Electronic Notes in Theoretical Computer Science **20** (1999)
URL: http://www.elsevier.nl/locate/entcs/volume20.html 29 pages

Bisimulation in Untyped Lambda Calculus: Böhm Trees and Bisimulation up to Context

S. B. Lassen [1]

University of Cambridge Computer Laboratory, Pembroke Street, Cambridge CB2 3QG, United Kingdom
Soeren.Lassen@cl.cam.ac.uk

Abstract

On the basis of an operational bisimulation account of Böhm tree equivalence, a novel operationally-based development of the Böhm tree theory is presented, including an elementary congruence proof for Böhm tree equivalence. The approach is also applied to other sensible and lazy tree theories. Finally, a syntactic proof principle, called bisimulation up to context, is derived from the congruence proofs. It is used to give a simple syntactic proof of the least fixed point property of fixed point combinators. The paper surveys notions of bisimulation and trees for sensible λ-theories based on reduction to head normal forms as well as for lazy λ-theories based on weak head normal forms.

1 Introduction

This paper is about operational methods for reasoning about term equivalences in the pure untyped λ-calculus. Bisimulation and tree characterisations of operational equivalences are surveyed, covering both the sensible and lazy call-by-name theories. The sensible theories are based on reduction to head normal form, including reduction under λ-abstraction. They are treated in depth in Barendregt's standard reference on the untyped λ-calculus [3]. The maximal sensible λ-theory is characterised by Böhm trees (modulo η-conversion). The lazy theory is based on reduction to weak head normal form, where reduction stops at λ-abstraction. It was introduced and studied by Abramsky and Ong [1,19,2]. The maximal lazy λ-theory is characterised by applicative bisimulation.

On the basis of an operational bisimulation account of Böhm tree equivalence, a novel operationally-based development of the Böhm tree theory is

[1] Supported by grant number GR/L38356 from the UK EPSRC.

presented, including an elementary congruence proof for Böhm tree equivalence. The approach is also applied to other sensible and lazy tree theories. The bisimulations are variants of Sangiorgi's open bisimulation [21]. Finally, a syntactic proof principle, called bisimulation up to context, is derived from the congruence proofs. It is used to give a simple syntactic proof of the least fixed point property of fixed point combinators.

The development is operationally-based, meaning that it uses elementary techniques of inductive definitions and proofs by rule induction, as in Howe's method for proving congruence of bisimulation equivalences in functional languages [12]. By this approach, co-induction and induction proof rules are established, based purely on bisimulation formulations of tree equivalences, without reference to continuity, semantic models, or even the trees themselves.

A rather different syntactical method for proving congruence of Böhm tree equivalence modulo η-conversion suggested by David and Nour [6] uses techniques based on the definition and analysis of tailored reduction relations. Other existing congruence proofs for tree equivalences [3,19] are more semantically-based, either by connections to domain-theoretic models of the λ-calculus or by continuity properties of the tree models.

Outline

Section 2 surveys the basic concepts of the λ-calculus, tree theories, and applicative bisimulation. Section 3 presents bisimulation formulations of three tree equivalences. They are shown to be compatible in Section 4. In Section 5 bisimulation up to context results are derived from the compatibility proofs. The compatibility proofs are detailed in an appendix.

Familiarity with basic (co-)induction concepts is assumed, see e.g. [14].

2 Lambda theories and trees

This section surveys call-by-name λ-calculus theory; first the basic concepts of λ-calculus, β-reduction and λ-theories, and then the sensible and lazy theories. The presentation gives a simple account of tree-based λ-theories and discusses their congruence properties which are the subject of the remainder of the paper.

2.1 The λ-calculus

The terms $M, N, P, Q, \ldots$ of the pure λ-calculus are just variables $x, y, z, \ldots$ (an infinite supply is assumed), λ-abstractions $\lambda x.\, M$, and applications $M\, N$. The scope of a λ-abstraction extends as far to the right as possible and applications associate to the left, e.g., $\lambda x.\, M\, N\, P = \lambda x.\, ((M\, N)\, P)$. The variable x is said to be bound in $\lambda x.\, M$. As is usual, terms are identified up to α-renaming of bound variables.

The following standard notation will be used in the sequel. $M[{}^{N}\!/_{x}]$ denotes the capture-free substitution of N for all free occurrences of x in M; $\vec{M}$ is an abbreviation for a possibly empty finite sequence of terms $M_1 \ldots M_m$; $\vec{x}$ is notation for a possibly empty sequence of distinct variables $x_1 \ldots x_n$; with $\vec{M}$ and $\vec{x}$ as above, $N\,\vec{M}$ abbreviates $(...(N\,M_1)\ldots)\,M_m$ and $\lambda\vec{x}.\,M$ abbreviates $\lambda x_1.\,\ldots\lambda x_n.\,M$.

The (many-step) β-reduction relation between terms, $\twoheadrightarrow$, is defined inductively as the smallest reflexive, transitive relation which includes β-reduction,

$$(\beta)\quad (\lambda y.\,M)\,N \twoheadrightarrow M[{}^{N}\!/_{y}]$$

and is closed under the two compatibility rules:

$$(\mu\nu)\ \frac{M \twoheadrightarrow N \quad P \twoheadrightarrow Q}{M\,P \twoheadrightarrow N\,Q} \qquad (\xi)\ \frac{M \twoheadrightarrow N}{\lambda y.\,M \twoheadrightarrow \lambda y.\,N}$$

β-conversion, $=_\beta$, is the smallest equivalence relation (reflexive, transitive, and symmetric) which includes β-reduction and is closed under the two compatibility rules.

Let R be a binary relation between terms. R is *compatible* if it is reflexive and is closed under the $(\mu\nu)$ and (ξ) rules.[2] A *congruence* is a compatible equivalence relation (reflexive, transitive, and symmetric). A relation R is a *λ-theory* if (1) R is a congruence; (2) R includes β-reduction; and (3) R is *consistent*, viz. it does not relate all pairs of terms. A λ-theory is *extensional* if it includes η-conversion, viz. it equates M and $\lambda x.\,M\,x$ whenever x is not free in M.

For instance, β-conversion is a λ-theory: it satisfies (1) and (2) by definition, and consistency can be derived from the Church–Rosser confluence theorem. Its consistency also follows from the consistency of other λ-theories because β-conversion is included in every λ-theory. β-conversion is not extensional.

2.2 Sensible theories

The standard, so-called sensible theories [3] are based on reduction to *head normal forms (hnf's)* which are terms, H, of the form $\lambda\vec{x}.\,y\,\vec{M}$, where y is called the *head variable*; it is either free or it is one of the bound variables, $y \in \vec{x}$. A hnf of the form $y\,\vec{M}$ is *λ-free*.

The hnf's are the normal forms of a restricted (one-step) reduction relation called head reduction and written $\rightarrow_{\mathrm{h}}$. It is defined succinctly by the axiom

[2] That is, $(M\,P)\;R\;(N\,Q)$ and $(\lambda x.\,M)\;R\;(\lambda x.\,N)$ hold whenever $M\;R\;N$ and $P\;R\;Q$. This definition of compatibility corresponds to closure of the relation R under arbitrary, possibly many-holed variable-capturing term contexts C; for all C, $C[M]\;R\;C[N]$ holds whenever $M\;R\;N$. The notion of compatibility in [3] is slightly different; it corresponds to closure under single-holed term contexts.

scheme:

$$\text{(red.h)}\quad \lambda\vec{x}.\,(\lambda y.\,M)\,N\,\vec{P} \rightarrow_{\mathrm{h}} \lambda\vec{x}.\,M[{}^{N}\!/\!_{y}]\,\vec{P}$$

The reflexive transitive closure is written $\twoheadrightarrow_{\mathrm{h}}$. Head reduction is deterministic, so any hnf H that M head reduces to, $M \twoheadrightarrow_{\mathrm{h}} H$, is unique and we say that M has hnf H.

There is good evidence for taking the property of having a hnf as the definition of when a term is 'meaningful'; see [23,3]. The 'meaningless' terms, those without hnf's, have infinite head reduction sequences. For instance,

$$\Omega \overset{\mathrm{def}}{=} (\lambda x.\,x\,x)\,(\lambda x.\,x\,x), \qquad \Xi \overset{\mathrm{def}}{=} (\lambda x.\,\lambda y.\,x\,x)\,(\lambda x.\,\lambda y.\,x\,x)$$

have no hnf because $\Omega \rightarrow_{\mathrm{h}} \Omega \rightarrow_{\mathrm{h}} \ldots$ and $\Xi \rightarrow_{\mathrm{h}} \lambda y.\,\Xi \rightarrow_{\mathrm{h}} \lambda y.\,\lambda y.\,\Xi \rightarrow_{\mathrm{h}} \ldots$

It is sensible to equate all terms without hnf, and a λ-theory is called *sensible* if it does just that. A relation is *semi sensible* if it does not relate a term with hnf to a term without hnf. β-conversion is semi sensible but not sensible.

One sensible λ-theory is Barendregt's Böhm tree λ-theory. The Böhm tree of a term M is the possibly infinite labelled tree, $\mathrm{BT}(M)$, defined (co-inductively [14]) as follows.

(1) $\mathrm{BT}(M) = \bot, \quad$ if M has no hnf

(2) $\mathrm{BT}(M) = \begin{array}{c} \lambda\vec{x}.\,y \\ \swarrow \;\cdots\; \searrow \\ \mathrm{BT}(M_1) \qquad \mathrm{BT}(M_n) \end{array}, \quad$ if $M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\,y\,M_1 \ldots M_n$

For instance, consider Curry's and Turing's fixed point combinators:

$$\begin{aligned} \mathsf{Y} &\overset{\mathrm{def}}{=} \lambda f.\,\Delta\,\Delta, \quad \text{where } \Delta \overset{\mathrm{def}}{=} \lambda x.\,f\,(x\,x) \\ \Theta &\overset{\mathrm{def}}{=} (\lambda g.\,\lambda f.\,f\,(g\,g\,f))\,(\lambda g.\,\lambda f.\,f\,(g\,g\,f)) \end{aligned}$$

They have the same infinite Böhm tree, $\mathrm{BT}(\mathsf{Y}) = \mathrm{BT}(\Theta) = \lambda f.\,\begin{array}{c} f \\ | \\ f \\ | \\ \vdots \end{array}$

Equality between Böhm trees, $\mathrm{BT}(M) = \mathrm{BT}(N)$, induces an equivalence relation between terms, *Böhm tree equivalence*, written $M \eqsim_{\mathrm{B}} N$. It is obviously sensible, equating all terms without hnf. It is not extensional, viz. it does not include η-conversion, e.g., $x \not\eqsim_{\mathrm{B}} \lambda y.\,x\,y$.

It is straightforward to show that Böhm tree equivalence is an equivalence relation which includes β-reduction and is consistent. The non-trivial part in establishing that it is a λ-theory is compatibility, in particular, closure under the $(\mu\nu)$ rule. In existing expositions of the Böhm tree theory, compatibility is established either by means of continuity properties of the Böhm tree model [3] or by a correspondence between Böhm trees and a compositional denotational semantics—for instance, compatibility is derivable from Hyland's syntactic characterisation of the equality in Plotkin and Scott's $\mathcal{P}\omega$ model for the λ-calculus [13].

An extensional sensible λ-theory is induced by a notion of Böhm tree equality up to possibly infinite η-expansion. It can be defined co-inductively as the greatest relation $=_\eta$ between Böhm trees such that $\mathrm{BT} =_\eta \mathrm{BT}'$ implies

(1) $\mathrm{BT} = \bot$ and $\mathrm{BT}' = \bot$, or

(η2) $$\mathrm{BT} = \begin{array}{ccc} & \lambda x_1 \dots x_p .\, y & \\ \swarrow & \dots & \searrow \\ \mathrm{BT}_1 & & \mathrm{BT}_m \end{array} \quad \text{and} \quad \mathrm{BT}' = \begin{array}{ccc} & \lambda x_1 \dots x_{p+q} .\, y & \\ \swarrow & \dots & \searrow \\ \mathrm{BT}'_1 & & \mathrm{BT}'_{m+q} \end{array}$$

where $x_{p+1} \dots x_{p+q}$ are not free in $y, \mathrm{BT}_1, \dots, \mathrm{BT}_m$ and $\mathrm{BT}_i =_\eta \mathrm{BT}'_i$ for $1 \le i \le m$ and $x_{p+j} =_\eta \mathrm{BT}'_{m+j}$ for $1 \le j \le q$, or

(2η) the symmetrical case where BT has more leading λ's than BT'.

