Acta Informatica (2010) 47:111–132
DOI 10.1007/s00236-009-0112-9

ORIGINAL ARTICLE

On the observational theory of the CPS-calculus

Massimo Merro

Received: 28 February 2009 / Accepted: 12 November 2009 / Published online: 12 December 2009

Abstract We apply powerful proof-techniques of concurrency theory to study the observational theory of Thielecke's *CPS-calculus*, a distillation of the target language of Continuation-Passing Style transforms. We define a *labelled transition system* from which we derive a (weak) *labelled bisimilarity* that completely characterises Morris' context-equivalence. We prove a *context lemma* showing that Morris' context-equivalence coincides with a simpler context-equivalence closed under a smaller class of contexts. Then we profit of the determinism of the CPS-calculus to give a simpler labelled characterisation of Morris' equivalence, in the style of Abramsky's *applicative bisimilarity*. We enhance our bisimulation proof-methods with *up to bisimilarity* and *up to context* proof techniques. We use our bisimulation proof techniques to investigate a few algebraic properties on diverging terms that cannot be proved using the original axiomatic semantics of the CPS-calculus.

1 Introduction

Continuations represent a fundamental concept in the semantics of programming languages. In functional languages, a continuation is a parameter of a function that represents the "rest of the computation" [33,34]. Functions taking continuations as arguments are called *functions in Continuation-Passing Style* (briefly CPS functions), and have a special syntactic form: they terminate their computation by passing the result to the continuation.

A fairly vast literature on functional programming studies transformations of functions into CPS functions. These transformations are called *CPS transforms*. CPS transforms, as syntactic technique for introducing continuations, were first introduced by Fisher [5] and

A preliminary extended abstract of this article appeared in the Proceedings of the 22nd International Conference on the Mathematical Foundations of Program Semantics, Electronic Notes in Theoretical Computer Science 158:307–330, Elsevier, 2006.

M. Merro (✉)
Dipartimento di Informatica, Università degli Studi di Verona,
Strada Le Grazie 15, 37134 Verona, Italy
e-mail: massimo.merro@univr.it

studied in detail by Plotkin in his seminal paper on call-by-name and call-by-value λ-calculus [25].

The target language of CPS transforms is usually a simple subset of the λ-calculus that admits a very "imperative" reading in terms of *jumping* [31]. Thielecke [37] proposed a target language, called CPS-calculus, similar to the intermediate language of Appel's compiler [2], designed to bring out the jumping, imperative nature of the continuation-passing. Thielecke showed that the more traditional CPS transforms factorise through his calculus.

The CPS-calculus is a small deterministic name-passing calculus. The calculus comes equipped with an *axiomatic semantics* defined as the congruence induced by four simple axioms. Merro and Sangiorgi [15] proved the soundness of those axioms with respect to Milner and Sangiorgi's *barbed congruence* [19], a standard contextually-defined program equality. Thielecke provided also a categorical account of the structure inherent in first-class continuations building a term model, from the syntax of the CPS-calculus, as an instance of the categorical framework. A more recent account of the state of the art of the axiomatic and categorical semantics in a simply-typed call-by-value setting can be found in [6].

The CPS-calculus consists of only two syntactic constructs and one operational rule; this is enough to embody a special programming paradigm. The problem with such calculi of minimal expressiveness is the difficulty in developing an handy behavioural theory. In the current paper, we apply powerful proof-techniques of concurrency theory to develop an observational theory of the recursive CPS-calculus (the results can be adapted to other variants of the calculus). More precisely, we are interested in establishing when two CPS-terms have the same *observable behaviour*, that is, they are indistinguishable in any context. *Behavioural equivalences* are fundamental for justifying program transformations performed either by programmers, during system development, or by the optimising phases of compilers. While several notions of behavioural equivalences can be found in the literature, most of them share two key properties:

- two terms are equivalent only if they offer identical interactions to any environment, that is, they expose the same *observables*;
- the equivalence is preserved by some key constructs of the calculus, as a consequence, proving the equivalence of two large terms can be reduced to proving the equivalence of their components.

A standard notion of behavioural equality is Morris' context-equivalence [20]. The definition of Morris' equivalence is simple and intuitive; in practise, however, it is difficult to use as the quantification on all contexts is a heavy proof obligation. Simpler proof techniques are based on *labelled bisimilarities* [17,21], which are co-inductive relations that characterise the behaviour of processes using a *labelled transition system* (LTS).

1.1 Contribution

- In Sect. 3 we define a higher-order LTS for the CPS-calculus which captures *external jumps* (i.e. jumps to continuations placed within the environment). The LTS is higher-order as labels may contain CPS-terms. An intuitive *reduction semantics* for the CPS-calculus was already given in [15], in terms of *internal jumps*. If on one hand reduction semantics are easier to grasp, on the other hand LTS-based semantics are better suited for defining reasoning techniques. We check the correctness of our LTS-based semantics by proving its consistency with respect to the reduction semantics of [15].
- In Sect. 4.1, we use our LTS to define a (weak) labelled bisimilarity. Our bisimilarity is a congruence, and completely characterises Morris' context-equivalence. This result

allows us to use the simpler definition of bisimilarity to verify whether or not two terms are Morris equivalent. Notice that, in general, congruence proofs for higher-order bisimulations are hard, in particular when the syntax of the calculus is very rigid. Our proof is relatively simple as it relies on up to bisimilarity proof-techniques [29,30]. As an easy corollary, we derive a *context lemma* showing that Morris' context-equivalence coincides with a simpler contextually-based equivalence closed under a smaller class of contexts.

- In Sect. 4.2, we provide a simpler labelled characterisation of Morris' equivalence in the style of Abramsky's *applicative bisimilarity* [1]. The characterisation proof is quite simple as we profit of the determinism of the calculus to show that applicative bisimilarity and bisimilarity coincide. When defining a behavioural equality for a confluent/deterministic calculus as the CPS-calculus, applicative bisimilarity is the natural choice. On the other hand, the definition of a "standard" bisimilarity allows us to derive easier proofs for the characterisation theorem and the context lemma. Had we worked directly with the applicative bisimilarity, the proofs would have been much harder.
- In Sect. 4.3, we enhance our proof methods by providing *up to context* proof techniques [28,30] for both bisimilarities. Up to context proof techniques are very effective to reduce the size of the candidate bisimulation. In particular, these proof techniques are very useful when working with higher-order bisimulations to factor out the universally quantified terms provided by the environment.
- In Sect. 5 we investigate on divergent CPS-terms. We use our bisimulation-based proof techniques to prove a number of algebraic laws that cannot be derived using Thielecke's axiomatic semantics.

2 The CPS-calculus

In this section we introduce the syntax and the existing semantics of the CPS-calculus.

2.1 Syntax and reduction semantics

The CPS-calculus is very simple and low-level: only variables can be passed as arguments, and applications are like jumps, with variables as argument. The terms of the CPS-calculus are given by the following grammar:

$$M, N ::= a\langle b\rangle \;\big|\; M\{a\langle b\rangle \Leftarrow N\}$$

where lowercase letters $a, b, c, \ldots$ range over variables (names) and uppercase letter $L, M, N, \ldots$ range over terms. The intended meaning is that $a\langle b\rangle$ is a jump to the continuation a with actual parameter b, while $M\{a\langle b\rangle \Leftarrow N\}$ binds the continuation with body N and formal parameter b to a in M.

We study the *monadic and recursive* variant of the calculus, in that jumps have a single argument, and in a term $M\{a\langle b\rangle \Leftarrow N\}$ the sub-term N may refer to itself under a. More precisely, in a term $M\{a\langle b\rangle \Leftarrow N\}$ the scope of variable a comprehends both M and N, while that of b extends to N only.

Remark 1 The theory developed in this article can be easily extended to the *polyadic* variant of the CPS-calculus, where a jump may contain several parameters.

The set of free variables $\mathrm{fv}(M)$ of a CPS term M is defined as follows.

- $\mathrm{fv}(a\langle b\rangle) \overset{\mathrm{def}}{=} \{a, b\}$
- $\mathrm{fv}(M\{a\langle b\rangle \Leftarrow N\}) \overset{\mathrm{def}}{=} (\mathrm{fv}(M)\backslash\{a\}) \cup (\mathrm{fv}(N)\backslash\{a, b\})$.

We write $\mathrm{fv}(M, N)$ as an abbreviation for $\mathrm{fv}(M) \cup \mathrm{fv}(N)$. In a jump $a\langle b\rangle$ we say that a is in *subject* and b in *object* position. We write $M[{}^{a}\!/\!{}_{b}]$ for the capture avoiding *substitution* of variable of a for each free occurrence of variable b in M. So, alpha-conversion can be formally defined by the following two equations:

$$\begin{aligned} M\{a\langle b\rangle \Leftarrow N\} &= M\{a\langle c\rangle \Leftarrow N[{}^{c}\!/\!{}_{b}]\} && \text{for } c \notin \mathrm{fv}(N) \\ M\{a\langle b\rangle \Leftarrow N\} &= M[{}^{c}\!/\!{}_{a}]\{c\langle b\rangle \Leftarrow N[{}^{c}\!/\!{}_{a}]\} && \text{for } c \notin \mathrm{fv}(M) \cup \mathrm{fv}(N). \end{aligned}$$

We will identify terms up to alpha-conversion. It is easy to see that every CPS-term is in the form

$$a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}$$

for some $n \geq 0$. We adopt the *reduction semantics* proposed in [15]; a slight variant of the operational semantics given by Thielecke. Thus, the behaviour of CPS-terms is modelled by means of just one (global) reduction rule:

$$\begin{gathered} a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\} \\ \rightarrow \\ M_i[{}^{b}\!/\!{}_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\} \end{gathered}$$

with $1 \leq i \leq n$ and $a_j \notin \mathrm{fv}(M_i) \cup \{a_i\}$, for $1 \leq j < i$. We denote with $\rightarrow^*$ the reflexive and transitive closure of $\rightarrow$.

2.2 Behavioural semantics

In operational semantics two terms are deemed equivalent if they have the same observable behaviour in all contexts. In the CPS-calculus the notion of observability is represented by the "external" jump that a term can perform to interact with the context. We define an *observability predicate* $\downarrow_a$, for each variable a, which detects the possibility of a term to interact with the environment via a. For instance, in a jump of the form $a\langle b\rangle$, we can observe (the occurrence of a jump to) a, whereas the argument b does not play any direct role. More generally, a free variable in the leftmost position can be observed.

