N_k\approx^{\Psi_{\mathsf{Eq}_h}}N'_n\implies x_iN_1\ldots N_k\approx^{\Psi_{\mathsf{Eq}_h}}N'_n$.
Using Theorem 4.13, $N'_n\Downarrow_h^o\Rightarrow\ \lambda x_n.N'_n\Downarrow_h^o\Rightarrow N'_{n-1}\ \Downarrow_h^o\Rightarrow\ \ldots N'_1\Downarrow_h^o\Rightarrow\ \lambda x_1.N'_1\Downarrow_h^o$.

- $M\equiv\lambda x.P$ and M is not a head normal form. Then

$$\exists N'\quad\text{such that}\quad\dfrac{P\ \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'\quad\lambda x.N'\approx^{\Psi_{\mathsf{Eq}_h}}N}{\lambda x.P\ \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N}$$

Without loss of generality, we assume that $\lambda x.N'\in\Lambda^0$, otherwise one can apply the Substitutivity Lemma. Since $\lambda x.P\Downarrow_h^o\lambda x.P$, then, by definition of $\Downarrow_h^o$, $\exists P'.\ P[P'/x]\Downarrow_h^o$. From $P\ \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'$ and $P'\ \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}P'$, by the Substitutivity Lemma, $P[P'/x]\ \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'[P'/x]$. Hence, by induction hypothesis, $N'[P'/x]\Downarrow_h^o$, and then $\lambda x.N'\Downarrow_h^o\lambda x.N'$ and $P\Downarrow_h^oN'$. But, since $\lambda x.N'\approx^{\Psi_{\mathsf{Eq}_h}}N$, we have $\exists Q.\ N\Downarrow_h^o\lambda x.Q$ and then, using Lemma 6.7 (i), $\lambda x.N'\approx^{\Psi_{\mathsf{Eq}_h}}\lambda x.Q$. Using Lemma 6.6, we get $N'\approx^{\Psi_{\mathsf{Eq}_h}}Q$. Finally, from $P\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N'$ and $N'\approx^{\Psi_{\mathsf{Eq}_h}}Q$, we get $P\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}Q$.

- $M\equiv(\lambda x.M_1)M_2\ldots M_k$. Then, by hypothesis $\exists V$ such that

$$\dfrac{M_1[M_2/x]M_3\ldots M_k\Downarrow_h^0V}{(\lambda x.M_1)M_2\ldots M_k\Downarrow_h^0V}\quad k\geqslant 2$$

and $\exists N_1,\ldots,N_k,N^1,\ldots,N^{k-1}$ such that

$$\dfrac{\dfrac{\dfrac{M_1\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N_1\quad\lambda x.N_1\approx^{\Psi_{\mathsf{Eq}_h}}N^1}{\lambda x.M_1\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N^1}\qquad M_2\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N_2\quad N^1N_2\approx^{\Psi_{\mathsf{Eq}_h}}N^2}{(\lambda x.M_1)M_2\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N^2}}{\vdots}}{\dfrac{(\lambda x.M_1)M_2\ldots M_{k-1}\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N^{k-1}\qquad M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N_k\quad N^{k-1}N_k\approx^{\Psi_{\mathsf{Eq}_h}}N}{(\lambda x.M_1)M_2\ldots M_k\widehat{\approx}^{\Psi_{\mathsf{Eq}_h}}N}}$$

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Hence

$$N^{k-2}N_{k-1} \approx^{\Psi_{\mathsf{Eq}_h}} N^{k-1} \ \wedge\ N^{k-1}N_k \approx^{\Psi_{\mathsf{Eq}_h}} N \implies N^{k-2}N_{k-1}N_k \approx^{\Psi_{\mathsf{Eq}_h}} N$$
$$N^{k-3}N_{k-2} \approx^{\Psi_{\mathsf{Eq}_h}} N^{k-2} \ \wedge\ N^{k-2}N_{k-1}N_k \approx^{\Psi_{\mathsf{Eq}_h}} N \implies N^{k-3}N_{k-2}N_{k-1}N_k \approx^{\Psi_{\mathsf{Eq}_h}} N$$
$$\vdots$$
$$N^1N_2 \approx^{\Psi_{\mathsf{Eq}_h}} N^2 \ \wedge\ N^2N_3 \dots N_k \approx^{\Psi_{\mathsf{Eq}_h}} N \implies N^1N_2 \dots N_k \approx^{\Psi_{\mathsf{Eq}_h}} N$$
$$\lambda x.N_1 \approx^{\Psi_{\mathsf{Eq}_h}} N^1 \ \wedge\ N^1N_2 \dots N_k \approx^{\Psi_{\mathsf{Eq}_h}} N \implies (\lambda x.N_1)N_2 \dots N_k \approx^{\Psi_{\mathsf{Eq}_h}} N.$$