It is an equivalence relation. By its co-inductive definition $=_\eta$ equates Böhm trees that differ by even an infinite amount of η-conversion, as it were. For instance, let $J \stackrel{\text{def}}{=} \mathsf{Y}\, \lambda f x x'.\, x\, (f\, x')$, then

$$\mathrm{BT}(J\, x) = \begin{array}{lll} \lambda x'.\, x & & \\ \quad | & & \\ & \lambda x''.\, x' & \\ & \quad | & \\ & & \lambda x'''.\, x'' \\ & & \quad | \\ & & \quad \vdots \end{array}$$

is an infinite η-expansion of x, so $x =_\eta \mathrm{BT}(J\, x)$.

Equality of Böhm trees up to possibly infinite η-expansion, $\mathrm{BT}(M) =_\eta \mathrm{BT}(N)$, induces an equivalence relation on terms, *Böhm tree equivalence up to* η, written $M \eqsim_{\mathrm{B}\eta} N$. (See [3] for other tree characterisations of this equivalence relation.)

Clearly, $M \eqsim_{\mathrm{B}} N$ implies $M \eqsim_{\mathrm{B}\eta} N$. Böhm tree equivalence up to η is sensible. It is also extensional: $\mathrm{BT}(M) =_\eta \mathrm{BT}(\lambda x.\, M\, x)$ if x is not free in M.

In establishing that Böhm tree equivalence up to η is a λ-theory, again the main task is to prove is that it is compatible. There do not seem to be any direct proofs of this result in the literature. It can be extracted from the proofs in [13,23] of the coincidence between the equational theory of Scott's D_∞ models and the greatest compatible semi sensible relation; see [3]. The latter relation is the Morris-style contextual equivalence [18], or observational congruence, obtained by observing reduction to hnf ('contextual' equivalence because it can be defined in terms of reduction to hnf in all term contexts). By some characterisations of this relation in [13,23] it can be seen to coincide with Böhm tree equivalence up to η. It also follows that this is the maximal sensible λ-theory.

2.3 Lazy theories

The lazy theories [1,19,2] are based on reduction to *weak head normal forms (whnf's)*, W, consisting of λ-free hnf's, $y\,\vec{M}$, and λ-abstractions, $\lambda x.\,M$.

Whnf's are the normal forms of a (one-step) weak head reduction relation, $\rightarrow_{\mathrm{wh}}$, defined by the axiom scheme:

$$\text{(red.wh)}\quad (\lambda y.\,M)\,N\,\vec{P} \rightarrow_{\mathrm{wh}} M[{}^{N}\!/\!_{y}]\,\vec{P}$$

A term M has whnf W if $M \twoheadrightarrow_{\mathrm{wh}} W$. Weak head reduction is included in head reduction, so every term with a hnf has a whnf. Terms without hnf, either reduce to $\lambda\vec{x}.\,N$ for some N without whnf, or belong to the set of terms of order ∞, defined co-inductively as the largest set of terms such that for every term M of order ∞ there exists a term N of order ∞ such that $M \twoheadrightarrow_{\mathrm{wh}} \lambda x.\,N$. For instance, Ξ and $\Theta\,\lambda fx.f$ have order ∞.

Let us say that a relation is *semi lazy* if it does not relate a term with a whnf to a term with no whnf, and let us call a λ-theory *lazy* if it is semi lazy and equates all terms without whnf.[3] Lazy λ-theories are not sensible and not extensional, e.g., Ω and $\lambda x.\,\Omega\,x$ are discriminated: one has no whnf, the other is a whnf. A simple analysis of terms without hnf reveals that every lazy λ-theory is semi sensible.

Abramsky's applicative bisimulation equivalence [1] is a lazy λ-theory. On closed terms it can be described co-inductively as the greatest semi lazy relation, $\sim$, which is closed under the (μ_0) rule:

$$(\mu_0)\ \frac{M \sim N}{M\,P\ \sim\ N\,P}\ \text{ if } M, N, P \text{ are closed}$$

Open terms M and N with free variables $\vec{x}$ are applicative bisimilar if $\lambda\vec{x}.\,M$ and $\lambda\vec{x}.\,N$ are.

Applicative bisimulation is the maximal lazy λ-theory, and it is the greatest compatible semi lazy relation. The difficult part in proving these results is compatibility. This was proved by Abramsky [1] via a domain logic. Two more elementary, syntactic proofs are the adaptation in [2] of Berry's proof [4] of the context lemma for PCF, and Howe's relational proof in [12]. (See [16] for an overview of different compatibility proofs for various forms of applicative bisimulation for untyped λ-calculus).

For some problems, applicative bisimulation gives a tractable characterisation of the maximal lazy λ-theory. By co-induction, it includes every semi lazy relation which is closed under (μ_0) and (ξ). For instance, in this fashion it is easy to show that it includes β-conversion, and that it relates all terms of order ∞.

[3] This can be viewed as the whnf analogue of the hnf-based notion of sensible λ-theories if one notes that every sensible λ-theory is also semi sensible, because the maximal sensible λ-theory is semi sensible. Ong's *fully lazy* λ-theories [19] are the subclass of the lazy λ-theories considered here which equate all terms of order ∞.

The principle of applicative bisimulation is not specific to the maximal lazy λ-theory, e.g., the maximal sensible λ-theory can also be described co-inductively in a similar way [16]. However, it seems that the proofs and co-induction principles one obtains in this fashion follow as easily from the tree theories. In contrast, certain results are much simpler to prove for trees, e.g., the equality of the fixed point combinators Y and Θ.

Is there a tree characterisation of the maximal lazy λ-theory as we have for the maximal sensible λ-theory? The answer appears to be no. [4] But in analogy with the 'raw' Böhm tree theory, Ong [19] has developed another lazy λ-theory, the Lévy–Longo tree λ-theory, which is based on a notion of tree introduced by Longo [17]. The Lévy–Longo tree of a term M is the possibly infinite labelled tree, $\mathrm{LT}(M)$, defined as follows.

(1) $\mathrm{LT}(M) = \top, \quad \text{if } M \text{ has order } \infty$

(2) $\mathrm{LT}(M) = \lambda\vec{x}.\,\bot, \quad \text{if } M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\,N$ for some N without whnf

(3) $$\mathrm{LT}(M) = \begin{array}{c} \lambda\vec{x}.\,y \\ \swarrow \quad \cdots \quad \searrow \\ \mathrm{LT}(M_1) \qquad\quad \mathrm{LT}(M_n) \end{array}, \quad \text{if } M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\,y\,M_1 \ldots M_n$$

This definition in terms of head reduction and hnf's is easy to relate to Böhm trees: $\mathrm{BT}(M)$ can be obtained from $\mathrm{LT}(M)$ by replacing every $\top$ or $\lambda\vec{x}.\,\bot$ leaf by $\bot$.

Equality between Lévy–Longo trees, $\mathrm{LT}(M) = \mathrm{LT}(N)$, induces an equivalence relation on terms called *Lévy–Longo tree equivalence*, written $M \approx_{\mathrm{L}} N$. It is included in Böhm tree equivalence because of the relationship between Lévy–Longo trees and Böhm trees. It is lazy, because a term M has no whnf if and only if $\mathrm{LT}(M) = \bot$, and it is a λ-theory. Once again, the difficult part of the proof is showing that it is compatible. Ong [19] proves this by an extension of Barendregt's proof of compatibility for Böhm tree equivalence. Sangiorgi [21] outlines a proof via an encoding into the π-calculus.

The Lévy–Longo tree λ-theory provides the benefits of intensional tree representations for reasoning about equality, e.g., it is easy to show that fixed point combinators have identical Lévy–Longo trees. As it is included in the maximal lazy λ-theory, this is a valid method for proving equalities in the maximal lazy λ-theory.

There are a number of independent characterisations of Lévy–Longo tree equivalence. For instance, it is the induced equivalence relation resulting from Milner's translation of λ-terms into the π-calculus [21], from Plotkin's call-by-name CPS transform on λ-terms [5], and from embedding into a concurrent λ-calculus [7].

[4] At least for the 'intensional' or 'analytical' style of Böhm trees. Gordon's labelled transition system presentation of applicative bisimulation [9] can be viewed as an 'extensional' or 'synthetic' tree characterisation.

3 Bisimulation

In this section, the three tree equivalences from the previous section are formulated as bisimulation equivalences, without explicit reference to trees. This co-inductive formulation will form the basis for the operationally-based development of the theories in the remainder of the paper. The bisimulation characterisation of Lévy–Longo tree equivalence is due to Sangiorgi [21] and has served as inspiration for the two Böhm tree bisimulation equivalences. The three forms of bisimulation are related to syntactic orderings in [13,19].

In Section 2.2 we mentioned two examples of equivalences between infinite trees, $\mathrm{BT}(\mathsf{Y}) = \mathrm{BT}(\Theta)$ and $\mathrm{BT}(x) =_\eta \mathrm{BT}(J\,x)$, where $\mathrm{BT}(\mathsf{Y})$, $\mathrm{BT}(\Theta)$, and $\mathrm{BT}(J\,x)$ are all infinite Böhm trees. The formal proofs of these facts are by co-induction, because the definition of Böhm trees and the definition of $=_\eta$ are co-inductive.

To prove the first equation, $\mathrm{BT}(\mathsf{Y}) = \mathrm{BT}(\Theta)$, we must find a suitable 'bisimulation' relationship between $\mathrm{BT}(\mathsf{Y})$ and $\mathrm{BT}(\Theta)$. First, observe that $\mathrm{BT}(\mathsf{Y})$ and $\mathrm{BT}(\Theta)$ have root nodes labelled $\lambda f.\, f$ and sub-trees $\mathrm{BT}(\Delta\,\Delta)$ and $\mathrm{BT}(\Theta\, f)$, respectively, and these have root nodes labelled f and each recurs as its own sub-tree. We see that the smallest relation $\mathcal{R}$ such that $\mathrm{BT}(\mathsf{Y})\ \mathcal{R}\ \mathrm{BT}(\Theta)$ and $\mathrm{BT}(\Delta\,\Delta)\ \mathcal{R}\ \mathrm{BT}(\Theta\, f)$ is a *bisimulation*, viz. $\mathcal{R}$ only relates trees BT and BT$'$ with identically labelled root nodes and with sub-trees pairwise related by $\mathcal{R}$. The co-induction proof principle asserts that every such bisimulation relation is reflexive. Hence $\mathrm{BT}(\mathsf{Y}) = \mathrm{BT}(\Theta)$.

In fact, this bisimulation proof of $\mathsf{Y} \eqsim_{\mathrm{B}} \Theta$ is more conveniently expressed just at the level of terms rather than their trees. Böhm tree equivalence can be characterised co-inductively, without reference to trees, as follows. Let a *hnf bisimulation* be a relation between terms R such that $M\ R\ N$ implies

(hnf.1) M and N have no hnf, or

(hnf.2) $M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, M_1 \ldots M_n$ and $N \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, N_1 \ldots N_n$ for some $\vec{x}$, y, $M_1 \ldots M_n$, and $N_1 \ldots N_n$ such that $M_i\ R\ N_i$ for $1 \le i \le n$.

Böhm tree equivalence is the greatest hnf bisimulation.

In a similar fashion we can go from the co-inductive definition of Böhm tree equality up to possibly infinite η-expansion, $=_\eta$, to a direct co-inductive definition of Böhm tree equivalence up to η, $\eqsim_{\mathrm{B}\eta}$, without reference to Böhm trees. Let a *hnf bisimulation up to* η be a relation between terms R such that $M\ R\ N$ implies (hnf.1) or

(hnf.η2) $M \twoheadrightarrow_{\mathrm{h}} \lambda x_1 \ldots x_p.\, y\, M_1 \ldots M_m$ and $N \twoheadrightarrow_{\mathrm{h}} \lambda x_1 \ldots x_{p+q}.\, y\, N_1 \ldots N_{m+q}$ for some $x_1 \ldots x_{p+q}$, y, $M_1 \ldots M_m$, and $N_1 \ldots N_{m+q}$ such that $x_{p+1} \ldots x_{p+q}$ are not free in $y\, M_1 \ldots M_m$ and $M_i\ R\ N_i$ for $1 \le i \le m$ and $x_{p+j}\ R\ N_{m+j}$ for $1 \le j \le q$, or

(hnf.2η) the symmetrical case where M reduces to more leading λ's than N.