Definition 1 (*Observability/convergence*) Let M be a term of the CPS-calculus and a be a name, we say that M *converges to* a, written $M \downarrow_a$, if there are names $b, a_1, b_1, \ldots, a_n, b_n$, for some integer $n \geq 0$ with $a \neq a_i$, for all $1 \leq i \leq n$, such that $M = a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}$. We say that M *(weakly) converges to* a, written $M \Downarrow_a$, if there exists a CPS-term N such that $M \rightarrow^* N \downarrow_a$.

The definition of divergence is as expected. Formally,

Definition 2 (*Divergence*) A CPS-term M diverges, written $M \Uparrow$, if whenever there is a term M' such that $M \rightarrow^* M'$ then there is also a term M'' such that $M' \rightarrow M''$.

Obviously, since the calculus is deterministic, a term either diverges or (weakly) converges to some name a.

In order to define contextually-based equivalences we need to specify what a context is. A (monadic) *context* $C[\cdot]$ is a CPS-term with a hole, denoted by $[\cdot]$. CPS-contexts are generated by the following grammar:

$$C[\cdot] ::= [\cdot] \;\big|\; C[\cdot]\{a\langle x\rangle \Leftarrow M\} \;\big|\; M\{a\langle x\rangle \Leftarrow C[\cdot]\}.$$

A *static context* is a context that can be generated only applying the first two productions of the grammar above.

Everything is in place to define Morris' context-equivalences for the CPS-calculus.

Definition 3 (*Observational equalities*) Let M and N be two CPS-terms. We say that M and N are *observationally equivalent*, written $M \simeq N$, if for all static contexts $C[\cdot]$ and variables a, it holds that $C[M] \Downarrow_a$ iff $C[N] \Downarrow_a$. M and N are *observationally congruent*, written $M \cong N$, if for all contexts $C[\cdot]$ and variables a, it holds that $C[M] \Downarrow_a$ iff $C[N] \Downarrow_a$.

The intuition is that two terms are observationally indistinguishable if no amount of programming can tell them apart; obviously, $\cong \subseteq \simeq$.

Barbed equalities are branching-time contextually-based equivalences introduced by Milner and Sangiorgi in the realm of concurrent processes [19]. Their definitions can be easily adapted to the CPS-calculus.

Definition 4 (*Barbed equalities*) A symmetric relation $\mathcal{S}$ on CPS-terms is a *barbed bisimulation* if $M \; \mathcal{S} \; N$ implies:

1. If $M \rightarrow M'$ then there exists N' such that $N \rightarrow^* N'$ and $M' \; \mathcal{S} \; N'$.
2. If $M \downarrow_a$, for some a, then $N \Downarrow_a$.

Two terms M and N are *barbed bisimilar*, written $M \stackrel{\bullet}{\approx} N$, if $M \; \mathcal{S} \; N$ for some barbed bisimulation $\mathcal{S}$. Two terms M and N are *barbed equivalent* if for each static context $C[\cdot]$, it holds that $C[M] \stackrel{\bullet}{\approx} C[N]$; they are *barbed congruent* if for each context $C[\cdot]$, it holds that $C[M] \stackrel{\bullet}{\approx} C[N]$.

As the CPS-calculus is deterministic (and hence confluent), it is well-known that observational congruence (respectively, observational equivalence) coincides with barbed congruence (respectively, barbed equivalence) [17].

2.3 Axiomatic semantics

The original semantics of the CPS-calculus is an *axiomatic semantics* [37] defined as the congruence induced by the following four axioms:[1]

(DISTR) $L\{a\langle \vec{b}\rangle \Leftarrow M\}\{c\langle \vec{d}\rangle \Leftarrow N\} \equiv L\{c\langle \vec{d}\rangle \Leftarrow N\}\{a\langle \vec{b}\rangle \Leftarrow M\{c\langle \vec{d}\rangle \Leftarrow N\}\}$ where $a \neq c$ and $a, \vec{b} \notin \mathrm{fv}(N)$

(GC) $a\langle \vec{b}\rangle\{c\langle \vec{d}\rangle \Leftarrow N\} \equiv a\langle \vec{b}\rangle$ where $c \notin \mathrm{fv}(a\langle \vec{b}\rangle)$

(JMP) $a\langle \vec{b}\rangle\{a\langle \vec{c}\rangle \Leftarrow N\} \equiv N[{}^{\vec{b}}/_{\vec{c}}]\{a\langle \vec{c}\rangle \Leftarrow N\}$

(ETA) $M\{a\langle \vec{b}\rangle \Leftarrow c\langle \vec{b}\rangle\} \equiv M[{}^{c}/_{a}]$ where $a \neq c$.

The (JMP) axiom is in some sense what drives the computation. In fact, our reduction rule can be seen as a "contextual" variant of the (JMP) axiom. The axiom (GC) allows us to garbage collect unreachable continuations. The axiom (DISTR) serves to bring the components of a (jumping) redex into contiguous positions. In this respect, it is similar to the notion of structural congruence used when giving the operational semantics in process calculi [18]. We write CPS $\vdash M \equiv N$ when the axiomatic semantics can be used to derive the equality between M and N.

The axiomatic semantics is sound with respect to barbed congruence, and hence also with respect to observational congruence. Formally,

[1] Most of these axioms comes from [2].

Theorem 1 (Merro and Sangiorgi [15]) *Let M and N be two CPS-terms. Then,*

$$\mathrm{CPS} \vdash M \equiv N \text{ implies } M \cong N.$$

Notice that the axiomatic semantics is meant to be the least equality on CPS-terms that one would wish to impose, while observational equivalence is arguably the greatest such notion one could consider.

3 A labelled transition system

In Table 1 we provide a *labelled transition system* (LTS) for the CPS-calculus. Transitions are of the form $M \xrightarrow{\alpha} M'$ where α can be either τ, to model *internal jumps*, or $a\langle x\rangle N$, for some variable a and CPS-term N, to model *external jumps*. In particular, the observable action $a\langle x\rangle N$ models the capability to perform an external jump $a\langle b\rangle$, for some parameter b. Notice that our actions do not mention the argument of the jump (in this case b), although such argument has its influence on the derivative M'. Intuitively, in a transition

$$M \xrightarrow{a\langle x\rangle N} M'$$

the action $a\langle x\rangle N$ codifies the discriminating context $[\cdot]\{a\langle x\rangle \Leftarrow N\}$ with which M can interact. This context becomes part of the derivative of the transition. The main inference rules are (Jmp) and (Tau), modelling external and internal jumps, respectively. Rules (Cxt Jmp) and (Cxt Tau) are their contextual counterparts.

Our LTS is necessarily higher-order to properly model the interaction with the environment while preserving the transition closure of the calculus. In the sequel, we write $\xrightarrow{\tau}{}^{*}$ to denote the reflexive and transitive closure of $\xrightarrow{\tau}$.

Proposition 1 (Transition closure) *Let M be a CPS-term. If $M \xrightarrow{\alpha} M'$ then M' is a CPS-term.*

Proof By a simple transition induction. □

As said in the introduction the CPS-calculus is deterministic. Formally,

Proposition 2 (Determinism) *Let M be a CPS-term. Then, only one of the following two cases applies.*

1. *M may perform at most one transition of the form $M \xrightarrow{\tau} M'$, for some M'.*
2. *Fixed an arbitrary term L, the term M may perform at most one transition of the form $M \xrightarrow{a\langle x\rangle L} M'$, for some variable a and term M'.*

Table 1 Labelled transition system for the CPS-calculus

$$\text{(Jmp)}\ \frac{x \notin \mathrm{fv}(a\langle b\rangle)}{a\langle b\rangle \xrightarrow{a\langle x\rangle M} M[{}^{b}\!/\!{}_{x}]\{a\langle x\rangle \Leftarrow M\}} \qquad \text{(Tau)}\ \frac{M \xrightarrow{a\langle x\rangle N} M'}{M\{a\langle x\rangle \Leftarrow N\} \xrightarrow{\tau} M'}$$

$$\text{(Cxt Tau)}\ \frac{M \xrightarrow{\tau} M'}{M\{a\langle x\rangle \Leftarrow N\} \xrightarrow{\tau} M'\{a\langle x\rangle \Leftarrow N\}}$$

$$\text{(Cxt Jmp)}\ \frac{M \xrightarrow{a\langle x\rangle N} M'\{a\langle x\rangle \Leftarrow N\} \qquad a \neq b \qquad b \notin \mathrm{fv}(N)}{M\{b\langle y\rangle \Leftarrow O\} \xrightarrow{a\langle x\rangle N} M'\{b\langle y\rangle \Leftarrow O\}\{a\langle x\rangle \Leftarrow N\}}$$

Proof It follows from the definition of the LTS and because every CPS-term is in the form $a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}$. □

In the next result we prove the correctness of our LTS-based semantics showing that it coincides with the reduction semantics given in Sect. 2.1.

Theorem 2 (Harmony Theorem) *Let M and N be two CPS-terms. Then,*

1. $M \downarrow_a$ *iff* $M \xrightarrow{a\langle x\rangle L} M'\{a\langle x\rangle \Leftarrow L\}$ *for any CPS-term L.*
2. $M \rightarrow N$ *iff* $M \xrightarrow{\tau} N$.

Proof See the Appendix. □

The following result says that the operational semantics is preserved under (name) substitution.

Proposition 3 1. *If $M \xrightarrow{\tau} M'$ then $M\sigma \xrightarrow{\tau} M'\sigma$, for any capture avoiding substitution σ.*
2. *If $M \Downarrow_a$ and $a \neq x$ then $M[{}^b\!/\!_x] \Downarrow_a$.*

Proof By Theorem 2(2) we can rewrite the first statement as: If $M \rightarrow M'$ then $M\sigma \rightarrow M'\sigma$, for any capture avoiding substitution σ. By definition of $\rightarrow$, we have

$$M = a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}$$

and

$$M' = M_i[{}^b\!/\!_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}.$$

Notice that, for $1 \leq i \leq n$, both a_i and b_i are bound in M. Since σ is a capture avoiding substitution we can assume that, for $1 \leq i \leq n$, both a_i and b_i do not appear in σ. As a consequence,

$$M\sigma = a_i\langle b\sigma\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\sigma\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\sigma\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\sigma\}$$

and

$$\begin{aligned} M'\sigma &= (M_i[{}^b\!/\!_{b_i}]\sigma)\{a_1\langle b_1\rangle \Leftarrow M_1\sigma\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\sigma\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\sigma\} \\ &= (M_i\sigma[{}^{b\sigma}\!/\!_{b_i}])\{a_1\langle b_1\rangle \Leftarrow M_1\sigma\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\sigma\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\sigma\} \end{aligned}$$

with $M\sigma \rightarrow M'\sigma$.