Without loss of generality, we assume that $(\lambda x.N_1)N_2 \dots N_k \in \Lambda^0$, otherwise one can apply the Substitutivity Lemma. In order to show that $\exists Q.N \Downarrow_h^o \lambda x.Q \ \wedge\ P \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} Q$, it is sufficient to show that $\exists Q'.(\lambda x.N_1)N_2 \dots N_k \Downarrow_h^o \lambda x.Q' \ \wedge\ P \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} Q'$. Then the thesis follows using Lemma 6.6 and Lemma 6.7 (i). Hence we only need to show that $\exists Q'.N_1[N_2/x] \dots N_k \Downarrow_p^o \lambda x.Q' \ \wedge\ P \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} Q'$. Using the Substitutivity Lemma, we get $M_1[M_2/x] \dots M_k \widehat{\approx}^{\Psi_{\mathsf{Eq}_h}} N_1[N_2/x] \dots N_k$. Finally, applying the induction hypothesis, we get the thesis. □

7. Set-theoretical applicative structures

In this section we briefly discuss set-theoretical applicative structures, and their connections with Eq_D-cartesian applicative structures. First of all, we introduce the crucial definition.

Definition 7.1. A set D is a *set-theoretical applicative structure over a set* U_D, if

1 $U_D \cap (D \times D) = \varnothing$
2 $D \subseteq \mathscr{P}((D \times D) \cup U_D)$;
3 $\forall d_1, d_2 \in D.\ \exists! d_3 \in D.\ (d_2, d_3) \in d_1$.

In the above definition the set U_D plays the role of a set of *Urelementen* used for tagging different copies of the same graph. The condition on U_D allows us to introduce the notion of set-theoretical applicative structure even in an atomless universe.

Definition 7.2. A *functional* set-theoretical applicative structure is a set-theoretical applicative structure D such that $D \subseteq D^D$.

The above definitions are justified by the following obvious fact.

Proposition 7.3. Let D be a set-theoretical applicative structure. Then the structure $(D, \bullet_D)$, where

$$d_1 \bullet_D d_2 = d_3 \iff (d_2, d_3) \in d_1\ ,$$

is an applicative structure.

The following proposition shows that the existence of set-theoretical applicative structures is sensitive to the foundation/antifoundation properties of the universe.

Proposition 7.4.

1 The Foundation Axiom implies that there are no set-theoretical applicative structures.
2 BAFA (see Aczel (1988)) implies that all functional applicative structures are applicatively isomorphic to an extensional set-theoretical applicative structure.
3 $\mathsf{ZF}^\circ\mathsf{X}_1$ (see Appendix A) implies that there are no non-trivial functional set-theoretical applicative structures.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Proof. (1) This is immediate, since a set-theoretical applicative structure is a non-wellfounded set.
(2) This is immediate from the definition of BAFA.
(3) This is immediate from the strong extensionality of a universe satisfying X_1. □

Hence the theory of set-theoretical applicative structures is most interesting when the universe satisfies the Axiom X_1 of Forti and Honsell (1983) (see Appendix A). In this case, the universe itself satisfies the well-known *strong extensionality* property, which amounts exactly to the fact that the universe is a final $\mathscr{P}(\ )$-coalgebra (see Appendix A).

The following proposition casts light on the connections between set-theoretical applicative structures and Ψ_{Eq_D}-cartesian coinduction principles in a universe satisfying X_1.

Proposition 7.5. Assume $\mathsf{ZF}^{\circ}\mathsf{X}_1$.