Böhm tree equivalence up to η is the greatest hnf bisimulation up to η.

Now, we can prove $x \precsim_{\mathrm{B}\eta} J\,x$ by exhibiting a hnf bisimulation up to η, R, such that $x\ R\ (J\,x)$. Since $x \twoheadrightarrow_{\mathrm{h}} x$ and $J\,x \twoheadrightarrow_{\mathrm{h}} \lambda x'.\, x\,(J\,x')$, we let R be the smallest relation such that $x\ R\ (J\,x)$ for every variable x.

For Lévy–Longo trees we can proceed as for Böhm trees to obtain a bisimulation characterisation of Lévy–Longo tree equivalence based on head reduction and hnf's in accordance with the definition of Lévy–Longo trees. Sangiorgi [21] has given an alternative bisimulation characterisation, based on weak head reduction to whnf, which seems more natural for a lazy equivalence. Let a *whnf bisimulation* be a relation R such that $M\ R\ N$ implies

(whnf.1) M and N have no whnf, or

(whnf.2) $M \twoheadrightarrow_{\mathrm{wh}} \lambda x.\, P$ and $N \twoheadrightarrow_{\mathrm{wh}} \lambda x.\, Q$ for some x, P, Q such that $P\ R\ Q$, or

(whnf.3) $M \twoheadrightarrow_{\mathrm{wh}} y\, M_1 \ldots M_n$ and $N \twoheadrightarrow_{\mathrm{wh}} y\, N_1 \ldots N_n$ for some y, $M_1 \ldots M_n$, and $N_1 \ldots N_n$ such that $M_i\ R\ N_i$ for $1 \leq i \leq n$.

Lévy–Longo tree equivalence is the greatest whnf bisimulation.

There are various tree pre-orders in the literature. They can be expressed as simulation orderings with associated simulation co-induction principles. For instance, let a *hnf simulation* be defined like a hnf bisimulation but with (hnf.1) weakened to:

(hnf.1') M has no hnf.

Let *Böhm tree pre-order*, $\sqsubseteq_{\mathrm{B}}$, be the greatest hnf simulation.

4 Compatibility

This section outlines the compatibility proofs for the three tree equivalences, based on the bisimulation formulations from the previous section.

First, we introduce an important relational construction, the *(substitutive) context closure*, R^{SC} of a relation R, defined inductively as the smallest relation closed under the rules:

$$(\text{sc}.R)\ \frac{M\ R\ N}{M\ R^{\mathsf{SC}}\ N} \qquad (\text{sc.var})\ \ x\ R^{\mathsf{SC}}\ x \qquad (\text{sc}.\xi)\ \frac{M\ R^{\mathsf{SC}}\ N}{\lambda x.\, M\ \ R^{\mathsf{SC}}\ \ \lambda x.\, N}$$

$$(\text{sc}.\mu\nu)\ \frac{M\ R^{\mathsf{SC}}\ N \quad P\ R^{\mathsf{SC}}\ Q}{M\,P\ \ R^{\mathsf{SC}}\ \ N\,Q} \qquad (\text{sc.subst})\ \frac{M\ R^{\mathsf{SC}}\ N \quad P\ R^{\mathsf{SC}}\ Q}{M[P/x]\ \ R^{\mathsf{SC}}\ \ N[Q/x]}$$

By definition, R^{SC} is the smallest relation such that (1) R^{SC} includes R, by (sc.R); (2) R^{SC} is compatible, by (sc.var), (sc.ξ) and (sc.$\mu\nu$) (one may check that closure under these three rules implies that R is reflexive); and (3) R^{SC} is closed under (sc.subst). Property (3) is called *substitutivity.*

Lemma 4.1 (Main Lemma) *If R is a hnf bisimulation, so is R^{SC}.*

Proof outline. In order to show that R^{SC} is a hnf simulation, by symmetry it suffices to show that whenever $M\ R^{\mathsf{SC}}\ N$ and $M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, M_1 \dots M_n$, there exist $N_1 \dots N_n$ such that $N \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, N_1 \dots N_n$ and $M_i\ R^{\mathsf{SC}}\ N_i$ for $1 \leq i \leq n$. The proof is by nested inductions: at the top level, by induction on the number of reduction steps p in the head reduction sequence $M \to_{\mathrm{h}}^{p} \lambda\vec{x}.\, y\, M_1 \dots M_n$ and, for each p, by induction on the derivation of the judgment $M\ R^{\mathsf{SC}}\ N$ according to the inductive definition of R^{SC}.

The key properties of head reduction that are needed in the proof are conveniently expressed as:

(1) $M \to_{\mathrm{h}}^{p} H$ holds iff it is derivable from the axiom $x \to_{\mathrm{h}}^{0} x$ and the rules:

$$\frac{M \to_{\mathrm{h}}^{m} H}{\lambda x.\, M \to_{\mathrm{h}}^{m} \lambda x.\, H} \qquad \frac{M \to_{\mathrm{h}}^{m} \lambda x.\, H' \quad H'[{}^{N}\!/\!_{x}] \to_{\mathrm{h}}^{p} H}{M\, N \to_{\mathrm{h}}^{m+1+p} H} \qquad \frac{M \to_{\mathrm{h}}^{m} y\, \vec{P}}{M\, N \to_{\mathrm{h}}^{m} y\, \vec{P}\, N}$$

(2) $M[{}^{N}\!/\!_{x}] \to_{\mathrm{h}}^{p} H$ iff there exist m, n and H' such that $p = m + n$, $M \to_{\mathrm{h}}^{m} H'$ and $H'[{}^{N}\!/\!_{x}] \to_{\mathrm{h}}^{n} H$.

The proof of these facts is not entirely straightforward. The backward implication of (1) is proved by induction on derivations. The forward implication is proved simultaneously with (2) by induction on p.

The full proof appears in the appendix. □

Theorem 4.2 *Böhm tree equivalence is compatible.*

Proof. By the Main Lemma and co-induction, $\eqsim_{\mathrm{B}}^{\mathsf{SC}} \subseteq \eqsim_{\mathrm{B}}$. The reverse inclusion holds by (sc.R). The conclusion follows from the compatibility of R^{SC}. □

The Main Lemma and Theorem 4.2 carry over to hnf bisimulation up to η and Böhm tree equivalence up to η by essentially the same proofs (but there are several extra cases to consider in the induction proof of the Main Lemma).

For whnf bisimulation and Lévy–Longo tree equivalence the proofs proceed analogously to the case for hnf bisimulation and Böhm tree equivalence. In the Main Lemma the results about head reduction to hnf need to be suitably adapted to weak head reduction to whnf (and the proofs are easier).

The proofs of the Main Lemma for hnf bisimulation up to η and for whnf bisimulation can be found in the appendix.

The compatibility proof can also be adapted to the tree pre-orders in the literature. For instance, the compatibility of the Böhm tree pre-order mentioned at the end of Section 3 is immediate from the compatibility proof for Böhm tree equivalence.

5 Bisimulation up to context

This section introduces a refinement of the bisimulation proof method, called bisimulation up to context. It is used to give a simple syntactic proof of the least fixed point property of fixed point combinators.

In bisimulation proofs, in general we want the bisimulations that we need to exhibit to be as small as possible to minimise the task of verifying that it is a bisimulation. To this end several refinements of bisimulation have been introduced (see [22]), including bisimulation up to context. For instance, the definition of a *hnf bisimulation up to context* R relaxes (hnf.2) in the definition of hnf bisimulation to:

(hnf.ctx.2) $M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, M_1 \ldots M_n$ and $N \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\, N_1 \ldots N_n$ for some $\vec{x}$, y, $M_1 \ldots M_n$, and $N_1 \ldots N_n$ such that $M_i\ R^{\mathsf{SC}}\ N_i$ for $1 \leq i \leq n$.

That is, for terms related by R that have hnf's, we only require that the immediate sub-terms of their hnf's are related up to the context closure of R rather than R itself. We wish to establish an associated proof rule which asserts that every hnf bisimulation up to context is included in the greatest hnf bisimulation, $\eqsim_{\mathrm{B}}$.

For process calculus bisimulations and for applicative bisimulation, bisimulation up to context proof rules substantially simplify some proofs about recursion [22,20,15]. But for higher-order calculi, it is an open problem whether the proof rule is valid in the general form of (hnf.ctx.2). Here, we can solve the problem for all the forms of bisimulation and simulation considered in this paper. Revisiting the proof of the Main Lemma, one finds that wherever one uses the assumption that R is a bisimulation, the weaker assumption that R is a bisimulation up to context would suffice. That is, the Main Lemma can be strengthened to:

Lemma 5.1 *If R is a hnf bisimulation up to context, R^{SC} is a hnf bisimulation.*

As a corollary we get that every hnf bisimulation up to context R is included in Böhm tree equivalence: by the lemma and by co-induction, we get that $R^{\mathsf{SC}} \subseteq \eqsim_{\mathrm{B}}$ and then $R \subseteq \eqsim_{\mathrm{B}}$ follows by (sc.R).

Similarly for the other forms of bisimulation and simulation.

We will illustrate the proof principle by a syntactic proof of recursion induction, the least (pre-)fixed point property of fixed point combinators. We consider hnf simulation and introduce a further refinement of simulation up to context: a relation R is a *hnf simulation up to context and up to* $\precsim_{\mathrm{B}}$ if, whenever $M\ R\ Q$, either (hnf.1') or (hnf.ctx.2) holds for some $N \precsim_{\mathrm{B}} Q$. Again, one can strengthen the Main Lemma to show that for any such R, the composed relation $R^{\mathsf{SC}} \precsim_{\mathrm{B}}$ is a hnf simulation and, hence, $R \subseteq \precsim_{\mathrm{B}}$.

Proposition 5.2 (Recursion induction) *If $M\, N \precsim_{\mathrm{B}} N$ then $\Theta\, M \precsim_{\mathrm{B}} N$.*

Proof. Let R be the singleton relation between $\Theta\, M$ and N.

If M has a λ-free hnf H, the reflexive closure of R is a simulation up to $\precsim_{\mathrm{B}}$, because $\Theta\, M \twoheadrightarrow_{\mathrm{h}} H\,(\Theta\, M)$ and $M\, N \twoheadrightarrow_{\mathrm{h}} H\, N$, and $H\,(\Theta\, M)$ and $H\, N$ are hnf's with the arguments to the head variable pairwise identical or related by R. For identical terms, related by the reflexive closure, (hnf.1') or (hnf.2) follows easily.

If M has hnf $\lambda f.\, H$ with head variable different from f, we prove that R is a hnf simulation up to context and up to $\precsim_{\mathrm{B}}$. Since $M\,N \precsim_{\mathrm{B}} N$, it suffices to show that $\Theta\, M$ and $M\,N$ satisfy (hnf.ctx.2). But $\Theta\, M \twoheadrightarrow_{\mathrm{h}} H[{}^{\Theta\, M}\!/\!_f]$ and $M\,N \twoheadrightarrow_{\mathrm{h}} H[{}^{N}\!/\!_f]$, and the result follows by inspection of the hnf's $H[{}^{\Theta\, M}\!/\!_f]$ and $H[{}^{N}\!/\!_f]$.

If M has hnf $\lambda f.\, H$ with head variable f or if M has no hnf, one can derive that $\Theta\, M$ has no hnf. Hence $\Theta\, M \precsim_{\mathrm{B}} N$. □

A Proofs of the Main Lemma

In this appendix the Main Lemma of the compatibility proofs for the different tree pre-orders and equivalences is proved for hnf simulation, for hnf simulation up to η, and for whnf simulation. The proofs exploit some characterisations of reduction to hnf and to whnf as inductively defined big-step evaluation relations between (possibly open) terms and their hnf's and whnf's, respectively. For whnf's the evaluation relation is closely related to Goguen's evaluation relations for capturing strong normalisation [8]. The evaluation characterisation of reduction to hnf appears to be new. Honsell and Lenisa [11] have another big-step characterisation of reduction to hnf but it seems less useful for our proofs of the Main Lemma.

Throughout the appendix, L, M, N, P and Q range over terms, G and H range over hnf's, and V and W range over whnf's.

A.1 Head reduction

Recall the definition of the head reduction relation:

$$\text{(red.h)}\quad \lambda\vec{x}.\,(\lambda y.\, M)\, N\, \vec{P} \rightarrow_{\mathrm{h}} \lambda\vec{x}.\, M[{}^{N}\!/\!_y]\, \vec{P}$$

Head reduction can be characterised as "left-most, outer-most reduction to hnf", that is, a head redex is the left-most, outer-most redex in a non-hnf.