The second statement follows from the first one and Theorem 2(2). In fact, if $M \Downarrow_a$ then, by Theorem 2(2), there is M' such that $M \xrightarrow{\tau}{}^{*} M' \downarrow_a$. As name substitution does not affect silent actions it follows that $M[{}^b\!/\!_x] \xrightarrow{\tau}{}^{*} M'[{}^b\!/\!_x] \downarrow_a$. □

4 Bisimulation proof methods

In this section, we propose two labelled characterisations of Morris' context-equivalence. We then prove a context lemma showing that static contexts have the same discriminating power as full contexts. As a consequence, observational congruence and observational equivalence coincide. Finally, we enhance our proof methods with up to context proof techniques.

4.1 A labelled characterisation of Morris' context-equivalence

Starting from the labelled transition system we can define our notion of bisimulation for CPS-terms. We write $\Rightarrow$ to denote the reflexive and transitive closure of $\xrightarrow{\tau}$. We write $\xRightarrow{\alpha}$ for $\Rightarrow\xrightarrow{\alpha}$, and $\xRightarrow{\hat{\alpha}}$ for $\xRightarrow{\alpha}$ if $\alpha \neq \tau$, and for $\Rightarrow$ if $\alpha = \tau$.

Definition 5 (*Bisimulation*) A symmetric relation $\mathcal{S}$ on CPS-terms is a *bisimulation* if whenever $M \mathrel{\mathcal{S}} N$ and $M \xrightarrow{\alpha} M'$ there exists a CPS-term N' such that $N \xRightarrow{\hat{\alpha}} N'$ and $M' \mathrel{\mathcal{S}} N'$. Two CPS-terms M and N are *bisimilar*, written $M \approx N$ if there is some bisimulation $\mathcal{S}$ such that $M \mathrel{\mathcal{S}} N$.

Our bisimulation is defined in a *delay* style [27,38], as weak actions always end with an observable label. It is easy to see that $\approx$ is an equivalence relation.

In order to show that our bisimilarity characterises the observational congruence we first prove the completeness of the bisimilarity with respect to the *observational equivalence* (and not the observational congruence). Notice that, in general, a result of this kind would not hold in concurrency theory. However, in our case, Lemma 1, on the insensitiveness of behavioural equalities to τ-actions, allows us to easily prove the completeness result.

Lemma 1 (Insensitiveness to τ-actions) *Let M be a CPS-term. If $M \xrightarrow{\tau} M'$ then $M \cong M'$.*

Proof By Theorem 2(2) the relations $\rightarrow$ and $\xrightarrow{\tau}$ coincide. As a consequence, there are an integer n and variables b, a_i, b_i, M_i, for $1 \leq i \leq n$, such that

$$\begin{aligned} M &= a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\} \\ M' &= M_i[{}^{b}\!/\!{}_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}. \end{aligned}$$

We prove that CPS $\vdash M \equiv M'$ by using the axioms of the previous section. In particular, M' can be derived from M by applying $i-1$ times the axiom (DISTR) to shift the continuation $\{a_i\langle x_i\rangle \Leftarrow M_i\}$ at the extreme left, once axiom (JMP) to reduce along variable a_i, and $i-1$ times the axiom (DISTR) to put back the $\{a_i\langle b_i\rangle \Leftarrow M_i\}$ at its original place. By Theorem 1 we obtain $M \cong M'$. □

Lemma 2 (Completeness of $\approx$ w.r.t. $\simeq$) *Let M and N be two CPS-terms. Then $M \simeq N$ implies $M \approx N$.*

Proof We prove that the relation $\simeq$ is a bisimulation. Let $M \simeq N$.

- Suppose that $M \xrightarrow{\tau} M'$. By Lemma 1 we have $M \cong M'$ and hence also $M \simeq M'$. Now, let us choose as matching transition $N \Rightarrow N$. By transitivity of $\simeq$ it follows that $M' \simeq N$.
- Suppose that $M \xrightarrow{a\langle x\rangle L} M'\{a\langle x\rangle \Leftarrow L\}$. By Theorem 2(1) we have $M \downarrow_a$. As $M \simeq N$ we also have $N \Downarrow_a$. By several applications of Theorem 2(2) and one application of Theorem 2(1) we have $N \xRightarrow{a\langle x\rangle L} N'\{a\langle x\rangle \Leftarrow L\}$. As $M \simeq N$ and $\simeq$ is preserved by all static contexts, we also have $M\{a\langle x\rangle \Leftarrow L\} \simeq N\{a\langle x\rangle \Leftarrow L\}$. As $M \xrightarrow{a\langle x\rangle L} M'\{a\langle x\rangle \Leftarrow L\}$ and $N \xRightarrow{a\langle x\rangle L} N'\{a\langle x\rangle \Leftarrow L\}$, by several applications of rules (Tau) and (Cxt Tau) we have $M\{a\langle x\rangle \Leftarrow L\} \xrightarrow{\tau} M'\{a\langle x\rangle \Leftarrow L\}$ and $N\{a\langle x\rangle \Leftarrow L\} \Rightarrow N'\{a\langle x\rangle \Leftarrow L\}$. By applying Lemma 1 on the two derivations we obtain $M'\{a\langle x\rangle \Leftarrow L\} \simeq N'\{a\langle x\rangle \Leftarrow L\}$.

□

Now, let us focus on the sounness proof. The main difficulty here is proving that $\approx$ is preserved by all contexts. A direct proof of that is far from trivial, due to the rigid syntax of the calculus. To this end we define an up to weak bisimilarity proof technique. Up-to proof techniques allow us to prove a bisimulation result using a concise relation that in general is not itself a bisimulation, but contained in a bisimulation [30]. Notice that, in general, the up to weak bisimilarity proof technique is not sound [17,29]. Thus, here we use a stronger definition, along the lines of Exercise 2.4.64 of [30].

In the rest of the paper, we adopt the following notation on binary relations. If $\mathcal{R}$ and $\mathcal{S}$ are binary relations over CPS-terms then we write $\mathcal{RS}$ for the binary relation resulting by the composition of $\mathcal{R}$ and $\mathcal{S}$. Thus, $M \ \mathcal{RS}\ N$ if there is M' such that $M\ \mathcal{R}\ M'$ and $M'\ \mathcal{S}\ N$.

Definition 6 (*Bisimulation up to* $\approx$) A symmetric relation $\mathcal{S}$ over CPS-terms is a *bisimulation up to* $\approx$ if $M\ \mathcal{S}\ N$ implies,

1. whenever $M \xrightarrow{\tau} M'$ then, for some N', $N \Rightarrow N'$ and $M'\ \mathcal{S}{\approx}\ N'$
2. whenever $M \xrightarrow{\alpha} M'$, $\alpha \neq \tau$, then, for some N', $N \overset{\alpha}{\Longrightarrow} N'$ and $M'\ {\approx}\mathcal{S}{\approx}\ N'$.

Lemma 3 *If $\mathcal{S}$ is a bisimulation up to $\approx$ then $\mathcal{S} \subseteq \approx$.*

Proof See the Appendix. □

Now, everything is in place to prove that $\approx$ is a congruence. Our proof relies on the up to-bisimilarity proof technique, the axiom (DISTR) for permuting continuations, and the axiom (ETA) for encoding substitutions. We then use Theorem 1 and Lemma 2 to validate these two axioms with respect to bisimilarity.

Lemma 4 ($\approx$ is preserved by all contexts) *Let M and N be two CPS-terms such that $M \approx N$. Then,*

1. $M\{a\langle b\rangle \Leftarrow O\} \approx N\{a\langle b\rangle \Leftarrow O\}$ *for all terms O and variables a and b.*
2. $O\{a\langle b\rangle \Leftarrow M\} \approx O\{a\langle b\rangle \Leftarrow N\}$ *for all terms O and variables a and b.*

Proof Let us prove that $M\{a\langle b\rangle \Leftarrow O\} \approx N\{a\langle b\rangle \Leftarrow O\}$. Let $\mathcal{S}$ be the relation defined as:

$$\mathcal{S} \overset{\text{def}}{=} \{(M\{a\langle x\rangle \Leftarrow O\}, N\{a\langle x\rangle \Leftarrow O\}) \text{ for all } a \text{ and } O \text{ s.t. } M \approx N\} \cup \approx$$

We prove that $\mathcal{S}$ is a bisimulation up to $\approx$. We do a case analysis on the transition $M\{a\langle x\rangle \Leftarrow O\} \xrightarrow{\alpha} M'$.

- Let $M\{a\langle x\rangle \Leftarrow O\} \xrightarrow{\tau} M'{=}M''\{a\langle x\rangle \Leftarrow O\}$, by an application of rule (Cxt Tau), because $M \xrightarrow{\tau} M''$. Since $M \approx N$ there is N'' such that $N \Rightarrow N''$ with $M'' \approx N''$. By several applications of rule (Cxt Tau), we get $N\{a\langle x\rangle \Leftarrow O\} \Rightarrow N''\{a\langle x\rangle \Leftarrow O\}$. By definition of $\mathcal{S}$ it follows that
$$\big(M''\{a\langle x\rangle \Leftarrow O\}, N''\{a\langle x\rangle \Leftarrow O\}\big) \in \mathcal{S}.$$
- Let $M\{a\langle x\rangle \Leftarrow O\} \xrightarrow{\tau} M'$, by an application of rule (Tau), because $M \xrightarrow{a\langle x\rangle O} M'$. Since $M \approx N$ there is N' such that $N \overset{a\langle x\rangle O}{\Longrightarrow} N'$ and $M' \approx N'$. By several applications of rule (Cxt Tau) and one application of rule (Tau) we have $N\{a\langle x\rangle \Leftarrow O\} \Rightarrow N'$. By definition of $\mathcal{S}$ it follows that $(M', N') \in \mathcal{S}$.