1 Any applicative structure of cardinality κ is applicatively isomorphic to a set-theoretical applicative structure D over a set U_D of cardinality κ.
2 Let D be a set-theoretical applicative structure over the set U_D. Then $(D, \bullet_D)$ is an Eq_D-cartesian applicative structure, for Eq_D defined by $d\ \mathsf{Eq}_D\ d' \Leftrightarrow d \cap U_D = d' \cap U_D$. Moreover, (D, id_D) is a strongly extensional $H^{\Psi_{\mathsf{Eq}_D}}_{|U_D|}$-coalgebra in Set^*.
3 Any Eq_D-cartesian applicative structure is applicatively isomorphic to a set-theoretical applicative structure over a set U_D whose cardinality is the cardinality of the equivalence classes of Eq_D.

Proof. The proof is straightforward using the definitions. □

It is amusing to note that the very last proposition of this paper was what actually triggered the whole investigation carried out in the rest of the paper.

8. Concluding remarks

— In Honsell and Lenisa (1995), we have already given a proof of the fact that the theory $\approx_v$ is Eq_v-cartesian. The proof that derives from the general method of Section 5 is conceptually simpler. The results concerning the strategies l and v, and techniques in Section 6 are essentially those of Lenisa (1998).

— In this paper we did not fully address all possible natural questions that can arise in connection with the operator Ψ_{Eq_D}, let alone all the other operators of Section 2. For instance, one could ask whether $\approx_n$ is Eq_n-cartesian. Although we confidently conjecture that this is the case, the proof could be extremely technical, since in defining $\Downarrow^o_n$ for abstractions, one should test termination on infinitely many closed terms.
Some interesting observations concerning the other operators can be made readily. For example, the $\Phi^n_{\mathsf{Eq}_\sigma}$-coinduction principle, for $n > 1$, is clearly unsound for lazy strategies, but it is sound for those strategies that yield extensional term models such as h, n, o. More general results, however, seem extremely hard to establish.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Appendix A. Non-wellfounded sets

Non-wellfounded sets are elements of a Universe of a Zermelo Fräenkel-like set-theory $\mathsf{ZF}^{\circ}\mathsf{X}_1$. $\mathsf{ZF}^{\circ}\mathsf{X}_1$ is the theory consisting of the axioms *extensionality*, *Pairing*, *Union*, *Power Set*, *Replacement*, *Infinity*, *Choice*, and the Antifoundation Axiom X_1 of Forti and Honsell (1983) (or equivalently, the Antifoundation Axiom *AFA* of Aczel (1988)).

Let V denote the Universe of sets (without atoms).

Definition 1. (X_1) Let X be a set. For every function $f : X \to \mathscr{P}(X)$, there is a *unique* function $g : X \to V$ which makes the following diagram commute

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ {\scriptstyle \alpha_X}\downarrow & & \downarrow{\scriptstyle \alpha_Y} \\ F(X) & \xrightarrow[F(f)]{} & F(Y) \end{array}$$

That is, for all $x \in X$, $g(x) = \{g(y) \mid y \in f(x)\}$.

It is interesting to note that X_1 expresses precisely the fact that the universe V is *final* coalgebra for the functor $\mathscr{P}(\cdot)$.

The Antifoundation Axiom X_1 immediately yields a coinductive characterization of equality between sets, that is, strong extensionality (Forti and Honsell 1983; Aczel 1988).

Proposition 2. (Strong extensionality) Two sets x, y are equal if and only if there exists a Φ^+- *bisimulation* $\mathscr{R}$ such that $x \mathscr{R} y$, where Φ^+ is the following operator on binary relations on the universe V:

$$\Phi^+(\mathscr{R}) = \{(x, y) \mid \forall x_1 \in x.\ \exists y_1 \in y.\ x_1 \mathscr{R} y_1 \ \wedge\ \forall y_1 \in y.\ \exists x_1 \in x.\ x_1 \mathscr{R} y_1\} .$$

Notice that Φ^+-bisimulations are called *id-admissible relations* in Forti and Honsell (1983).