Head reduction can be specified as a small-step structural operational semantics, given by the structurally inductive rules in Table A.1. It is easy to check, for all terms M and N, that $M \rightarrow_{\mathrm{h}} N$ can be derived from the inductive rules if and only if $M \rightarrow_{\mathrm{h}} N$ is an instance of (red.h).

Next, we introduce a big-step evaluation semantics of reduction to hnf which is convenient in the proof of the Main Lemma for the different Böhm tree pre-orders and equivalences. It is specified as a ternary relation, $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$, between terms M, non-negative integers c, and hnf's H. The integer c counts the number of β-reductions in the derivation. The relation is defined inductively by the rules in Table A.2.

A binary relation between terms M and hnf's H is obtained by erasing the integer superscripts from the rules, or we can simply define $M \rightsquigarrow_{\mathrm{h}} H$ to mean that $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ for some c.

$$(\text{red.}\beta) \qquad (\lambda y.\, M)\, N \rightarrow_{\mathrm{h}} M[{}^{N}\!/\!{}_{y}]$$

$$(\text{red.}\mu.@) \qquad \frac{M_1\, M_2 \rightarrow_{\mathrm{h}} M'}{M_1\, M_2\, N \rightarrow_{\mathrm{h}} M'\, N}$$

$$(\text{red.}\xi) \qquad \frac{M \rightarrow_{\mathrm{h}} M'}{\lambda y.\, M \rightarrow_{\mathrm{h}} \lambda y.\, M'}$$

Table A.1
Head reduction as a small-step structural operational semantics

$$(\text{eval.var}) \qquad x \overset{0}{\leadsto}_{\mathrm{h}} x$$

$$(\text{eval.}\xi) \qquad \frac{M \overset{c}{\leadsto}_{\mathrm{h}} H}{\lambda x.\, M \overset{c}{\leadsto}_{\mathrm{h}} \lambda x.\, H}$$

$$(\text{eval.}\beta) \qquad \frac{M \overset{c_1}{\leadsto}_{\mathrm{h}} \lambda x.\, G \quad G[{}^{N}\!/\!{}_{x}] \overset{c_2}{\leadsto}_{\mathrm{h}} H}{M\, N \overset{c_1+c_2+1}{\leadsto}_{\mathrm{h}} H}$$

$$(\text{eval.}\mu.\text{var}@) \qquad \frac{M \overset{c}{\leadsto}_{\mathrm{h}} y\, \vec{M}}{M\, N \overset{c}{\leadsto}_{\mathrm{h}} y\, \vec{M}\, N}$$

Table A.2
Head reduction to hnf as a big-step evaluation semantics

The hnf meta-variables G and H are used in the rules to indicate where terms are in hnf; if we let G and H in the rules range over arbitrary terms, we can prove that $M \leadsto_{\mathrm{h}} H$ implies that H is a hnf, for all terms M and H, by induction on the derivation of $M \leadsto_{\mathrm{h}} H$. Furthermore, all hnf's evaluate to themselves (so the evaluation relation is idempotent, cf. [12]).

We are going to prove that head evaluation corresponds to head reduction:

$$M \overset{c}{\leadsto}_{\mathrm{h}} H \quad \text{if and only if} \quad M \rightarrow^{c}_{\mathrm{h}} H \tag{A.1}$$

This is a bit tricky because the reduction strategy implicit in the evaluation rules is left-most, inner-most: the operator in a function application is evaluated to hnf—even if it is a λ-abstraction—before the application redex is contracted. In contrast, the left-most, outer-most strategy of head reduction contracts the application redex as soon as the operator reaches a λ-abstraction, rather than first reduce the body of the λ-abstraction to hnf.

We are going to need the following property of head reduction to hnf and substitution:

Lemma A.1 *If* $M \to_{\mathrm{h}}^{c_1} G$ *and* $G[N/x] \to_{\mathrm{h}}^{c_2} H$ *then* $M[N/x] \to_{\mathrm{h}}^{c_1+c_2} H$.

Proof. First, we note by inspection of the (red.h) rule that the head reduction relation is closed under substitutions:

$$M \to_{\mathrm{h}} M' \quad \Rightarrow \quad M[N/x] \to_{\mathrm{h}} M'[N/x] \tag{A.2}$$

Using this fact c_1 times we find that $M[N/x] \to_{\mathrm{h}}^{c_1} G[N/x]$. The result follows by composing this reduction sequence with $G[N/x] \to_{\mathrm{h}}^{c_2} H$. □

One direction of the correspondence between evaluation and reduction to hnf, (A.1), is straightforward by induction on derivations:

Lemma A.2 *If* $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ *then* $M \to_{\mathrm{h}}^{c} H$.

Proof. By induction on the derivation of $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$.

Case (eval.var) Immediate because variables are in hnf.

Case (eval.ξ) Easy using the induction hypothesis and (red.ξ).

Case (eval.β) Here $M = M_1\, M_2$, $M_1 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} \lambda x.\, G$, $G[M_2/x] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$, and $c = c_1 + 1 + c_2$. By the induction hypothesis,

$$M_1 \to_{\mathrm{h}}^{c_1} \lambda x.\, G \tag{A.3}$$
$$G[M_2/x] \to_{\mathrm{h}}^{c_2} H \tag{A.4}$$

Let $\lambda x.\, N$ be the first λ-abstraction in the reduction sequence (A.3), that is,

$$M_1 \to_{\mathrm{h}}^{c_{11}} \lambda x.\, N \tag{A.5}$$
$$\lambda x.\, N \to_{\mathrm{h}}^{c_{12}} \lambda x.\, G \tag{A.6}$$

with $c_1 = c_{11} + c_{12}$ and all terms in the first reduction sequence (A.5) other than L are not λ-abstractions; as they are clearly also not variables, they must be applications, so we get $M_1\, M_2 \to_{\mathrm{h}}^{c_{11}} (\lambda x.\, N)\, M_2$ by c_{11} uses of (red.μ.@). Furthermore, by (eval.β), $(\lambda x.\, N)\, M_2 \to_{\mathrm{h}} N[M_2/x]$. We observe that (A.6) must be derived from $N \to_{\mathrm{h}}^{c_{12}} G$ by (redh.ξ). Hence, by (A.4) and Lemma A.1, $N[M_2/x] \to_{\mathrm{h}}^{c_{12}+c_2} H$. Finally, we compose the reduction sequences:

$$M = M_1\, M_2 \to_{\mathrm{h}}^{c_{11}} (\lambda x.\, N)\, M_2 \to_{\mathrm{h}} N[M_2/x] \to_{\mathrm{h}}^{c_{12}+c_2} H$$

which is the desired result because $c_{11} + 1 + c_{12} + c_2 = c$.

Case (eval.μ.var@) Here $M = M_1\, M_2$, $M_1 \overset{c}{\rightsquigarrow}_{\mathrm{h}} y\, \vec{N}$, and $H = y\, \vec{N} M_2$. By the induction hypothesis, $M_1 \to_{\mathrm{h}}^{c} y\, \vec{N}$. It is easy to check that, whenever $P \to_{\mathrm{h}} Q$ and Q is not a λ-abstraction, P is an application. Since $y\, \vec{N}$ is not a λ-abstraction all preceding terms in the reduction sequence $M_1 \to_{\mathrm{h}}^{c} y\,\vec{N}$ are applications. So, by c uses of (red.μ.@), we get the desired result: $M_1\,M_2 \to_{\mathrm{h}}^{c} y\,\vec{N} M_2$. □

The other direction of the correspondence between evaluation and reduction to hnf is proved simultaneously with a result about substitution and evaluation:

Lemma A.3 (1) *If $M \to_{\mathrm{h}}^{c} H$ then $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$.*

(2) *If $M[{}^{N}\!/x] \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ there exist c_1, c_2 and G such that $c = c_1 + c_2$, $M \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ and $G[{}^{N}\!/x] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$.*

Proof. The proof is by induction on c. As the induction hypothesis we assume that (2) and (1) hold for all $c < c_0$. In the induction step we prove that they hold for $c = c_0$.

First, we prove that (1) holds for $c = c_0$.

If $c_0 = 0$, $M \to_{\mathrm{h}}^{c_0} H$ means that $M = H$, so we must prove that

$$H \overset{0}{\rightsquigarrow}_{\mathrm{h}} H, \quad \text{for all hnf's } H. \tag{A.7}$$

If $H = \lambda x_1 \ldots x_n.\, y\, M_1 \ldots M_m$ say, we derive $H \overset{0}{\rightsquigarrow}_{\mathrm{h}} H$ by (eval.var), m times (eval.μ.var@), and n times (eval.ξ).

If $c_0 > 0$, $M \to_{\mathrm{h}}^{c_0} H$ implies that $M \to_{\mathrm{h}} N$ for some N such that $N \to_{\mathrm{h}}^{c_0 - 1} H$. By the induction hypothesis, $N \overset{c_0-1}{\rightsquigarrow}_{\mathrm{h}} H$. To derive the desired conclusion, $M \overset{c_0}{\rightsquigarrow}_{\mathrm{h}} H$, from $M \to_{\mathrm{h}} N$ and $N \overset{c_0-1}{\rightsquigarrow}_{\mathrm{h}} H$, we prove more generally that

$$M \to_{\mathrm{h}} N \;\Rightarrow\; (\; N \overset{c}{\rightsquigarrow}_{\mathrm{h}} H \;\Rightarrow\; M \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} H \;)$$

for all $c < c_0$, by induction on the derivation of $M \to_{\mathrm{h}} N$.

Case (red.β) $M = (\lambda x.\, M_1)\, M_2$ and $N = M_1[{}^{M_2}\!/x]$. Since $c < c_0$, we can use part (2) of the outer induction hypothesis to obtain c_1, c_2 and H_1 such that $c = c_1 + c_2$, $M_1 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} H_1$ and $H_1[{}^{M_2}\!/x] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$. By (eval.$\xi$), $\lambda x.\, M_1 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} \lambda x.\, H_1$ and, by (eval.β), we conclude $(\lambda x.\, M_1)\, M_2 \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} H$.

Case (red.μ.@) In this case $M \to_{\mathrm{h}} N$ is derived from a premise of the form $M_1\, M_2 \to_{\mathrm{h}} N_1$ where $M = M_1\, M_2\, N_2$ and $N = N_1\, N_2$. Here $N \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ must be derived by (eval.β) or (eval.μ.var@). In either case there are $c_1 + c_2 = c$ and H_1 such that $N_1 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} H_2$ and $H_1\, N_2 \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$. By the induction hypothesis, $M_1\, M_2 \overset{c_1+1}{\rightsquigarrow}_{\mathrm{h}} H_2$. From this and $H_1\, N_2 \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$ we derive, by (eval.β) or (eval.μ.var@), that $M_1\, M_2\, N_2 \overset{c_1+1+c_2}{\rightsquigarrow}_{\mathrm{h}} H$, that is, $M \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} H$.

Case (red.ξ) Here $M \to_{\mathrm{h}} N$ is derived from a premise $M' \to_{\mathrm{h}} N'$, where $M = \lambda x.\, M'$ and $N = \lambda x.\, N'$. We observe that $N \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ must be derived from $N' \overset{c}{\rightsquigarrow}_{\mathrm{h}} H'$ where $\lambda x. H' = H$. By the induction hypothesis, $M' \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} H'$ and, by (eval.ξ), we conclude that $M \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} H$.

This concludes part (1) of the induction step.

In part (2) of the induction step we prove, more generally, that (2) holds for all $c \leq c_0$, by induction on the derivation of $M[{}^{N}\!/\!_{x}] \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$. If $M = x$ the result is immediate by taking $c_1 = 0$ and $G = x$. Otherwise we proceed by analysis of the derivation of $M[{}^{N}\!/\!_{x}] \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$.

Case (eval.var) Here $M[{}^{N}\!/\!_{x}]$ is some variable, y say. Since $M \neq x$, $M = y$ and we let $c_1 = c_2 = 0$ and $G = y$.