- Let $M\{a\langle x\rangle \Leftarrow O\} \xrightarrow{b\langle y\rangle O'} M'$, by an application of rule (Cxt Jmp), because $M \xrightarrow{b\langle y\rangle O'} M''\{b\langle y\rangle \Leftarrow O'\}$, with $M' = M''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\}$. As $M \approx N$ there is N'' such that $N \xRightarrow{b\langle y\rangle O'} N''\{b\langle y\rangle \Leftarrow O'\}$ with $M''\{b\langle y\rangle \Leftarrow O'\} \approx N''\{b\langle y\rangle \Leftarrow O'\}$.
  By applying rules (Cxt Tau) and (Cxt Jmp) we get
  $$N\{a\langle x\rangle \Leftarrow O\} \xRightarrow{b\langle y\rangle O'} N''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\}.$$
  So, we have to prove that
  $$M''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\} \approx\mathcal{S}\approx N''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\}.$$
  As $M \xrightarrow{b\langle y\rangle O'} M''\{b\langle y\rangle \Leftarrow O'\}$, by applying rule (Tau) we obtain
  $$M\{b\langle y\rangle \Leftarrow O'\} \xrightarrow{\tau} M''\{b\langle y\rangle \Leftarrow O'\}.$$
  By an application of rule (Cxt Tau) we get
  $$M\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \xrightarrow{\tau} M''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\}.$$
  By Lemma 1, the inclusion of $\cong\subseteq\simeq$, and Lemma 2 it follows that
  $$M\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \approx M''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \quad (1)$$
  With a similar reasoning, from $N \xRightarrow{b\langle y\rangle O'} N''\{b\langle y\rangle \Leftarrow O'\}$, by several applications of rules (Cxt Tau) and one application of rule (Tau) we get
  $$N\{b\langle y\rangle \Leftarrow O'\} \Rightarrow N''\{b\langle y\rangle \Leftarrow O'\}.$$
  By several applications of rule (Cxt Tau) we obtain
  $$N\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \Rightarrow N''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\}.$$
  By Lemmas 1, the inclusion $\cong\subseteq\simeq$, Lemma 2, and the transitivity of $\approx$ it follows that
  $$N\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \approx N''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \quad (2)$$
  Now, by axiom (DISTR), Theorem 1, the inclusion $\cong\subseteq\simeq$, and Lemma 2 we get
  $$M''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\} \approx M''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \quad (3)$$
  $$N''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\} \approx N''\{b\langle y\rangle \Leftarrow O'\}\{a\langle x\rangle \Leftarrow O\{b\langle y\rangle \Leftarrow O'\}\} \quad (4)$$
  Finally, since $M''\{b\langle y\rangle \Leftarrow O'\} \approx N''\{b\langle y\rangle \Leftarrow O'\}$, by definition of $\mathcal{S}$ it follows that
  $$M''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\} \approx\mathcal{S}\approx N''\{a\langle x\rangle \Leftarrow O\}\{b\langle y\rangle \Leftarrow O'\}.$$

This concludes the proof of first part of the statement.

Let us prove now that $M \approx N$ implies $O\{a\langle x\rangle \Leftarrow M\} \approx O\{a\langle x\rangle \Leftarrow N\}$. We show that the relation
$$\mathcal{S} \stackrel{\text{def}}{=} \{(O\{a\langle x\rangle \Leftarrow M\}, O\{a\langle x\rangle \Leftarrow N\}) : \text{ for all } a, M, N, O \text{ such that } M \approx N\} \cup \approx$$
is a bisimulation up to $\approx$. We do a case analysis on the transition $O\{a\langle x\rangle \Leftarrow M\} \xrightarrow{\alpha} M'$.

- Let $O\{a\langle x\rangle \Leftarrow M\} \xrightarrow{\tau} O'\{a\langle x\rangle \Leftarrow M\}$, because $O \xrightarrow{\tau} O'$ by an application of rule (Cxt Tau). This case is easy.

- Let $O\{a\langle x\rangle \Leftarrow M\} \xrightarrow{\tau} M'$, by an application of rule (Tau), because

  $$O \xrightarrow{a\langle x\rangle M} C[M[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow M\} = M',$$

  for some variable b and some context $C[\cdot] = [\cdot]\{a_1\langle x_1\rangle \Leftarrow M_1\} \ldots \{a_n\langle x_n\rangle \Leftarrow M_n\}$. As a consequence, there is N' such that

  $$O \xrightarrow{a\langle x\rangle N} C[N[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow N\} = N'$$

  and hence, by an application of rule (Tau), we have $O\{a\langle x\rangle \Leftarrow N\} \xrightarrow{\tau} N'$. As $M \approx N$, by applying the first part of the current lemma we derive

  $$M\{x\langle y\rangle \Leftarrow b\langle y\rangle\} \approx N\{x\langle y\rangle \Leftarrow b\langle y\rangle\}.$$

  By applying in sequence the axiom (ETA), Theorem 1, the inclusion $\cong \subseteq \simeq$, and Lemma 2 we obtain:

  - $M[{}^{b}\!/\!_{x}] \approx M\{x\langle y\rangle \Leftarrow b\langle y\rangle\}$
  - $N[{}^{b}\!/\!_{x}] \approx N\{x\langle y\rangle \Leftarrow b\langle y\rangle\}$.

  By the transitivity of $\approx$ we derive $M[{}^{b}\!/\!_{x}] \approx N[{}^{b}\!/\!_{x}]$. As $C[\cdot]$ is a static context, by several applications of the first part of the current lemma we obtain

  $$C[M[{}^{b}\!/\!_{x}]] \approx C[N[{}^{b}\!/\!_{x}]].$$

  Again, by an application of the first part of the current lemma we derive

  $$C[M[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow N\} \approx C[N[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow N\}.$$

  We recall that $M' = C[M[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow M\}$ and $N' = C[N[{}^{b}\!/\!_{x}]]\{a\langle x\rangle \Leftarrow N\}$. This allows us to conclude that $M' \mathcal{S}\!\approx N'$.
- Let $O\{a\langle x\rangle \Leftarrow M\} \xrightarrow{b\langle y\rangle L} M'$, by an application of rule (Cxt Jmp), because $O \xrightarrow{b\langle y\rangle L} O'\{b\langle y\rangle \Leftarrow L\}$,with $M' = O'\{a\langle x\rangle \Leftarrow M\}\{b\langle y\rangle \Leftarrow L\}$. By an application of rule (Cxt Jmp), we have $O\{a\langle x\rangle \Leftarrow N\} \xrightarrow{b\langle y\rangle L} N'$, with $N' = O'\{a\langle x\rangle \Leftarrow N\}\{b\langle y\rangle \Leftarrow L\}$. By applying in sequence the axiom (DISTR), Theorem 1, the inclusion $\cong \subseteq \simeq$, and Lemma 2 we obtain:

  - $O'\{a\langle x\rangle \Leftarrow M\}\{b\langle y\rangle \Leftarrow L\} \approx O'\{b\langle y\rangle \Leftarrow L\}\{a\langle x\rangle \Leftarrow M\{b\langle y\rangle \Leftarrow L\}\}$
  - $O'\{a\langle x\rangle \Leftarrow N\}\{b\langle y\rangle \Leftarrow L\} \approx O'\{b\langle y\rangle \Leftarrow L\}\{a\langle x\rangle \Leftarrow N\{b\langle y\rangle \Leftarrow L\}\}$.

  As $M \approx N$, by the first part of the current lemma we also have $M\{b\langle y\rangle \Leftarrow L\} \approx N\{b\langle y\rangle \Leftarrow L\}$. As a consequence,

  $$O'\{a\langle x\rangle \Leftarrow M\}\{b\langle y\rangle \Leftarrow L\} \approx\mathcal{S}\approx O'\{a\langle x\rangle \Leftarrow N\}\{b\langle y\rangle \Leftarrow L\}.$$

□

We can now prove the characterisation result.

Theorem 3 (Characterisation of $\cong$) *Let M and N be two CPS-terms. Then $M \approx N$ iff $M \cong N$.*

Proof As to the implication from left to right, by Lemma 4 we have $C[M] \approx C[N]$, for all contexts $C[\cdot]$. By Definition 5 and Theorem 2 we derive $C[M] \Downarrow_a$ iff $C[N] \Downarrow_a$. The implication from right to left follows from Lemma 2 and the fact that $\cong \subseteq \simeq$. □

An easy consequence of the previous result and Lemma 2 is the following.

Theorem 4 (Context lemma) *The relations $\simeq$ and $\cong$ coincide.*

This result shows that static contexts retain all distinguishing power of Morris' context-equivalence. Said in other words, contexts of the form $M\{a\langle x\rangle \Leftarrow C[\cdot]\}$ do not add extra distinguishing power to Morris' equivalence.

4.2 Applicative bisimilarity

As our equivalences are insensitive to τ-actions (see Lemma 1) we can simplify the definition of bisimulation by removing the clause on τ-actions. In this manner, we basically get a definition of Abramsky's *applicative bisimilarity* [1] for the CPS-calculus. In applicative bisimulations only observable (weak) actions are taken into account.

Definition 7 (*Applicative bisimilarity*) A symmetric relation $\mathcal{S}$ on CPS-terms is an *applicative bisimulation* if whenever $M \; \mathcal{S} \; N$ and $M \overset{\alpha}{\Longrightarrow} M', \alpha \neq \tau$, then there exists a CPS-term N such that $N \overset{\alpha}{\Longrightarrow} N'$ and $M' \; \mathcal{S} \; N'$. Two CPS-terms M and N are *applicative bisimilar*, written $M \approx_{\mathrm{A}} N$, if there is some applicative bisimulation $\mathcal{S}$ such that $M \; \mathcal{S} \; N$.

In general, applicative bisimulations are smaller in size than bisimulations as they allow us to collapse terms that differ only for τ-actions. It is easy to show that the applicative bisimilarity is an equivalence relation.

In the following theorem we prove that bisimilarity and applicative bisimilarity coincide.