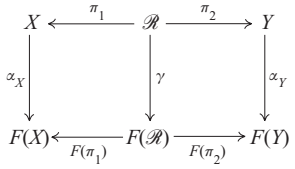
Appendix B. Coalgebraic description of coinduction

In this appendix we recall the categorical *coalgebraic* description of coinduction. This arises from the *Final Semantics Paradigm* introduced by Aczel (Aczel 1988; Aczel and Mendler 1989), and further developed by Rutten and Turi (Rutten and Turi 1993; Rutten 1996; Turi 1996).

Coalgebraically, coinduction principles for reasoning on the possibly circular and infinite objects of a data type X arise when X can be viewed as *F-coalgebra* for a suitable endofunctor $F : \mathscr{C} \to \mathscr{C}$.

Definition 3. Let $F : \mathscr{C} \to \mathscr{C}$ be a functor. An *F-coalgebra* is a pair (X, α_X), where $\alpha_X : X \to F(X)$ is a morphism of $\mathscr{C}$.
The F-coalgebras are the objects of a category whose morphisms between F-coalgebras (X, α_X) and (Y, α_Y) are morphisms $f : X \to Y$ of the category $\mathscr{C}$ such that the following diagram commutes

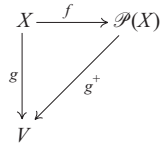
https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press



The categorical counterpart of the the set-theoretic notion of maximal fixpoint is the notion of *final F-coalgebra.*

The categorical counterpart of the set-theoretic notion of bisimulation is the notion of *F-bisimulation.* We will now give the definition of *F-bisimulation* in the category Set^*.

Definition 4. An F-bisimulation on the F-coalgebras (X, α_X) and (Y, α_Y) is a set-theoretic relation $R \subseteq X \times Y$ such that there exists an arrow of $\mathscr{C}$, $\gamma : \mathscr{R} \to F(\mathscr{R})$, making the following diagram commute:



Definition 5. (Strong extensionality) An F-coalgebra, (U, α_U), is *strongly F-extensional* if for all $u, u' \in U$,

$$u = u' \iff \exists\, \mathscr{R} \;\; F\text{-bisimulation on } (U, \alpha_U).\; u\, \mathscr{R}\, u' \;.$$

We now recall the crucial theorem of the Final Semantics Paradigm, which allows us to characterize coinductively the equivalence induced by the unique F-coalgebra morphism into the final F-coalgebra (Aczel 1988; Rutten and Turi 1993).

Theorem 6. Let F preserve weak pullbacks, and let (X, α_X) be an F-coalgebra. If there exists the final F-coalgebra (U, α_U), then the equivalence $\sim_f$ induced by the unique morphism $f : (X, \alpha_X) \to (U, \alpha_U)$ can be characterized as follows:

$$\sim_f \;=\; \bigcup \{ \mathscr{R} \mid \mathscr{R} \text{ is an } F\text{-bisimulation on } (X, \alpha_X) \} \;.$$

Acknowledgements

The authors would like to thank the referees for their useful comments, and A. Quattrocchi for her help.

References

Abramsky, S. (1989) The lazy lambda-calculus. In: Turner, D. (ed.) *Research Topics in Functional Programming*, Addison Wesley 65–116.

Abramsky, S. and Ong, L. (1993) Full Abstraction in the Lazy Lambda Calculus. *Inf. and Comp.* **105** (2) 159–267.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Aczel, P. (1988) Non-well-founded sets. *CSLI Lecture Notes* **14** Stanford.

Aczel, P. and Mendler, N (1989) A Final Coalgebra Theorem. In: Pitt, D. H. *et al.* (eds.) CTCS'89 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **389** 357–365.

Barendregt, H. (1984) *The Lambda Calculus, its Syntax and Semantics,* North Holland, Amsterdam.

Barendregt, H., Coppo, M. and Dezani-Ciancaglini, M. (1983) A filter lambda-model and the completeness of the type assignment. *J. of Symbolic Logic* **48** (4) 931–940.

Coppo, M., Dezani-Ciancaglini, M. and Zacchi, M. (1987) Type Theories, Normal Forms and D_∞-Lambda-Models. *Inf. and Comp.* **72** (2) 85–116.