Case (eval.ξ) Here $M[{}^{N}\!/\!_{x}] = \lambda y. P'$, $H = \lambda y. H'$, and $M[{}^{N}\!/\!_{x}] \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ is derived from $P' \overset{c}{\rightsquigarrow}_{\mathrm{h}} H'$. Since $M \neq x$, $M = \lambda y. M'$ for some M' where $M'[{}^{N}\!/\!_{x}] = P'$. By the induction hypothesis, there are c_1, c_2 and G' such that $c = c_1 + c_2$, $M' \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G'$ and $G'[{}^{N}\!/\!_{x}] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H'$. Letting $G = \lambda y. G'$ we get $M \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ and $G[{}^{N}\!/\!_{x}] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$ by (eval.ξ).

Case (eval.β) Here $M[{}^{N}\!/\!_{x}] = P' \, P''$ and $M[{}^{N}\!/\!_{x}] \overset{c}{\rightsquigarrow}_{\mathrm{h}} H$ is derived from $P' \overset{c'}{\rightsquigarrow}_{\mathrm{h}} \lambda y. H''$ and $H''[{}^{P''}\!/\!_{y}] \overset{c''}{\rightsquigarrow}_{\mathrm{h}} H$ where $c = c' + c'' + 1$. Since $M \neq x$, M is of the form $M = M' \, M''$ with $M'[{}^{N}\!/\!_{x}] = P'$ and $M''[{}^{N}\!/\!_{x}] = P''$. By the induction hypothesis there are c_1', c_2' and G' such that $c' = c_1' + c_2'$, $M' \overset{c_1'}{\rightsquigarrow}_{\mathrm{h}} G'$ and $G'[{}^{N}\!/\!_{x}] \overset{c_2'}{\rightsquigarrow}_{\mathrm{h}} \lambda y. H''$.

If G' is λ-free, let $c_1 = c_1'$, $c_2 = c_2' + c'' + 1$, and $G = G' \, M''$. From $M' \overset{c_1'}{\rightsquigarrow}_{\mathrm{h}} G'$ we derive $M \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ by (eval.μ.var@), and from $G'[{}^{N}\!/\!_{x}] \overset{c_2'}{\rightsquigarrow}_{\mathrm{h}} \lambda y. H''$ and $H''[{}^{P''}\!/\!_{y}] \overset{c''}{\rightsquigarrow}_{\mathrm{h}} H$ we derive $G[{}^{N}\!/\!_{x}] = G'[{}^{N}\!/\!_{x}] \, P'' \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$ by (eval.β).

If G' is a λ-abstraction, $G' = \lambda y. G''$ for some hnf G''. In this case $G'[{}^{N}\!/\!_{x}] \overset{c_2'}{\rightsquigarrow}_{\mathrm{h}} \lambda y. H''$ must be derived by (eval.ξ) from $G''[{}^{N}\!/\!_{x}] \overset{c_2'}{\rightsquigarrow}_{\mathrm{h}} H''$. We split the argument into two cases according to whether x is the head variable of G''.

If x is the head variable of G'', $G''[{}^{M''}\!/\!_{y}]$ is a hnf. Let $G = G''[{}^{M''}\!/\!_{y}]$, $c_1 = c_1' + 1$ and $c_2 = c_2' + c''$. From (A.7) we know that $G \overset{0}{\rightsquigarrow}_{\mathrm{h}} G$. As $M' \overset{c_1'}{\rightsquigarrow}_{\mathrm{h}} G' = \lambda y. G''$ we derive $M = M' \, M'' \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ by (eval.β). Observing that

$$G[{}^{N}\!/\!_{x}] = (G''[{}^{M''}\!/\!_{y}])[{}^{N}\!/\!_{x}] = (G''[{}^{N}\!/\!_{x}])[{}^{M''[{}^{N}\!/\!_{x}]}\!/\!_{y}] = (G''[{}^{N}\!/\!_{x}])[{}^{P''}\!/\!_{y}]$$

we use Lemma A.2 and Lemma A.1 to deduce $G[{}^{N}\!/\!_{x}] \to_{\mathrm{h}}^{c_2} H$ from $G''[{}^{N}\!/\!_{x}] \overset{c_2'}{\rightsquigarrow}_{\mathrm{h}} H''$ and $H''[{}^{P''}\!/\!_{y}] \overset{c''}{\rightsquigarrow}_{\mathrm{h}} H$. Since $c_2 \leq c'' < c \leq c_0$, we conclude $G[{}^{N}\!/\!_{x}] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$ by part (1) of the outer induction hypothesis.

If x is not the head variable of G'', $G''[{}^{N}\!/\!_{x}]$ is a hnf, so $G''[{}^{N}\!/\!_{x}] = H''$, $c_2' = 0$ and $c_1' = c'$. Noting that $H''[{}^{P''}\!/\!_{y}] = G''[{}^{M''}\!/\!_{y}][{}^{N}\!/\!_{x}]$ we get from the induction hypothesis applied to $H''[{}^{P''}\!/\!_{y}] \overset{c''}{\rightsquigarrow}_{\mathrm{h}} H$ that there are c_1'', c_2 and G

such that $G''[{}^{M''}\!/\!_y] \overset{c_1''}{\rightsquigarrow}_{\mathrm{h}} G$ and $G[{}^{N}\!/\!_x] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} H$. Let $c_1 = c' + c_1'' + 1$. Finally, from $M' \overset{c'}{\rightsquigarrow}_{\mathrm{h}} \lambda y.\, G''$ and $G''[{}^{M''}\!/\!_y] \overset{c_1''}{\rightsquigarrow}_{\mathrm{h}} G$ we derive $M \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ by (eval.β).

Case (eval.μ.var@) Here $M[{}^{N}\!/\!_x] = P\,Q$, $P \overset{c}{\rightsquigarrow}_{\mathrm{h}} y\,\vec{P}$ and $H = y\,\vec{P}\,Q$ for some P, Q, y and $\vec{P}$. Since $M \neq x$, M is of the form $M = M_1\,M_2$ with $M_1[{}^{N}\!/\!_x] = P$ and $M_2[{}^{N}\!/\!_x] = Q$. By the induction hypothesis there exist c_1, c_2, and a hnf G_1 such that $c = c_1 + c_2$, $M_1 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G_1$ and $G_1[{}^{N}\!/\!_x] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} y\,\vec{P}$. Clearly, G_1 is a λ-free hnf. We let $G = G_1\,M_2$. By (eval.μ.var@) both $M = M_1\,M_2 \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G$ and $G[{}^{N}\!/\!_x] = G_1[{}^{N}\!/\!_x]\,Q \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} y\,\vec{P}\,Q = H$, as required.

This concludes part (2) of the induction step. $\square$

The next lemma lists some facts about evaluation that we need in the proof of the Main Lemma for the different Böhm tree pre-orders and equivalences.

Lemma A.4 (1) *$M[{}^{N}\!/\!_x] \to_{\mathrm{h}}^{c} H$ if and only if there exist c_1, c_2 and G such that $c = c_1 + c_2$, $M \to_{\mathrm{h}}^{c_1} G$ and $G[{}^{N}\!/\!_x] \to_{\mathrm{h}}^{c_2} H$.*

(2) *$\lambda\vec{x}.\, M \to_{\mathrm{h}}^{c} H$ if and only if there exists G such that $M \to_{\mathrm{h}}^{c} G$ and $H = \lambda\vec{x}.\, G$.*

(3) *$M\,\vec{N} \to_{\mathrm{h}}^{c} H$ if and only if there exist c_1, c_2 and G such that $c = c_1 + c_2$, $M \to_{\mathrm{h}}^{c_1} G$ and $G\,\vec{N} \to_{\mathrm{h}}^{c_2} H$.*

(4) *$\lambda\vec{x}.\, (\lambda y.\, M)\,N\,\vec{P} \to_{\mathrm{h}}^{c} H$ if and only if $c > 0$ and $\lambda\vec{x}.\, M[{}^{N}\!/\!_x]\,\vec{P} \to_{\mathrm{h}}^{c-1} H$.*

Proof. The forward implication of part (1) follows from Lemma A.3(2) and (A.1). The reverse implication is just Lemma A.1.

Parts (2) and (3) are proved by induction on the length of $\vec{x}$ and $\vec{N}$, respectively. The proofs exploit the correspondence between evaluation and reduction to hnf, and the structural definition of the evaluation relation.

Part (4) is immediate from the definition of $\to_{\mathrm{h}}$ and determinacy. $\square$

A.2 Proof of Main Lemma for hnf simulation

We are now going to prove the Main Lemma for hnf simulation. Then the Main Lemma for hnf bisimulation follows by symmetry.

The following is a helpful notation:

$$M \left\langle R \right\rangle_{\mathrm{h}} N \overset{\mathrm{def}}{\Leftrightarrow} \forall \vec{x}, y, \vec{M}.\ M \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\,\vec{M} \ \Rightarrow \ \exists \vec{N}.\ N \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}.\, y\,\vec{N} \;\&\; \vec{M}\ R\ \vec{N}$$

A relation R is a hnf simulation if and only if $R \subseteq \left\langle R \right\rangle_{\mathrm{h}}$. It is a hnf simulation up to context if and only if $R \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{h}}$.

In the following we assume that R is a hnf simulation up to context, $R \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{h}}$, and we must prove that R^{SC} is a hnf simulation, $R^{\mathsf{SC}} \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{h}}$. Expanding the definition of $\left\langle \cdot \right\rangle_{\mathrm{h}}$ and using the correspondence between head reduction and head evaluation, (A.1), this amounts to showing that

$$M \; R^{\mathsf{SC}} \; N \;\Rightarrow\; \forall G.\; M \overset{c}{\leadsto}_{\mathrm{h}} G \;\Rightarrow\; \exists H.\; N \leadsto_{\mathrm{h}} H \;\&\; G \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} H \tag{A.8}$$

where $\left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}}$ denotes the smallest relation between hnf's such that

$$\lambda\vec{x}.\, y\,\vec{M} \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} \lambda\vec{x}.\, y\,\vec{N} \;\overset{\mathrm{def}}{\Leftrightarrow}\; \vec{M} \; R^{\mathsf{SC}} \; \vec{N}$$

for all $\vec{x}$, y, $\vec{M}$ and $\vec{N}$. The proof of (A.8) is by induction (I) on c and (II) on the derivation of $M \; R^{\mathsf{SC}} \; N$, ordered lexicographically. We proceed by analysis of the derivation of $M \; R^{\mathsf{SC}} \; N$.

(sc.R) Immediate from the assumption $R \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{h}}$.

(sc.var) Immediate by (eval.var).

(sc.ξ) In this case, M and N are λ-abstractions. $M \overset{c}{\leadsto}_{\mathrm{h}} G$ must be derived by (eval.ξ), and the conclusion follows easily by part (II) of the induction hypothesis and (eval.ξ).

(sc.$\mu\nu$) Here $M = M'\,M''$ and $N = N'\,N''$, and $M \; R^{\mathsf{SC}} \; N$ is derived from

$$M' \; R^{\mathsf{SC}} \; N' \tag{A.9}$$

$$M'' \; R^{\mathsf{SC}} \; N'' \tag{A.10}$$

Since $M = M'\,M''$, $M \overset{c}{\leadsto}_{\mathrm{h}} G$ must be derived by either (eval.β) or (eval.μ.var@).

In the first case, $M \overset{c}{\leadsto}_{\mathrm{h}} G$ is derived by (eval.β) from premises of the form:

$$M' \overset{c_1}{\leadsto}_{\mathrm{h}} \lambda x.\, G' \tag{A.11}$$

$$G'[M''/x] \overset{c_2}{\leadsto}_{\mathrm{h}} G \tag{A.12}$$

where $c_1 + 1 + c_2 = c$. Hence $c_1, c_2 < c$. By (A.9), (A.11), and either part of the induction hypothesis, there exists an H' such that

$$N' \leadsto_{\mathrm{h}} \lambda x.\, H' \tag{A.13}$$

and $\lambda x.\, G' \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} \lambda x.\, H'$. From the definition of $\left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}}$ and the compatibility of R^{SC}we deduce that $G' \; R^{\mathsf{SC}} \; H'$. By (A.10) and substitutivity of R^{SC}, we have $G'[M''/x] \; R^{\mathsf{SC}} \; H'[N''/x]$. Since $c_2 < c$, (A.12) and part (I) of the induction hypothesis imply that

$$H'[N''/x] \leadsto_{\mathrm{h}} H \tag{A.14}$$

with $G \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} H$. Finally, from (A.13), (A.14) and (eval.β) we conclude that $N \leadsto_{\mathrm{h}} H$, as desired.