Theorem 5 *The relations $\approx$ and $\approx_{\mathrm{A}}$ coincide.*

Proof Let us prove that $\approx \subseteq \approx_{\mathrm{A}}$. We show that the relation $\approx$ is an applicative bisimulation. Let $M \approx N$ and $M \overset{\alpha}{\Longrightarrow} M', \alpha \neq \tau$. Then, by definition there is M'' such that $M \Rightarrow M'' \xrightarrow{\alpha} M'$. By Lemmas 1 and 2 we have $M'' \approx M \approx N$. As $M'' \approx N$ there is N' such that $N \overset{\alpha}{\Longrightarrow} N'$ and $M' \approx N'$.

Let us prove now that $\approx_{\mathrm{A}} \subseteq \approx$. We show that the relation $\approx_{\mathrm{A}}$ is a bisimulation. Let $M \approx_{\mathrm{A}} N$.

1. If $M \xrightarrow{\tau} M'$, then by Lemmas 1 and 2, and the inclusion $\approx \subseteq \approx_{\mathrm{A}}$ we derive $M \approx_{\mathrm{A}} M'$. As matching transition we choose $N \Rightarrow N$. By $M \approx_{\mathrm{A}} N$ and the transitivity of $\approx_{\mathrm{A}}$ we obtain $M' \approx_{\mathrm{A}} N$.
2. If $M \xrightarrow{\alpha} M'$, as $M \approx_{\mathrm{A}} N$, there is N' such that $N \overset{\alpha}{\Longrightarrow} N'$ and $M' \approx_{\mathrm{A}} N'$. □

This result, together with Theorem 3, allows us to show that applicative bisimilarity is a labelled characterisation of Morris' equivalence. More precisely, all behavioural equivalences defined up to now coincide, as stated below.

Corollary 1 *The relations $\cong$, $\simeq$, $\approx$, and $\approx_{\mathrm{A}}$ coincide.*

Proof By an application of Theorems 3, 4, and 5. □

4.3 Up to context proof techniques

In this section we introduce *up to context* proof techniques [28,30] for both bisimilarity and applicative bisimilarity. When comparing terms in higher-order calculi, (equipped with a higher-order LTS) up to context proof techniques are very useful to reduce the size of the candidate bisimulation. Intuitively, these techniques allow us to strip off a common context from the terms under consideration.

Remark 2 Up to context techniques are particularly useful when working with applicative bisimulations. However, it is technically easier to prove the correctness of these techniques with respect to the notion of bisimulation. The correctness of the up to context technique for applicative bisimulation is an easy consequence of that for bisimulation.

Definition 8 (*Bisimulation up to context and up to* $\approx$) A symmetric relation $\mathcal{S}$ over CPS-terms is a *bisimulation up to context and up to* $\approx$ if whenever $M \; \mathcal{S} \; N$ and $M \overset{\alpha}{\Longrightarrow} M''$, there is a term N'' such that $N \overset{\hat{\alpha}}{\Longrightarrow} N''$ and there is a static context $C[\cdot]$, and terms M' and N' such that $M'' \approx C[M']$, $C[N'] \approx N''$ and $M' \; \mathcal{S} \; N'$.

In order to prove the soundness of the above proof technique we need a technical lemma.

Lemma 5 *Let $\mathcal{R}$ be a bisimulation up to context and up to $\approx$. If $M \; \mathcal{R} \; N$ and for some static context $C[\cdot]$ and term M'' it holds that $C[M] \xrightarrow{\alpha} M''$, then there exists a term N'' such that $C[N] \overset{\hat{\alpha}}{\Longrightarrow} N''$ and there are a static context $C'[\cdot]$ and terms M' and N' such that $M'' \approx C'[M']$, $C'[N'] \approx N''$ and $M' \; \mathcal{R} \; N'$.*

Proof See the Appendix. □

The previous lemma can be easily generalised to the weak case as stated below.

Lemma 6 *Let $\mathcal{R}$ be a bisimulation up to context and up to $\approx$. If $M \; \mathcal{R} \; N$ and for some static context $C[\cdot]$ and term M'' it holds that $C[M] \overset{\alpha}{\Longrightarrow} M''$, then there exists a term N'' such that $C[N] \overset{\hat{\alpha}}{\Longrightarrow} N''$ and there are a static context $C'[\cdot]$ and terms M' and N' such that $M'' \approx C'[M']$, $C'[N'] \approx N''$ and $M' \; \mathcal{R} \; N'$.*

Proof The result follows by induction on the length of the transition $C[M] \overset{\alpha}{\Longrightarrow} M''$, using Lemma 5. □

Theorem 6 *If $\mathcal{R}$ is a bisimulation up to context and up to $\approx$, then $\mathcal{R} \subseteq \approx$.*

Proof We recall that we only use static contexts. The proof consists in showing that the relation

$$\mathcal{S} \overset{\text{def}}{=} \{(M, N) : \exists C[\cdot], M', N', \text{ such that } M \approx C[M'], C[N'] \approx N, \text{ and } M' \; \mathcal{R} \; N'\}$$

is a bisimulation.

Suppose $(M, N) \in \mathcal{S}$ and $M \overset{\alpha}{\Longrightarrow} M_1$. Since $(M, N) \in \mathcal{S}$ there exist $C[\cdot], M', N'$ such that $M \approx C[M']$, $C[N'] \approx N$, and $M' \; \mathcal{R} \; N'$. As $M \approx C[M']$, the definition of bisimilarity ensures that there exists M_1' such that $C[M'] \overset{\hat{\alpha}}{\Longrightarrow} M_1'$ and $M_1 \approx M_1'$.

As $M' \; \mathcal{R} \; N'$, Lemma 6 tells us that there exist N_1', $C'[\cdot]$, M_2, N_2 such that $C[N'] \overset{\hat{\alpha}}{\Longrightarrow} N_1'$, $M_1' \approx C'[M_2]$ and $N_1' \approx C'[N_2]$, with $M_2 \; \mathcal{R} \; N_2$. As $N \approx C[N']$, the definition of bisimilarity ensures that there exists N_1 such that $N \overset{\hat{\alpha}}{\Longrightarrow} N_1$ and $N_1 \approx N_1'$. The transitivity of $\approx$ and the definition of $\mathcal{S}$ ensures that $(M_1, N_1) \in \mathcal{S}$. □

In deterministic higher-order calculi, as the CPS-calculus, it is more convenient to work with applicative bisimulations up to context and up to $\approx_{\mathrm{A}}$.

Definition 9 (*Applicative bisimulation up to context and up to* $\approx_{\mathrm{A}}$) A symmetric relation $\mathcal{S}$ over CPS-terms is an *applicative bisimulation up to context and up to* $\approx_{\mathrm{A}}$ if whenever $M \; \mathcal{S} \; N$ and $M \overset{\alpha}{\Longrightarrow} M''$, for $\alpha \neq \tau$, there is a term N'' such that $N \overset{\alpha}{\Longrightarrow} N''$ and there is a static context $C[\cdot]$, and terms M' and N' such that $M'' \approx_{\mathrm{A}} C[M']$, $C[N'] \approx_{\mathrm{A}} N''$ and $M' \; \mathcal{S} \; N'$.

The soundness of this proof technique follows from Theorems 5 and 6.

Theorem 7 *If $\mathcal{R}$ is an applicative bisimulation up to context and up to $\approx_{\mathrm{A}}$, then $\mathcal{R} \subseteq \approx_{\mathrm{A}}$.*

Proof Let us prove that $\mathcal{R}$ is also a bisimulation up to context and up to $\approx$. Let $(M, N) \in \mathcal{R}$.

1. If $M \Rightarrow M'$, for some M', then we can choose $N \Rightarrow N$. Let $C[\cdot] \overset{\text{def}}{=} [\cdot]$. By Lemmas 1 and 2 we have $M' \approx M = C[M]$, $C[N] \approx N$, and $M \mathrel{\mathcal{R}} N$.
2. If $M \overset{\alpha}{\Longrightarrow} M'$, with $\alpha \neq \tau$, the result follows from Definition 9 and Theorem 5.

By Theorem 6 it follows that $M \approx N$. By Theorem 5 we derive $M \approx_{\mathrm{A}} N$. □

5 On divergent terms

The axiomatic semantics of Thielecke, reported in Sect. 2.3, allows us to prove a wide number of equalities. However, none of those axioms deal with divergent terms. This means the axiomatic semantics does not provide any instrument to prove equality between divergent terms. On the other hand, the coinductive nature of bisimulation proof methods is particularly suited for dealing with divergent terms. So, as a workbench for both our bisimulation theory and up to context proof technique, we prove a few algebraic properties on divergent terms.

We start describing how the divergence of CPS-terms is preserved by the operators of the calculus.

Proposition 4 *Let M be a divergent CPS-term.*

1. *For any substitution σ, the term $M\sigma$ diverges.*
2. *For any static context $C[\cdot]$, the term $C[M]$ diverges.*
3. *For any term L such that $L \Downarrow_a$, the term $L\{a\langle x\rangle \Leftarrow M\}$ diverges.*

Proof 1. It follows from Proposition 3(1).
2. If $C[\cdot]$ is a static context and $M \Uparrow$, then the context $C[\cdot]$ does not play any role during the computation of $C[M]$.
3. As $L \Downarrow_a$, by Proposition 2 $L\{a\langle x\rangle \Leftarrow M\} \Rightarrow C[M\sigma]$, for some static context $C[\cdot]$ and some substitution σ. From the first and the second items of this proposition it follows that $L\{a\langle x\rangle \Leftarrow M\}$ diverges. □

Now, let us consider a few algebraic laws dealing with divergent terms. The first one equates two simple terms.

$$a\langle b\rangle\{a\langle x\rangle \Leftarrow a\langle b\rangle\} \cong a\langle c\rangle\{a\langle x\rangle \Leftarrow a\langle c\rangle\}. \tag{5}$$

According to Definition 2 the two terms diverge. However, Thielecke's axiomatic semantics cannot be used to prove this equality. In fact, we can only apply axiom (JMP) which leaves the two terms unchanged. The law above could be slightly complicated to equate two terms diverging in different ways.

$$a_1\langle b\rangle\{a_1\langle x_1\rangle \Leftarrow a_2\langle a_1\rangle\}\{a_2\langle x_2\rangle \Leftarrow x_2\langle b\rangle\} \cong a\langle c\rangle\{a\langle x\rangle \Leftarrow a\langle c\rangle\} \tag{6}$$

where the left hand term contains some kind of mutual recursion. More generally, in CPS-calculus it holds a reformulation of the Ω-equation [3] of the λ-calculus:

$$M \cong N \text{ if both } M \text{ and } N \text{ diverge.} \tag{7}$$

Notice that, in general, it may be useful to have some instruments to determine whether a term diverges. For instance, the terms appearing in Laws 5 and 6 diverge because they enter a loop.