Egidi, L., Honsell, F. and Ronchi della Rocca, S. (1992) Operational, denotational and logical descriptions: a case study. *Fundamenta Informaticae* **16** (2) 149–169.

Fiore, M. (1996) A Coinduction Principle for Recursive Data Types Based on Bisimulation. *Inf. and Comp.* **127** 186–198.

Forti, M. and Honsell, F. (1983) Set theory with free construction principles. *Ann. Scuola Norm. Sup. Pisa* Cl. Sci. (4) **10** 493–522.

Gordon, A. (1995) Bisimilarity as a Theory of Functional Programming. In: *MFPS'95 Conf. Proc., Electronic Notes in Computer Science* **1**, Elsevier.

Honsell, F. and Lenisa, M. (1993) Some Results on Restricted λ-calculi. In: Borzyszkowski, A. *et al.* (eds.) MFCS'93 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **711** 84–104.

Honsell, F. and Lenisa, M. (1995) Final Semantics for Untyped Lambda Calculus. In: Dezani, M. and Plotkin, G. (eds.) TLCA'95 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **902** 249–265.

Honsell, F. and Lenisa, M. (1999) Semantical analysis of perpetual strategies. *Theoretical Computer Science* **212** 183–209.

Honsell, F. and Ronchi della Rocca, S. (1992) An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *J. of Computer and System Sciences* **45** (1) 49–75.

Howe, D. (1989) Equality in Lazy Computation Systems *4th LICS Conf. Proc.*, IEEE Computer Society Press 198–203.

Howe, D. (1996) Proving Congruence of Bisimulation in Functional Programming Languages. *Inf. and Comp.* **124** (2) 103–112.

Lenisa, M. (1996) Final Semantics for Higher Order Concurrent Languages. In: Kirchner, H. *et al.* (eds.) CAAP'96 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **1059** 102–118.

Lenisa, M. (1997a) Semantic Techniques for Deriving Coinductive Characterizations of Observational Equivalences for λ-calculi. In: de Groote, P. and Hindley, R. (eds.) TLCA'97 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **1210** 63–81.

Lenisa, M. (1997b) A Uniform Syntactical Method for Proving Coinduction Principles in λ-calculi. In: Bidoit, M. *et al.* (eds.) TAPSOFT'97 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **1214** 248–266.

Lenisa, M. (1998) *Themes in Final Semantics*, Ph. D. thesis **TD-6/98**, Dipartimento di Informatica, Università di Pisa, Italy.

Mason, I., Smith, S. and Talcott, C. (1996) From Operational Semantics to Domain Theory. *Inf. and Comp.* (to appear).

Pitts, A. (1996) Relational Properties of Domains. *Inf. and Comp.* **127** 66–90.

Pitts, A. (1996b) A Note on Logical Relations Between Semantics and Syntax. *L. J. of the IGPL* **5** (4) 589–601.

Plotkin, G. (1977) Call-by-name, call-by-value and the lambda calculus. *Theoretical Computer Science* **1** 125–159.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press

Rutten, J. J. M. M. (1996) Universal coalgebra: a theory of systems. Report **CS-R9652**, CWI, Amsterdam.

Rutten, J. J. M. M. and Turi, D. (1993) On the Foundations of Final Semantics: Non-Standard Sets, Metric Spaces, Partial Orders. In: de Bakker, J. *et al.* (eds.) REX Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **666** 477–530.

Rutten, J. J. M. M. and de Vink, E. (1997) Bisimulation for probabilistic transition systems: a coalgebraic approach (extended abstract). In: Degano, P. *et al.* (eds.) ICALP'97 Conf. Proc. *Springer-Verlag Lecture Notes in Computer Science* **1256** 460–470.

Turi, D. (1996) *Functorial Operational Semantics and its Denotational Dual*, Ph. D. thesis, CWI, Amsterdam.

Turi, D. and Rutten, J. J. M. M. (1998) On the foundations of final coalgebra semantics: non-wellfounded sets, partial orders, metric spaces. *Math. Struct. in Comp. Science* **8** 481–540.

https://doi.org/10.1017/S096012959900287X Published online by Cambridge University Press