In the second case, $M \overset{c}{\rightsquigarrow}_{\mathrm{h}} G$ is derived by (eval.μ.var@) from a premise of the form $M' \overset{c}{\rightsquigarrow}_{\mathrm{h}} y\,\vec{P}$, where $y\,\vec{P}M'' = G$. By (A.9) and part (II) of the induction hypothesis, $N' \rightsquigarrow_{\mathrm{h}} y\,\vec{Q}$ with $\vec{P}\ R^{\mathsf{SC}}\ \vec{Q}$. Let $H = y\,\vec{Q}N''$. Then $N \rightsquigarrow_{\mathrm{h}} H$ follows from (eval.μ.var@), and $G\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H$ holds because $\vec{P}\ R^{\mathsf{SC}}\ \vec{Q}$ and $M''\ R^{\mathsf{SC}}\ N''$.

(sc.subst) Here $M = M'[P/y]$ and $N = N'[Q/y]$, and $M\ R^{\mathsf{SC}}\ N$ is derived from $M'\ R^{\mathsf{SC}}\ N'$ and $P\ R^{\mathsf{SC}}\ Q$.

By Lemma A.4(1), $M'[P/y] \overset{c}{\rightsquigarrow}_{\mathrm{h}} G$ implies that there exist c_1, c_2 and G' such that

$$M' \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G' \tag{A.15}$$

$$G'[P/y] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} G \tag{A.16}$$

and $c = c_1 + c_2$. Hence $c_1, c_2 \leq c$.

By part (II) of the induction hypothesis, $N' \rightsquigarrow_{\mathrm{h}} H'$ with $G'\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H'$. We need to exhibit an H such that

$$H'[Q/y] \rightsquigarrow_{\mathrm{h}} H \quad \&\quad G\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H \tag{A.17}$$

Then the desired conclusion, $N \rightsquigarrow_{\mathrm{h}} H$, follows by Lemma A.4(1).

Since $G'\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H'$, G' and H' are of the form $G' = \lambda\vec{x}.\,z\,\vec{M}$ and $H' = \lambda\vec{x}.\,z\,\vec{N}$ with

$$\vec{M}\ R^{\mathsf{SC}}\ \vec{N} \tag{A.18}$$

If $z \neq y$, $G'[P/y]$ and $H'[Q/y]$ are hnf's and $G = G'[P/y]$ and $H = H'[Q/y]$, and the desired conclusion follows easily using the fact that R^{SC} is substitutive. If $z = y$, (A.16) and Lemma A.4(3) imply that there exist c_{21}, c_{22} and G_P such that

$$P \overset{c_{21}}{\rightsquigarrow}_{\mathrm{h}} G_P \tag{A.19}$$

$$\lambda\vec{x}.\,G_P\,\vec{M}[P/y] \overset{c_{22}}{\rightsquigarrow}_{\mathrm{h}} G \tag{A.20}$$

and $c_2 = c_{21} + c_{22}$. Since $c_{21} \leq c_2 \leq c$, part (II) of the induction hypothesis implies that

$$Q \rightsquigarrow_{\mathrm{h}} H_Q \tag{A.21}$$

with $G_P\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H_Q$.

If $c_{22} = 0$, that is, $\lambda\vec{x}.\,G_P\,\vec{M}[P/y]$ is a hnf, either G_P is a λ-free hnf or $\vec{M}$ is empty. In the first case H_Q is also a λ-free hnf, since $G_P\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H_Q$. In the last case $\vec{N}$ is empty too, since $\vec{M}\ R^{\mathsf{SC}}\ \vec{N}$. In either case, $\lambda\vec{x}.\,H_Q\,\vec{N}[Q/y]$ is a hnf. Let H be this hnf, then we can derive $G\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H$ from $G_P\ \langle R^{\mathsf{SC}} \rangle_{\mathrm{hnf}}\ H_Q$, $\vec{M}\ R^{\mathsf{SC}}\ \vec{N}$, and $P\ R^{\mathsf{SC}}\ Q$ because R^{SC} is compatible and substitutive.

Furthermore, by (A.21) and Lemma A.4(3), $H'[{}^{Q}\!/\!{}_{y}] \rightsquigarrow_{\mathrm{h}} H$, as required to show (A.17).

If $c_{22} > 0$, $\lambda\vec{x}.\, G_P\, \vec{M}[{}^{P}\!/\!{}_{y}]$ is not a hnf, so G_P must be a λ-abstraction, $G_P = \lambda x'.\, G_P{}'$, and we must have that $\vec{M}[{}^{P}\!/\!{}_{y}] = M'_1 \dots M'_n$ for some $n \geq 1$, such that $G_P\, M'_1$ is a β-redex. By Lemma A.4(4),

$$\lambda\vec{x}.\, G_P{}'[{}^{M'_1}\!/\!{}_{x'}]\, M'_2 \dots M'_n \overset{c_{22}-1}{\rightsquigarrow}_{\mathrm{h}} G \tag{A.22}$$

Since $G_P \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} H_Q$, we must have that $H_Q = \lambda x'.\, H_Q{}'$ with $G_P{}' \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} H_Q{}'$. As $\vec{M}\ R^{\mathsf{SC}}\ \vec{N}$ and $P\ R^{\mathsf{SC}}\ Q$, there are $N'_1 \dots N'_n$ such that $\vec{N}[{}^{Q}\!/\!{}_{y}] = N'_1 \dots N'_n$ and $M'_1 \dots M'_n\ R^{\mathsf{SC}}\ N'_1 \dots N'_n$. Furthermore, we calculate that

$$\lambda\vec{x}.\, G_P{}'[{}^{M'_1}\!/\!{}_{x'}]\, M'_2 \dots M'_n \quad R^{\mathsf{SC}} \quad \lambda\vec{x}.\, H_Q{}'[{}^{N'_1}\!/\!{}_{x'}]\, N'_2 \dots N'_n$$

by substitutivity and compatibility of R^{SC}. From (A.22) and part (i) of the induction hypothesis it follows that

$$\lambda\vec{x}.\, H_Q{}'[{}^{N'_1}\!/\!{}_{x'}]\, N'_2 \dots N'_n \rightsquigarrow_{\mathrm{h}} H$$

with $G \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{hnf}} H$. By Lemma A.4(4),

$$\lambda\vec{x}.\, H_Q\, N'_1 \dots N'_n \rightsquigarrow_{\mathrm{h}} H$$

and, by (A.21) and Lemma A.4(3),

$$H'[{}^{Q}\!/\!{}_{y}] \rightsquigarrow_{\mathrm{h}} H$$

thus establishing (A.17), as required.

This concludes the proof of the Main Lemma for hnf simulation.

A.3 Proof of Main Lemma for hnf simulation up to η

The proof of the Main Lemma for hnf simulation up to η proceeds analogously to the preceding proof for hnf simulation. Again, the Main Lemma for hnf bisimulation up to η follows by symmetry.

We define:

$$\begin{aligned} M\ {}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{h}}\ N \ &\overset{\mathrm{def}}{\Leftrightarrow}\ \forall G.\ M \twoheadrightarrow_{\mathrm{h}} G \ \Rightarrow \\ &\qquad\qquad \exists H.\ N \twoheadrightarrow_{\mathrm{h}} H \ \ \&\ \ G\ {}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{hnf}}\ H \end{aligned}$$

where ${}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{hnf}}$ is the restriction of ${}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{h}}$ to hnf's: the smallest relation such that

$$\lambda x_1 \dots x_r.\, y\, \vec{M} \quad {}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{hnf}} \quad \lambda x_1 \dots x_s.\, y\, \vec{N}$$

if $\vec{M} x_{r+1} \dots x_t \ \ R\ \ \vec{N} x_{s+1} \dots x_t$, where $t = \max(r, s)$. A relation R is an η hnf simulation if and only if $R \subseteq {}^{\eta}\!\langle R\rangle^{\eta}_{\mathrm{h}}$.

Under the assumption that R is an η hnf simulation up to contexts, $R \subseteq {}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{h}}$, we prove that R^{SC} is an η hnf simulation, $R^{\mathsf{SC}} \subseteq {}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{h}}$. This amounts to showing that

$$M \; R^{\mathsf{SC}} \; N \;\Rightarrow\; \forall G.\; M \overset{c}{\rightsquigarrow}_{\mathrm{h}} G \;\Rightarrow\; \exists H.\; N \rightsquigarrow_{\mathrm{h}} H \;\&\; G \;{}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\; H.$$

The proof is by induction (I) on c and (II) on the derivation of $M \; R^{\mathsf{SC}} \; N$, ordered lexicographically. We proceed by analysis of the derivation of $M \; R^{\mathsf{SC}}$ N. Cases (sc.R), (sc.var), and (sc.ξ) are as in the preceding proof for $\sqsubseteq_{\mathrm{B}}$. Of the remaining cases we only discuss (sc.subst). The (sc.$\mu\nu$) case follows by a similar argument.

(sc.subst) Here $M = M'[P/y]$ and $N = N'[Q/y]$, and $M \; R^{\mathsf{SC}} \; N$ is derived from $M' \; R^{\mathsf{SC}} \; N'$ and $P \; R^{\mathsf{SC}} \; Q$. By Lemma A.4(1) there exist c_1, c_2 and G' such that $M' \overset{c_1}{\rightsquigarrow}_{\mathrm{h}} G'$, $G'[P/y] \overset{c_2}{\rightsquigarrow}_{\mathrm{h}} G$ and $c = c_1 + c_2$. By part (II) of the induction hypothesis, $N' \rightsquigarrow_{\mathrm{h}} H'$ with $G' \;{}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\; H'$. The interesting case is when the head variable of G' and H' is y, that is, $G' = \lambda x_1 \ldots x_r.\, y\,\vec{M}$, $H' = \lambda x_1 \ldots x_s.\, y\,\vec{N}$, and $\vec{M} x_{r+1} \ldots x_t \; R^{\mathsf{SC}} \; \vec{N} x_{s+1} \ldots x_t$ where $t = \max(r, s)$. Note that $\vec{M} = M_1 \ldots M_m$ and $\vec{N} = N_1 \ldots N_n$ with $r - m = s - n$, and

$$\vec{M}' = (\vec{M} x_{r+1} \ldots x_t)[P/y] \; R^{\mathsf{SC}} \; (\vec{N} x_{s+1} \ldots x_t)[Q/y] = \vec{N}' \tag{A.23}$$

where $\vec{M}' = \vec{M}[P/y] x_{r+1} \ldots x_t$ and $\vec{N}' = \vec{N}[Q/y] x_{s+1} \ldots x_t$.

By Lemma A.4(3) there exist c_{21}, c_{22} and G_P such that $P \overset{c_{21}}{\rightsquigarrow}_{\mathrm{h}} G_P$, $\lambda x_1 \ldots x_r.\, G_P \,\vec{M}[P/y] \overset{c_{22}}{\rightsquigarrow}_{\mathrm{h}} G$ and $c_2 = c_{21} + c_{22}$. Since $c_{21} \leq c_2 \leq c$, part (II) of the induction hypothesis implies that $Q \rightsquigarrow_{\mathrm{h}} H_Q$ for some H_Q such that $G_P \;{}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\; H_Q$. So $G_P = \lambda z_1 \ldots z_u.\, z\,\vec{P}$, $H_Q = \lambda z_1 \ldots z_v.\, z\,\vec{Q}$, and $\vec{P} z_{u+1} \ldots z_w \; R^{\mathsf{SC}} \; \vec{Q} z_{v+1} \ldots z_w$ where $w = \max(u, v)$.

At this stage the argument splits into many different cases according to the relative values of r, s, m, n, u and v. We consider the case when $r \leq s$, $u \leq v$, $m \leq u$, and $n \leq v$. Here

$$\begin{aligned} M \rightarrow_{\mathrm{h}}^{c_1 + c_{21} + m} M'' &= \lambda x_1 \ldots x_r z_{m+1} \ldots z_u.\, (z\,\vec{P})[M'_1 \cdots M'_m / z_1 \ldots z_m] \\ N \twoheadrightarrow_{\mathrm{h}} N'' &= \lambda x_1 \ldots x_s z_{n+1} \ldots z_v.\, (z\,\vec{Q})[N'_1 \cdots N'_n / z_1 \ldots z_n] \end{aligned}$$

and by determinacy $M'' \overset{c_{22} - m}{\rightsquigarrow}_{\mathrm{h}} G$. Suppose $r - m + u \geq s$. Then

$$M'' = \lambda \vec{x} z_{n+1} \ldots z_u.\, (z\,\vec{P})[\vec{M}'/\vec{z}]$$

where $\vec{x} = x_1 \ldots x_s$ and $\vec{z} = z_1 \ldots z_n$, by α-conversion. (Otherwise, if $r - m + u < s$, we rename $x_{r+1} \ldots x_s$ to $z_{m+1} \ldots z_n$ in N'' instead.) We need to show that $N'' \rightsquigarrow_{\mathrm{h}} H$ for some H with $G \;{}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\; H$.