Proposition 5 *Let M and N be CPS-terms such that $M \stackrel{\tau}{\Longrightarrow} N \stackrel{\tau}{\Longrightarrow} N$. Then $M \Uparrow$.*

Proof It follows from the determinism of the calculus. □

Theorem 8 (Ω-equation) *Let M and N be CPS-terms. If both M and N diverge then for any context $C[\cdot]$ it holds that $C[M] \cong C[N]$.*

Proof By Proposition 2, if M and N diverge then M and N are trivially applicative bisimilar. From Corollary 1 it follows $M \cong N$. From the definition of $\cong$ it follows $C[M] \cong C[N]$, for any context $C[\cdot]$. □

Now, let us focus on another algebraic law where divergence plays a crucial role.

$$b\langle a\rangle\{a\langle x\rangle \Leftarrow a\langle c\rangle\} \cong b\langle a\rangle\{a\langle x\rangle \Leftarrow a\langle d\rangle\} \tag{8}$$

Here, the two terms can perform an external jump to the continuation "b" passing, as an argument, the address a of two different but diverging, and hence equivalent, continuations.

Before proving an appropriate generalisation of the law above we need a couple of technical results.

Lemma 7 1. *If $M\{a\langle x\rangle \Leftarrow M\} \Uparrow$ then either $M \Uparrow$ or $M \Downarrow_a$.*
2. *If $M\{a\langle x\rangle \Leftarrow M\} \Uparrow$ and $L \Downarrow_a$ then $L\{a\langle x\rangle \Leftarrow M\} \Uparrow$.*

Proof See the Appendix. □

Now, everything is in place to prove a generalisation of Law 8.

Theorem 9 *If $M\{a\langle x\rangle \Leftarrow M\} \Uparrow$ and $M'\{a\langle x\rangle \Leftarrow M'\} \Uparrow$ then for any CPS-term L it holds that $L\{a\langle x\rangle \Leftarrow M\} \cong L\{a\langle x\rangle \Leftarrow M'\}$.*

Proof We prove that the binary relation $\mathcal{R}$ defined as:

$$\{\big(L\{a\langle x\rangle \Leftarrow M\}, L\{a\langle x\rangle \Leftarrow M'\}\big) \mid \forall\, a, L, M, M'.\ M\{a\langle x\rangle \Leftarrow M\} \Uparrow \wedge M'\{a\langle x\rangle \Leftarrow M'\} \Uparrow\}$$

is an applicative bisimulation up to context. Let $L\{a\langle x\rangle \Leftarrow M\} \stackrel{\alpha}{\Longrightarrow} \widehat{L}$, with $\alpha \neq \tau$. We can suppose $\alpha = b\langle y\rangle L'$, for some b and L', with $b \neq a$. By Lemma 7(2) it follows that $L \not\Downarrow_a$. As a consequence there cannot be any interaction between L and the continuation $\{a\langle x\rangle \Leftarrow M\}$. This means that the action α must be generated by L. More precisely, there is L'' such that

$$L \xRightarrow{b\langle y\rangle L'} L''\{b\langle y\rangle \Leftarrow L'\}.$$

By rule (Cxt Jmp) it follows that

$$L\{a\langle x\rangle \Leftarrow M\} \xRightarrow{b\langle y\rangle L'} L''\{a\langle x\rangle \Leftarrow M\}\{b\langle y\rangle \Leftarrow L'\}.$$

With a similar reasoning we derive

$$L\{a\langle x\rangle \Leftarrow M'\} \xRightarrow{b\langle y\rangle L'} L''\{a\langle x\rangle \Leftarrow M'\}\{b\langle y\rangle \Leftarrow L'\}.$$

If we factor out the context $[\cdot]\{b\langle y\rangle \Leftarrow L'\}$ we get $(L''\{a\langle x\rangle \Leftarrow M\}, L''\{a\langle x\rangle \Leftarrow M'\}) \in \mathcal{R}$. □

The reader may notice that the application of the up to context proof technique allows us to exhibit a more succinct proof of the previous theorem.

6 Conclusion and Related work

We have presented two labelled characterisations of Morris' observational equivalence for Thielecke's CPS-calculus. The former resembles Sangiorgi's context bisimulation for Higher-Order π-calculus [27], whereas the latter is in the style of Abramsky's applicative bisimilarity [1], an operational theory for higher-order languages, inspired by bisimulation theories for concurrency [17,21]. Our LTS has some similarities with that developed by Gordon [7] for PCF plus streams, in particular our higher-order rule (Jmp) has its counterpart in Gordon's rule (Trans Fun) for functions.

Since Abramsky's work, the idea of applicative bisimilarity has been applied to a variety of higher-order sequential languages; see [7,23] for surveys. Our characterisation proof for the applicative bisimilarity is quite different from that of [1] (due to Stoughton), as we use $\approx$ as an auxiliary relation. In fact, the presence of single arrow transitions on the left hand in the definition of $\approx$ is of great help in the congruence proof. Stoughton's proof uses a variant of Milner's [16] and Berry's [4] Context Lemma. Our congruence proof relies on up to bisimilarity proof techniques.

An immediate consequence of our characterisation result is a context lemma showing that the observational congruence coincides with the observational equivalence. In general, context lemmas are hard to prove. The literature on context lemmas for functional languages is quite large. Milner [16] showed that contextual equivalence on a combinatory-logic of PCF is unchanged if we restrict attention to 'applicative contexts' $[\cdot]a_1 \ldots a_n$. Berry [4] extended Milner's proof to the lambda-calculus form of PCF. Gordon [7] proved a context lemma for PCF plus streams showing that only evaluation contexts need to be considered. Pitts [24] proved a context lemma for a higher-order language with assignable variables that only store first-order values. The proof uses logical relations that are defined in terms of the operational semantics. Sullivan [35] defined a metalanguage based on PCF extended with I/O and dynamic store primitives. A context lemma for this language is proved by showing that an applicative simulation relation is a precongruence. Finally, Talcott [36] investigated a lambda-calculus augmented with primitive operations to manipulate the computation state (store, continuation), and the environment (sending messages, creating processes). She proved a context lemma, called 'ciu theorem', that allows us to consider only contexts which correspond to computation states in which the hole is associated to the expression to be evaluated next. The ciu theorem relies on the notion of 'uniform computation' which allows computation steps to be carried out on states with missing information.

In the current paper, we have enhanced our bisimulation proof-methods with up to bisimilarity and up to context proof techniques [28,30]. We have used these techniques to prove a few algebraic laws on divergent terms that cannot be derived by Thielecke's axiomatic semantics. In higher-order languages, up to context proof techniques are notoriously hard. Sangiorgi's bisimulation up to context is a powerful bisimulation proof method for process calculi [28,30]. Unfortunately, his correctness proof does not carry over to applicative bisimilarities for higher-order languages. Pitts [22] extended Howe's congruence proof to establish an up to context rule for applicative bisimulation. Gordon [7] and Sands [26] defined applicative bisimulations up to bisimilarity and/or up to context. They demonstrated the power of this approach to produce concise proofs of equivalences which are difficult to derive by other operational methods. However, the validity of general applicative bisimulations up to context remains an open problem [9]. Other examples of up to context bisimulation proof techniques in higher-order languages are [8] and [10]. More precisely, Koutavas and Wand [8] introduced a new notion of bisimulation for showing contextual equivalence in an untyped λ-calculus augmented with higher-order procedures and a general store. Lassen [10] provided an operational bisimulation account for Böhm tree equivalence including an elementary congruence proof, from which a bisimulation up to context technique is derived. This work is extended and generalised in [12], where underlying principles from Böhm tree [3] and Lévy-Longo tree equivalences [13,14] are adapted to the call-by-value λ-calculus. A notion of enf bisimulation is defined using eager normal form (enf) equivalence classes and eager reductions, to reduce function arguments to values before application. It is then shown that enf bisimulation congruences are analogues to Lévy-Longo tree equivalence and that they both coincide on terms in the target of the CPS transforms. An up to η-reduction proof technique for enf bisimulation is also introduced since enf bisimulation does not relate terms induced by the η equation $x = \lambda y.xy$. More recently, Støvring and Lassen [32] have defined an eager normal form bisimilarity for the untyped call-by-value lambda calculus extended with continuations and mutable references. The bisimilarity is proved to be sound and complete with respect to contextual equivalence.

Finally, Lassen and Levy [11] have developed a normal form bisimulation theory for a different CPS-calculus, Jump-With-Argument, called JWA. The paper makes three important contributions: (i) it extends normal form bisimulation to types; (ii) it provides a novel congruence proof, based on insight from game semantics; (iii) it presents a seamless treatment of η-expansion.

Acknowledgments The author would like to thank the anonymous referees for their valuable suggestions and comments. This work has been partially funded by the Italian Ministry of Research (MIUR) within the project “SOFT”.

Appendix A: Proofs

Proof of Theorem 2 1. Let us consider the implication from left to right. If $M\downarrow_a$ then, by definition, there are names $b, a_1, b_1, \ldots, a_n, b_n$, for some integer $n \geq 0$, with $a \neq a_i$ for every $1 \leq i \leq n$, such that $M = a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}$. By an application of rule (Jmp) and n applications of rule (Cxt Jmp) we get the required derivation.

Let us prove the implication from right to left. We do induction on the length of the derivation of an $a\langle x\rangle L$ action.