(1) If $z \notin \{z_1 \ldots z_n\}$, M'' and N'' are hnf's, so

$$G = M'' = \lambda\vec{x}z_{n+1}\ldots z_u.\, z\,\vec{P}[\vec{M}'/\vec{z}]$$

and we let

$$H = N'' = \lambda\vec{x}z_{n+1}\ldots z_v.\, z\,\vec{Q}[\vec{N}'/\vec{z}]$$

We conclude that $G\ {}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\ H$ by the calculation:

$$\vec{P}[\vec{M}'/\vec{z}]z_{u+1}\ldots z_v = (\vec{P}z_{u+1}\ldots z_v)[\vec{M}'/\vec{z}]\ R^{\mathsf{SC}}\ \vec{Q}[\vec{N}'/\vec{z}]$$

(2) If $z \in \{z_{m+1}\ldots z_n\}$, i.e., $z = z_i$ where $m < i \le n$, M'' is a hnf,

$$M'' = \lambda\vec{x}z_{n+1}\ldots z_u.\, x_{r+i-m}\,\vec{P}[\vec{M}'/\vec{z}]$$

so $G = M''$; and $N'' = \lambda\vec{x}z_{n+1}\ldots z_v.\, N'_i\,\vec{Q}[\vec{N}'/\vec{z}]$. Here we observe that the preceding cases in the induction proof suffice to prove the result when M is a variable; that is, we know by now that, for all variables x and terms N, if $x\ R^{\mathsf{SC}}\ N$, $N \rightsquigarrow_{\mathrm{h}} H$ for some H such that $x\ {}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\ H$. We use this result on x_{r+i-m} and N'_i, related by R^{SC} according to (A.23), to get that $N'_i \rightsquigarrow_{\mathrm{h}} \lambda\vec{y}.\, x_{r+i-m}\,\vec{L}$ for some $\vec{y} = y_1\ldots y_l$ and $\vec{L} = L_1\ldots L_l$ where $\vec{y}\ R^{\mathsf{SC}}\ \vec{L}$. Letting

$$H = \lambda\vec{x}z_{n+1}\ldots z_v y_{k+1}\ldots y_l.\, x_{r+i-m}\,\vec{L}'Q'_{k+1}\ldots Q'_q$$

where $Q'_1\ldots Q'_q = \vec{Q}[\vec{N}'/\vec{z}]$, $\vec{L}' = \vec{L}[Q'_1\cdots Q'_k/y_1\ldots y_k]$ and $k = \min(l,q)$, we see that

$$N \twoheadrightarrow_{\mathrm{h}} N'' \twoheadrightarrow_{\mathrm{h}} \lambda\vec{x}z_{n+1}\ldots z_v.\, H'_i\,\vec{Q}[\vec{N}'/\vec{z}] \rightarrow^{k}_{\mathrm{h}} H$$

and we easily calculate that $\vec{P}[\vec{M}'/\vec{z}]z_{u+1}\ldots z_v y_{k+1}\ldots y_l\ R^{\mathsf{SC}}\ \vec{L}'Q'_{k+1}\ldots Q'_q$ as required to show that $G\ {}^{\eta}\langle R^{\mathsf{SC}}\rangle^{\eta}_{\mathrm{hnf}}\ H$.

(3) If $z \in \{z_1\ldots z_m\}$, i.e., $z = z_i$ where $1 \le i \le m$,

$$M'' = \lambda\vec{x}z_{n+1}\ldots z_u.\, M'_i\,\vec{P}[\vec{M}'/\vec{z}]$$
$$N'' = \lambda\vec{x}z_{n+1}\ldots z_v.\, N'_i\,\vec{Q}[\vec{N}'/\vec{z}]$$

Let $P'_1\ldots P'_p = \vec{P}[\vec{M}'/\vec{z}]$ and $Q'_1\ldots Q'_q = \vec{Q}[\vec{N}'/\vec{z}]$, and note that $u - p = v - q$. Since $u \le v$, $p \le q$ and we calculate that

$$M'_i\,\vec{P}[\vec{M}'/\vec{z}] = M'_i\,P'_1\ldots P'_p\ R^{\mathsf{SC}}\ N'_i\,Q'_1\ldots Q'_p$$

So, observing that $M''\ \overset{c_{22}-m}{\rightsquigarrow}_{\mathrm{h}}\ G$ must be derived from $M'_i\,\vec{P}[\vec{M}'/\vec{z}]\ \overset{c_{22}-m}{\rightsquigarrow}_{\mathrm{h}}\ G'$ where $\lambda\vec{x}z_{n+1}\ldots z_u.\, G' = G$ and that $c_{22} - m < c$ because $c_{22} \le c_2 \le$

> c and $1 \leq i \leq m$, part (I) of the induction hypothesis gives us that $N'_i\, Q'_1 \ldots Q'_p \overset{c_{22}-m}{\rightsquigarrow}_{\mathrm{h}} H'$ for some H' with $G' \;{}^{\eta}\langle R^{\mathsf{SC}} \rangle^{\eta}_{\mathrm{hnf}}\; H'$. It remains to show that
>
> $$\lambda \vec{x} z_{n+1} \ldots z_v.\, H'\, Q'_{p+1} \ldots Q'_q \rightsquigarrow_{\mathrm{h}} H$$
>
> with $G \;{}^{\eta}\langle R^{\mathsf{SC}} \rangle^{\eta}_{\mathrm{hnf}}\; H$. The argument is similar to a repetition of the proof of the (sc.subst) case, now with $\vec{M}$ empty, $Q'_{p+1} \ldots Q'_q$ in place of $\vec{N}'$, and G' and H' in place G_P and H_Q, respectively (since $\vec{M}$ is empty, only sub-case (1) or (2) of the present argument applies, so there is no infinite regress). We omit the details.

This concludes the case when $r \leq s$, $u \leq v$, $m \leq u$, $n \leq v$, and $r - m + u \geq s$. The remaining cases are similar and omitted.

A.4 Weak head reduction

Recall the definition of the weak head reduction relation:

$$\text{(red.wh)} \quad (\lambda y.\, M)\, N\, \vec{P} \rightarrow_{\mathrm{wh}} M[{}^{N}\!/\!{}_{y}]\, \vec{P}$$

The weak head reduction strategy can be phrased as "left-most, outer-most reduction to whnf".

Weak head reduction can be specified equivalently as a small-step structural operational semantics, given by the structurally inductive rules in Table A.3.

Analogous to head evaluation, we specify a big-step weak head evaluation semantics as a ternary relation between terms, non-negative integers, and whnf's, defined inductively by the rules in Table A.4.

In function applications, if the operator evaluates to a λ-abstraction, its body is not evaluated prior to β-reduction. Therefore the weak head evaluation rules model the weak head reduction strategy more faithfully than was the case for head evaluation versus head reduction. We have the expected correspondence result:

$$M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} W \quad \text{if and only if} \quad M \rightarrow^{c}_{\mathrm{wh}} W \tag{A.24}$$

The proof is similar to the proof of correspondence between head evaluation and head reduction in Lemmas A.2 and A.3(1), but is simpler because of the better match between the weak head evaluation strategy and the weak head reduction strategy. The forward implication is proved by induction on the derivation of $M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} W$. For the reverse implication, first we prove

$$M \rightarrow_{\mathrm{wh}} N \;\Rightarrow\; (\; N \overset{c}{\rightsquigarrow}_{\mathrm{h}} W \;\Rightarrow\; M \overset{c+1}{\rightsquigarrow}_{\mathrm{h}} W \;)$$

by induction on the derivation of $M \rightarrow_{\mathrm{wh}} N$. The right-to-left implication of (A.24) is then proved by induction on c.

$$(\text{red}.\beta)\ \ (\lambda y.\,M)\,N \to_{\text{wh}} M[{}^{N}\!/\!{}_{y}]$$

$$(\text{red}.\mu)\qquad \frac{M \to_{\text{wh}} M'}{M\,N \to_{\text{wh}} M'\,N}$$

Table A.3
Weak head reduction as a small-step structural operational semantics

$$(\text{eval.var})\qquad x \overset{0}{\rightsquigarrow}_{\text{wh}} x$$

$$(\text{eval.fun})\qquad \lambda x.\,M \overset{0}{\rightsquigarrow}_{\text{wh}} \lambda x.\,M$$

$$(\text{eval}.\beta)\qquad \frac{M \overset{c_1}{\rightsquigarrow}_{\text{wh}} \lambda x.\,M' \quad M'[{}^{N}\!/\!{}_{x}] \overset{c_2}{\rightsquigarrow}_{\text{wh}} W}{M\,N \overset{c_1+c_2+1}{\rightsquigarrow}_{\text{wh}} W}$$

$$(\text{eval}.\mu.\text{var@})\qquad \frac{M \overset{c}{\rightsquigarrow}_{\text{wh}} y\,\vec{M}}{M\,N \overset{c}{\rightsquigarrow}_{\text{wh}} y\,\vec{M}\,N}$$

Table A.4
Weak head reduction to whnf as a big-step evaluation semantics

Lemma A.5 (1) *$M[{}^{N}\!/\!{}_{x}] \to^{c}_{\text{wh}} W$ if and only if there exist c_1, c_2 and V such that $c = c_1 + c_2$, $M \to^{c_1}_{\text{wh}} V$ and $V[{}^{N}\!/\!{}_{x}] \to^{c_2}_{\text{wh}} W$.*

(2) *$M\,\vec{N} \to^{c}_{\text{wh}} W$ if and only if there exist c_1, c_2 and V such that $c = c_1 + c_2$, $M \to^{c_1}_{\text{wh}} V$ and $V\,\vec{N} \to^{c_2}_{\text{wh}} W$.*

(3) *$(\lambda x.\,M)\,N\,\vec{P} \to^{c}_{\text{wh}} W$ if and only if $c > 0$ and $M[{}^{N}\!/\!{}_{x}]\,\vec{P} \to^{c-1}_{\text{wh}} W$.*

Proof. By (A.24) we can prove the forward implication of part (1) for weak head evaluation rather than weak head reduction; the proof is by induction on the derivation on $M[{}^{N}\!/\!{}_{x}] \overset{c}{\rightsquigarrow}_{\text{wh}} W$. The proof of the reverse implication is analogous to that of Lemma A.1.

Part (2) is proved by induction on the length of $\vec{N}$.

Part (3) is immediate from the definition of $\to_{\text{wh}}$. □

A.5 Proof of Main Lemma for whnf simulation

We are now going to prove the Main Lemma for whnf simulation.

We define:

$$M \left\langle R \right\rangle_{\mathrm{wh}} N \overset{\mathrm{def}}{\Leftrightarrow} \forall V.\, M \twoheadrightarrow_{\mathrm{wh}} V \;\Rightarrow$$
$$\exists W.\, N \twoheadrightarrow_{\mathrm{wh}} W \;\&\; V \left\langle R \right\rangle_{\mathrm{whnf}} W$$

where $\left\langle R \right\rangle_{\mathrm{whnf}}$ (the restriction of $\left\langle R \right\rangle_{\mathrm{wh}}$ to whnf's) is the smallest relation such that

$$\lambda y.\, M \left\langle R \right\rangle_{\mathrm{whnf}} \lambda y.\, N, \quad \text{if } M \; R \; N$$
$$x \,\vec{M} \left\langle R \right\rangle_{\mathrm{whnf}} x \,\vec{N}, \quad \text{if } \vec{M} \; R \; \vec{N}$$

A relation R is a whnf simulation if and only if $R \subseteq \left\langle R \right\rangle_{\mathrm{wh}}$.

In the following we assume that R is a weak head simulation up to context, $R \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{wh}}$, and we must prove that R^{SC} is a weak head simulation, $R^{\mathsf{SC}} \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{wh}}$. Expanding the definition of $\left\langle \cdot \right\rangle_{\mathrm{wh}}$ and using (A.24) this amounts to showing that

$$M \; R^{\mathsf{SC}} \; N \;\Rightarrow\; \forall V.\, M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} V \;\Rightarrow$$
$$\exists W.\, N \rightsquigarrow_{\mathrm{wh}} W \;\&\; V \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{whnf}} W.$$

The proof is by induction (i) on c and (ii) on the derivation of $M \; R^{\mathsf{SC}} \; N$, ordered lexicographically. We proceed by analysis of the derivation of $M \; R^{\mathsf{SC}}$ N.

(sc.R) Immediate from the assumption $R \subseteq \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{wh}}$.

(sc.var) Immediate by (eval.var).

(sc.ξ) Immediate by (eval.fun).

(sc.$\mu\nu$) Here $M = M'\, M''$ and $N = N'\, N''$, and $M \; R^{\mathsf{SC}} \; N$ is derived from

$$M' \; R^{\mathsf{SC}} \; N' \tag{A.25}$$
$$M'' \; R^{\mathsf{SC}} \; N'' \tag{A.26}$$

Since $M = M'\, M''$, $M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} V$ must be derived by either (eval.β) or (eval.μ.var@).

In the first case, $M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} V$ is derived by (eval.β) from premises of the form:

$$M' \overset{c_1}{\rightsquigarrow}_{\mathrm{wh}} \lambda x.\, P \tag{A.27}$$
$$P[M''/x] \overset{c_2}{\rightsquigarrow}_{\mathrm{wh}} V \tag{A.28}$$

where $c_1 + 1 + c_2 = c$. Hence $c_1, c_2 < c$. By (A.25), (A.27), and either part of the induction hypothesis, there exists a Q such that

$$N' \rightsquigarrow_{\mathrm{wh}} \lambda x.\, Q \tag{A.29}$$

and $P \ R^{\mathsf{SC}} \ Q$. By (A.26) and substitutivity of R^{SC}, we have $P[{}^{M''}\!/x] \ R^{\mathsf{SC}} \ Q[{}^{N''}\!/x]$. Since $c_2 < c$, (A.28) and part (I) of the induction hypothesis imply that

$$Q[{}^{N''}\!/x] \rightsquigarrow_{\mathrm{wh}} W \tag{A.30}$$

with $V \ \langle R^{\mathsf{SC}} \rangle_{\mathrm{whnf}} \ W$. Finally, from (A.29), (A.30) and (eval.β) we conclude that $N \rightsquigarrow_{\mathrm{wh}} W$, as desired.

In the second case, $M \overset{c}{\rightsquigarrow}_{\mathrm{wh}} V$ is derived by (eval.μ.var@) from a premise of the form $M' \overset{c}{\rightsquigarrow}_{\mathrm{wh}} y \, \vec{P}$, where $y \, \vec{P} M'' = V$. By (A.25) and part (II) of the induction hypothesis, $N' \rightsquigarrow_{\mathrm{wh}} y \, \vec{Q}$ with $\vec{P} \ R^{\mathsf{SC}} \ \vec{Q}$. Let $W = y \, \vec{Q} N''$. Then $N \rightsquigarrow_{\mathrm{wh}} W$ follows from (eval.μ.var@), and $V \ \langle R^{\mathsf{SC}} \rangle_{\mathrm{whnf}} \ W$ holds because $\vec{P} \ R^{\mathsf{SC}} \ \vec{Q}$ and $M'' \ R^{\mathsf{SC}} \ N''$.

(sc.subst) Here $M = M'[{}^{P}\!/y]$ and $N = N'[{}^{Q}\!/y]$, and $M \ R^{\mathsf{SC}} \ N$ is derived from $M' \ R^{\mathsf{SC}} \ N'$ and $P \ R^{\mathsf{SC}} \ Q$.

By Lemma A.5(1), $M'[{}^{P}\!/y] \overset{c}{\rightsquigarrow}_{\mathrm{wh}} V$ implies that there exist c_1, c_2 and V' such that

$$M' \overset{c_1}{\rightsquigarrow}_{\mathrm{wh}} V' \tag{A.31}$$

$$V'[{}^{P}\!/y] \overset{c_2}{\rightsquigarrow}_{\mathrm{wh}} V \tag{A.32}$$

and $c = c_1 + c_2$. Hence $c_1, c_2 \leq c$.

By part (II) of the induction hypothesis, $N' \rightsquigarrow_{\mathrm{wh}} W'$ with $V' \ \langle R^{\mathsf{SC}} \rangle_{\mathrm{whnf}} \ W'$. We need to exhibit a W such that

$$W'[{}^{Q}\!/y] \rightsquigarrow_{\mathrm{wh}} W \quad \& \quad V \ \langle R^{\mathsf{SC}} \rangle_{\mathrm{whnf}} \ W \tag{A.33}$$

Then the desired conclusion, $N \rightsquigarrow_{\mathrm{wh}} W$, follows by Lemma A.5(1).

Since $V' \ \langle R^{\mathsf{SC}} \rangle_{\mathrm{whnf}} \ W'$, either (1) V' and W' are both λ-abstractions, in which case $V = V'[{}^{P}\!/y]$ and $W = W'[{}^{Q}\!/y]$, and the desired conclusion follows easily using the fact that R^{SC} is substitutive; or (2) V' and W' are λ-free hnf's with the same head variable: if this is different from y, we argue as in case (1), otherwise we argue as follows. V' and W' are of the form $V' = y \, \vec{M}$ and $W' = y \, \vec{N}$ with

$$\vec{M} \ R^{\mathsf{SC}} \ \vec{N} \tag{A.34}$$

(A.32) and Lemma A.5(2) imply that there exist c_{21}, c_{22} and V'' such that

$$P \overset{c_{21}}{\rightsquigarrow}_{\mathrm{wh}} V'' \tag{A.35}$$

$$V'' \, \vec{M}[{}^{P}\!/y] \overset{c_{22}}{\rightsquigarrow}_{\mathrm{wh}} V \tag{A.36}$$

and $c_2 = c_{21} + c_{22}$. Since $c_{21} \leq c_2 \leq c$, part (II) of the induction hypothesis implies that there is a W'' such that

$$Q \rightsquigarrow_{\mathrm{wh}} W'' \tag{A.37}$$

$$V'' \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{whnf}} W'' \tag{A.38}$$

(i) Suppose V'' and W'' are λ-abstractions, $V'' = \lambda x.\, M''$ and $W'' = \lambda x.\, N''$ with $M''\ R^{\mathsf{SC}}\ N''$. If $\vec{M}$ and $\vec{N}$ are empty, that is, $V' = W' = y$, then $V = V''$, $W = W''$, and we conclude (A.33) by Lemma A.5(2) and (A.38). If $\vec{M}[{}^{P}\!/\!_{y}] = M'_1 \ldots M'_n$ and $\vec{N}[{}^{Q}\!/\!_{y}] = N'_1 \ldots N'_n$ for some $n \geq 1$, (A.36) and Lemma A.5(3) imply that $c_{22} > 0$ and $M''[{}^{M'_1}\!/\!_{x}]\, M'_2 \ldots M'_n \overset{c_{22}-1}{\rightsquigarrow}_{\mathrm{wh}} V$. Note that $c_{22} - 1 < c$, because $c_{22} \leq c_2 \leq c$, and that

$$M''[{}^{M'_1}\!/\!_{x}]\, M'_2 \ldots M'_n \quad R^{\mathsf{SC}} \quad N''[{}^{N'_1}\!/\!_{x}]\, N'_2 \ldots N'_n$$

by substitutivity and compatibility of R^{SC}. By part (i) of the induction hypothesis, there is a W such that

$$N''[{}^{N'_1}\!/\!_{x}]\, N'_2 \ldots N'_n \rightsquigarrow_{\mathrm{wh}} W \quad \& \quad V \left\langle R^{\mathsf{SC}} \right\rangle_{\mathrm{whnf}} W \tag{A.39}$$

By Lemma A.5(3), $W''\, N'_1 \ldots N'_n \rightsquigarrow_{\mathrm{wh}} W$ and, by (A.37) and Lemma A.5(2), $Q\, N'_1 \ldots N'_n \rightsquigarrow_{\mathrm{wh}} W$. Since $W'[{}^{Q}\!/\!_{y}] = Q\, N'_1 \ldots N'_n$, this together with (A.39) establishes (A.33), as desired.

(ii) If V'' and W'' are λ-free hnf's, $V = V''\, \vec{M}$ and $W = W''\, \vec{N}$. (A.33) follows by Lemma A.5(2), (A.38) and (A.34).

This concludes the proof of the Main Lemma for whnf simulation. The Main Lemma for whnf bisimulation follows by symmetry. □

References

[1] S. Abramsky, *The lazy lambda calculus,* In: D. Turner, editor, "Research Topics in Functional Programming," Addison-Wesley, 1990, pp. 65–116.

[2] S. Abramsky and C.-H. L. Ong, *Full abstraction in the lazy lambda calculus* Information and Computation **105** (1993), pp. 159–267.

[3] H. P. Barendregt, "The Lambda Calculus: Its Syntax and Semantics," Number 103 in Studies in Logic and the Foundations of Mathematics, North-Holland, revised edition, 1984.

[4] G. Berry, "Modèles complètement adéquats et stables des lambda calculs typés," Thèse de doctorat d'etat, Université Paris VII, 1979.

[5] G. Boudol, *On the semantics of the call-by-name CPS transform,* Theoretical Computer Science, 1999, Note, To appear.

[6] R. David and K. Nour, *A syntactical proof of the operational equivalence of two λ-terms,* Theoretical Computer Science **180** (1997), pp. 371–375.

[7] M. Dezani-Ciancaglini, J. Tiuryn, and P. Urzyczyn, *Discrimination by parallel observers,* In: *Proc. 12th Annual IEEE Symposium on Logic in Computer Science*, 1997, pages 396–407.

[8] H. Goguen, *Soundness of typed operational semantics for the logical framework,* In: *Proc. 4th International Conference on Typed Lambda Calculus and Applications, L'Aquila, Italy*, Lecture Notes in Computer Science **1581** (1999), Springer-Verlag.

[9] A. D. Gordon, *Bisimilarity as a theory of functional programming,* In: *Proc. 11th Conference of Mathematical Foundations of Programming Semantics*, Electronic Notes in Theoretical Computer Science **1** (1995), URL: `http://www.elsevier.nl/locate/entcs/volume1.html.`

[10] A. D. Gordon and A. M. Pitts, editors, "Higher Order Operational Techniques in Semantics," Publications of the Newton Institute. Cambridge University Press, 1998.

[11] F. Honsell and M. Lenisa, *Final semantics for untyped lambda-calculus,* In: *Proc. 2nd International Conference on Typed Lambda Calculus and Applications, Edinburgh*, Lecture Notes in Computer Science **902** (1995), pp. 249–265.

[12] D. J. Howe, *Proving congruence of bisimulation in functional programming languages,* Information and Computation **124** (1996), pp. 103–112.

[13] J. M. E. Hyland,*A syntactic characterisation of the equality in some models for the lambda calculus,* Journal of the London Mathematical Society **12** (1976), pp. 361–370.

[14] B. Jacobs and J. Rutten, *A tutorial on (co)algebras and (co)induction,* Bulletin of the EATCS **62** (1997), pp. 222–259.

[15] S. B. Lassen, *Relational reasoning about contexts,* In: Gordon and Pitts [10], pages 91–135.

[16] M. Lenisa, *A uniform syntactical method for proving coinduction principles in λ-calculi,* In: M. Bidoit and M. Dauchet, editors, *TAPSOFT'97*, Lecture Notes in Computer Science **1214** (1997), pp. 309–320.

[17] G. Longo, *Set-theoretical models of lambda calculus: Theories, expansions and isomophisms,* Annals of Pure and Applied Logic **24** (1983), pp. 153–188.

[18] J. H. Morris, "Lambda-Calculus Models of Programming Languages," PhD thesis, MIT, Dec. 1968.

[19] C.-H. L. Ong, "The Lazy Lambda Calculus: An Investigation into the Foundations of Functional Programming," PhD thesis, University of London, 1988.

[20] D. Sands, *Improvement theory and its applications,* In Gordon and Pitts [10], pages 275–306.

[21] D. Sangiorgi, *The lazy lambda calculus in a concurrency scenario,* Information and Computation **111** (1994), pp. 120–153.

[22] D. Sangiorgi, *On the bisimulation proof method,* Technical Report LFCS-94-299, University of Edinburgh, Aug. 1994.

[23] C. P. Wadsworth, *The relation between computational and denotational properties for Scott's D_∞-models of the lambda-calculus,* SIAM Journal on Computing **5** (1976), pp. 488–521.