- Suppose that $a\langle b\rangle \xrightarrow{a\langle x\rangle L} L[{}^{b}/_{x}]\{a\langle x\rangle \Leftarrow L\}$ then $a\langle b\rangle\downarrow_a$.
- Suppose that $M\{b\langle y\rangle \Leftarrow O\} \xrightarrow{a\langle x\rangle L} M'\{b\langle y\rangle \Leftarrow O\}\{a\langle x\rangle \Leftarrow L\}$ because $M \xrightarrow{a\langle x\rangle L} M'\{a\langle x\rangle \Leftarrow L\}$, with $a \neq b$ and $b \notin \mathrm{fv}(L)$, by an application of rule (Cxt Jmp). By inductive hypothesis $M\downarrow_a$, and since $a \neq b$ we also have $M\{b\langle y\rangle \Leftarrow O\}\downarrow_a$.

2. Let us prove the implication from left to right. The only reduction rule is

$$\begin{array}{c} a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\} \\ \to \\ M_i[{}^{b}/_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle b_i\rangle \Leftarrow M_i\}\ldots\{a_n\langle b_n\rangle \Leftarrow M_n\}. \end{array}$$

Now, by applying in sequence, once the rule (Jmp), $i-1$ times the rule (Cxt Jmp), one time the rule (Tau), and $n-i$ times the rule (Cxt Tau), we get

$$a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}$$
$$\xrightarrow{\tau}$$
$$M_i[{}^{b}\!/\!_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}.$$

For the implication from right to left we do rule induction on the derivation $M \xrightarrow{\tau} N$.

- Suppose that $M\{a\langle x\rangle \Leftarrow N\} \xrightarrow{\tau} M'$ by an application of rule (Tau) with premise $M \xrightarrow{a\langle x\rangle N} M'$. By applying the first part of the current theorem we have $M \downarrow_a$. By Definition 1 it follows that

  $$M = a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}$$

  with

  $$M' = N[{}^{b}\!/\!_{x}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}\{a\langle x\rangle \Leftarrow N\}.$$

  As a consequence,

  $$a\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}\{a\langle x\rangle \Leftarrow N\}$$
  $$\to$$
  $$N[{}^{b}\!/\!_{x}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}\{a\langle x\rangle \Leftarrow N\}$$

  and hence $M\{a\langle x\rangle \Leftarrow N\} \to M'$.
- Suppose that $M\{a\langle x\rangle \Leftarrow N\} \xrightarrow{\tau} M'\{a\langle x\rangle \Leftarrow N\}$ by an application of rule (Cxt Tau) with premise $M \xrightarrow{\tau} M'$. By inductive hypothesis $M \to M'$ which is a reduction of the form:

  $$a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}$$
  $$\to$$
  $$M_i[{}^{b}\!/\!_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}.$$

  This implies

  $$a_i\langle b\rangle\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}\{a\langle x\rangle \Leftarrow N\}$$
  $$\to$$
  $$M_i[{}^{b}\!/\!_{b_i}]\{a_1\langle b_1\rangle \Leftarrow M_1\}\dots\{a_i\langle b_i\rangle \Leftarrow M_i\}\dots\{a_n\langle b_n\rangle \Leftarrow M_n\}\{a\langle x\rangle \Leftarrow N\}$$

  and therefore $M\{a\langle x\rangle \Leftarrow N\} \to M'\{a\langle x\rangle \Leftarrow N\}$. □

Proof of Lemma 3 It suffices to prove that the symmetric relation $\approx\mathcal{S}\approx$ is a bisimulation. Let L and O be two CPS-terms such that $L \approx\mathcal{S}\approx O$. This means that there are M and N such that $L \approx M \;\mathcal{S}\; N \approx O$. There are two cases.

1. Suppose $L \xrightarrow{\tau} L'$. As $L \approx M$, there is M' such that $M \Rightarrow M'$ and $L' \approx M'$. As $M \;\mathcal{S}\; N$, if $M \Rightarrow M'$ then there is N' such that $N \Rightarrow N'$ and $M' \;\mathcal{S}\!\approx N'$. Now, if $N \Rightarrow N'$ then there is O' such that $O \Rightarrow O'$ and $N' \approx O'$. By transitivity of $\approx$ we have $L' \approx\mathcal{S}\approx O'$.
2. Suppose $L \xrightarrow{\alpha} L'$. As $L \approx M$, there are M' and M'' such that $M \Rightarrow M' \xrightarrow{\alpha} M''$ and $L' \approx M''$. As $M \;\mathcal{S}\; N$, if $M \Rightarrow M'$ then there is N' such that $N \Rightarrow N'$ and $M' \;\mathcal{S}\; N'$. As a consequence, if $M' \xrightarrow{\alpha} M''$ then there are N'' and N''' such that $N' \Rightarrow N'' \xrightarrow{\alpha} N'''$ and $M'' \approx\mathcal{S}\approx N'''$. Finally, as $N \approx O$, if $N \Rightarrow\xrightarrow{\alpha} N'''$ then there are O' and O'' such that $O \Rightarrow O' \xrightarrow{\alpha} O''$ and $N''' \approx O''$. By transitivity of $\approx$ we have $L'' \approx\mathcal{S}\approx O''$. □

Proof of Lemma 5 Let $\mathcal{R}$ be a bisimulation up to context and up to $\approx$. Suppose that $(M, N) \in\mathcal{R}$ and that $C[M] \xrightarrow{\alpha} M''$ for some static context

$$C[\cdot] \stackrel{\text{def}}{=} [\cdot]\{a_1\langle x_1\rangle \Leftarrow M_1\}\ldots\{a_n\langle x_n\rangle \Leftarrow M_n\}.$$

We decompose the transition $C[M] \xrightarrow{\alpha} M''$, distinguishing two cases.

1. The transition is performed by M, that is, $M \xrightarrow{\alpha} M'$. There are two sub-cases depending on α.

   a. Let $M \xrightarrow{\tau} M'$. By several applications of rule (Cxt Tau) we derive $M'' = C[M']$. As $(M, N) \in\mathcal{R}$, there is N' such that $N \Rightarrow N'$ and there is a static context $C'[\cdot]$, and terms M''' and N''' such that $M' \approx C'[M''']$ and $C'[N'''] \approx N'$, with $(M''', N''') \in\mathcal{R}$. Let $D[\cdot] \stackrel{\text{def}}{=} C[C'[\cdot]]$. $D[\cdot]$ is a static context. As $M' \approx C'[M''']$ and $\approx$ is preserved by all contexts, $M'' = C[M'] \approx C[C'[M''']] = D[M''']$. As $N \Rightarrow N'$, by several applications of rule (Cxt Tau) we have $C[N] \Rightarrow C[N']$ with $C[N'] \approx C[C'[N''']] = D[N''']$. To conclude, we recall that $(M''', N''') \in\mathcal{R}$.

   b. Let $M \xrightarrow{a\langle x\rangle L} M' = \widehat{M}\{a\langle x\rangle \Leftarrow L\}$, for some a, L and $\widehat{M}$. By several applications of rule (Cxt Jmp) we derive $M'' = C[\widehat{M}]\{a\langle x\rangle \Leftarrow L\}$.
   As $(M, N) \in\mathcal{R}$, there is N' such that $N \xRightarrow{a\langle x\rangle L} N' = \widehat{N}\{a\langle x\rangle \Leftarrow L\}$, for some $\widehat{N}$, and there is a static context $C'[\cdot]$, and terms M''' and N''' such that $M' \approx C'[M''']$ and $C'[N'''] \approx N'$, with $(M''', N''') \in\mathcal{R}$. As $N \xRightarrow{a\langle x\rangle L} N'$, by several applications of rules (Cxt Tau) and (Cxt Jmp), we have $C[N] \xRightarrow{a\langle x\rangle L} N'' = C[\widehat{N}]\{a\langle x\rangle \Leftarrow L\}$. Let

   $$\widehat{C}[\cdot] \stackrel{\text{def}}{=} [\cdot]\{a_1\langle x_1\rangle \Leftarrow M_1\{a\langle x\rangle \Leftarrow L\}\}\ldots\{a_n\langle x_n\rangle \Leftarrow M_n\{a\langle x\rangle \Leftarrow L\}\}.$$

   By applying n times the axiom (DISTR) (to shift the continuation $\{a\langle x\rangle \Leftarrow L\}$ at the extreme left), Theorem 1, the inclusion $\cong\subseteq\simeq$, Lemma 2 and Lemma 4 we obtain:

   $$C[\widehat{M}]\{a\langle x\rangle \Leftarrow L\} \approx \widehat{C}[\widehat{M}\{a\langle x\rangle \Leftarrow L\}].$$

   As $\widehat{M}\{a\langle x\rangle \Leftarrow L\} = M' \approx C'[M''']$ and $\approx$ is preserved by all contexts, it follows that

   $$\widehat{C}[\widehat{M}\{a\langle x\rangle \Leftarrow L\}] \approx \widehat{C}[C'[M''']].$$

   With a similar reasoning we derive:

   $$C[\widehat{N}]\{a\langle x\rangle \Leftarrow L\} \approx \widehat{C}[\widehat{N}\{a\langle x\rangle \Leftarrow L\}] \approx \widehat{C}[C'[N''']].$$

   Let $D[\cdot] \stackrel{\text{def}}{=} \widehat{C}[C'[\cdot]]$. $D[\cdot]$ is a static context. By the transitivity of $\approx$ we obtain $M'' \approx D[M''']$, $D[N'''] \approx N''$ and $(M''', N''') \in\mathcal{R}$.

2. The transition is due to an interaction between M and the context $C[\cdot]$. More precisely, $M \xrightarrow{a_i\langle x_i\rangle M_i} M' = \widehat{M}\{a_i\langle x_i\rangle \Leftarrow M_i\}$, for some $1 \leq i \leq n$ and term $\widehat{M}$. By $i-1$ applications of rule (Cxt Jmp) we have

$$\begin{gathered} M\{a_1\langle x_1\rangle \Leftarrow M_1\}\ldots\{a_{i-1}\langle x_{i-1}\rangle \Leftarrow M_{i-1}\} \\ \xrightarrow{a_i\langle x_i\rangle M_i} \\ \widehat{M}\{a_1\langle x_1\rangle \Leftarrow M_1\}\ldots\{a_{i-1}\langle x_{i-1}\rangle \Leftarrow M_{i-1}\}\{a_i\langle x_i\rangle \Leftarrow M_i\}. \end{gathered}$$

By an application of rule (Tau) we have

$$M\{a_1\langle x_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle x_i\rangle \Leftarrow M_i\} \xrightarrow{\tau} \widehat{M}\{a_1\langle x_1\rangle \Leftarrow M_1\}\ldots\{a_i\langle x_i\rangle \Leftarrow M_i\}.$$

By $n-i$ applications of rule (Cxt Tau) we have $M'' = C[\widehat{M}]$. As $(M, N) \in\mathcal{R}$ there is N' such that $N \xRightarrow{a_i\langle x_i\rangle M_i} N' = \widehat{N}\{a_i\langle x_i\rangle \Leftarrow M_i\}$, for some $\widehat{N}$, and there is a static context $C'[\cdot]$, and terms M''' and N''' such that $M' \approx C'[M''']$ and $N' \approx C'[N''']$ with $(M''', N''') \in\mathcal{R}$. As $N \xRightarrow{a_i\langle x_i\rangle M_i} N'$, by $i-1$ applications of rule (Cxt Jmp), and $n-i$ applications of rule (Cxt Tau) we have $C[N] \Rightarrow N'' = C[\widehat{N}]$.
Let

$$\begin{aligned}\widehat{C}[\cdot] \overset{\text{def}}{=} & [\cdot]\{a_1\langle x_1\rangle \Leftarrow M_1\{a_i\langle x_i\rangle \Leftarrow M_i\}\}\\ & \ldots\{a_{i-1}\langle x_{i-1}\rangle \Leftarrow M_{i-1}\{a_i\langle x_i\rangle \Leftarrow M_i\}\}\{a_{i+1}\langle x_{i+1}\rangle \Leftarrow M_{i+1}\}\\ & \ldots\{a_n\langle x_n\rangle \Leftarrow M_n\}.\end{aligned}$$

By applying $i-1$ times the axiom (DISTR) (to shift $\{a_i\langle x_i\rangle \Leftarrow M_i\}$ at the extreme left), Theorem 1, the inclusion $\cong\subseteq\simeq$, Lemma 2 and Lemma 4 we obtain:

$$C[\widehat{M}] \approx \widehat{C}[\widehat{M}\{a_i\langle x_i\rangle \Leftarrow M_i\}].$$

As $\widehat{M}\{a_i\langle x_i\rangle \Leftarrow M_i\} = M' \approx C'[M''']$ and $\approx$ is preserved by all contexts, it follows that

$$\widehat{C}[\widehat{M}\{a_i\langle x_i\rangle \Leftarrow M_i\}] \approx \widehat{C}[C'[M''']].$$

With a similar reasoning we derive

$$C[\widehat{N}] \approx \widehat{C}[\widehat{N}\{a_i\langle x_i\rangle \Leftarrow M_i\}] \approx \widehat{C}[C'[N''']].$$

Let $D[\cdot] \overset{\text{def}}{=} \widehat{C}[C'[\cdot]]$. $D[\cdot]$ is a static context. By the transitivity of $\approx$ we obtain $M'' \approx D[M''']$, $N'' \approx D[N''']$ and $(M''', N''') \in\mathcal{R}$. □

Proof of Lemma 7 1. There are two possibilities: either $M \Uparrow$ or $M \Downarrow_b$, for some variable b. However, if $M \Downarrow_b$ then it must be $b = a$, otherwise we would have $M\{a\langle x\rangle \Leftarrow M\} \Downarrow_b$.
2. If $L \Downarrow_a$ then

$$L\{a\langle x\rangle \Leftarrow M\} \Rightarrow C[M[{}^{b}\!/\!{}_{x}]]\{a\langle x\rangle \Leftarrow M\}$$

for some variable b and some static context $C[\cdot]$ that does not capture free occurrences of variable a. As $M\{a\langle x\rangle \Leftarrow M\} \Uparrow$, by the previous item of this lemma there are two possibilities: either $M \Uparrow$ or $M \Downarrow_a$. If $M \Uparrow$ by Proposition 3(1) also $M[{}^{b}\!/\!{}_{x}] \Uparrow$. By Proposition 4(2) we obtain $C[M[{}^{b}\!/\!{}_{x}]]\{a\langle x\rangle \Leftarrow M\} \Uparrow$. If $M \Downarrow_a$ by Proposition 3(2) also $M[{}^{b}\!/\!{}_{x}] \Downarrow_a$. As $C[\cdot]$ does not capture the free occurrences a, it follows that

$$C[M[{}^{b}\!/\!{}_{x}]]\{a\langle x\rangle \Leftarrow M\} \Rightarrow C'[M[{}^{b'}\!/\!{}_{x}]]\{a\langle x\rangle \Leftarrow M\}$$

for some variable b' and some static context $C'[\cdot]$ obtained by adding the continuations of $C[\cdot]$ to those of $M[{}^{b}\!/\!{}_{x}]$. As a consequence also $C'[\cdot]$ does not capture the free occurrences of a. As $M \Downarrow_a$, by Proposition 3(2) we have $M[{}^{b'}\!/\!{}_{x}] \Downarrow_a$ and the reduction sequence above can be reproduced "ad infinitum" showing that the term under investigation diverges.

□

References

1. Abramsky, S., Ong, L.: Full abstraction in the lazy lambda calculus. Inf. Comput. **105**, 159–267 (1993)
2. Appel, A.: Compiling with Continuations. Cambridge University Press, Cambridge (1992)
3. Barendregt, H.: The Lambda Calculus: Its Syntax and Semantics, Studies in Logic, vol. 103, Revised edn. North Holland, Amsterdam (1984)
4. Berry, G.: Some syntactic and categorical constructions of lambda calculus models. Technical Report RR-80, INRIA-Sophia Antipolis (1981)
5. Fischer, M.J.: Lambda-calculus schemata. In: Proceedings of ACM Conference on Proving Assertions about Programs, pp. 104–109. SIGPLAN Notice 7(1) (1972)
6. Führmann, C., Thielecke, H.: On the call-by-Value CPS transform and its semantics. Inf. Comput. **188**(2), 241–283 (2004)
7. Gordon, A.D.: Bisimilarity as a theory of functional programming. Theor. Comput. Sci. **228**, 5–47 (1999)
8. Koutavas, V., Wand, M.: Small bisimulations for reasoning about higher-order imperative programs. In: Proceedings of 33rd POPL, pp. 141–152. ACM Press (2006)
9. Lassen, S.B.: Relational reasoning about contexts. In: Higher Order Operational Techniques in Semantics, pp. 91–135. Cambridge University Press (1998)
10. Lassen, S.B.: Bisimulation in untyped lambda calculus: Böhm trees and bisimulation up to context. In: Proceedings of 15th MFPS, vol. 20 of Electronic Notes in Theoretical Computer Science (1999)
11. Lassen, S.B., Levy, P.B.: Typed normal form bisimulation. In: Proceedings of 21st CSL, Lecture Notes in Computer Science, vol. 4646, pp. 283–297 (2007)
12. Lassen, S.B.: Eager normal form bisimulation. In: Proceedings of 20th LICS, pp. 345–354. IEEE Computer Society (2005)
13. Lévy, J.J.: An algebraic interpretation of the $\lambda\beta\kappa$-calculus; and an application of a labelled λ-calculus. Theor. Comput. Sci. **2**(1), 97–114 (1976)
14. Longo, G.: Set theoretical models of lambda calculus: theory, expansions and isomorphisms. Ann. Pure Appl. Logic **24**, 153–188 (1983)
15. Merro, M., Sangiorgi, D.: On asynchrony in name-passing calculi. J. Math. Struct. Comput. Sci. **14**, 715–767 (2004)
16. Milner, R.: Fully abstract models of typed lambda calculus. Theor. Comput. Sci. **4**, 1–22 (1977)
17. Milner, R.: Communication and Concurrency. Prentice Hall, Englewood Cliffs (1989)
18. Milner, R.: The polyadic π-calculus: a tutorial. Technical Report ECS–LFCS–91–180, LFCS, Department of Computer Science, Edinburgh University, October 1991. Also in Bauer, F.L., Brauer, W., Schwichtenberg, H. (eds.) Logic and Algebra of Specification. Springer (1993)
19. Milner, R., Sangiorgi, D.: Barbed bisimulation. In: Proceedings of 19th ICALP, LNCS, vol. 623, pp. 685–695. Springer (1992)
20. Morris, J.: Lambda-calculus Models of Programming Languages. PhD thesis, Massachusetts Institute of Technology (1968)
21. Park, D.M.: Concurrency on automata and infinite sequences. In: Deussen, P. (ed.) Conference on Theoretical Computer Science, LNCS, vol. 104. Springer (1981)
22. Pitts, A.M.: An Extension of Howe's Construction to Yield Simulation-up-to-context Results. Unpublished Manuscript (1995)
23. Pitts, A.M.: Operationally-based theories of program equivalence. In: Dybjer, P., Pitts, A.M. (eds.) Semantics and Logics of Computation, pp. 241–298. Publications of the Newton Institute, Cambridge University Press, USA (1997)
24. Pitts, A.M.: Reasoning about local variables with operationally-based logical relations. In: O'Hearn, P.W., Tennent, R.D. (eds.) Algol-Like Languages, vol. 2, chap. 17, pp. 173–193. Birkhauser (1997). Reprinted from Proceedings 11th Annual IEEE Symposium on Logic in Computer Science, pp. 152–163, Brunswick (July 1996)
25. Plotkin, G.D.: Call by name, call by value and the λ-calculus. Theor. Comput. Sci. **1**, 125–159 (1975)
26. Sands, D.: From SOS rules to proof principles: an operational metatheory for functional languages. In: Proceedings of 24th POPL, pp. 428–441. ACM Press (1997)
27. Sangiorgi, D.: Bisimulation for higher-order process calculi. Inf. Comput. **131**(2), 141–178 (1996)
28. Sangiorgi, D.: On the bisimulation proof method. J. Math. Struct. Comput. Sci. **8**, 447–479 (1998)
29. Sangiorgi, D., Milner, R.: The problem of "Weak Bisimulation up to". In: Proceedings of CONCUR '92, LNCS, vol. 630, pp. 32–46. Springer (1992)
30. Sangiorgi, D., Walker, D.: The π-calculus: A Theory of Mobile Processes. Cambridge University Press, Cambridge (2001)
31. Steele, G.: RABBIT: A compiler for SCHEME. Technical Report AITR-474, MIT Artificial Intelligence Laboratory (1978)

32. Støvring, K., Lassen, S. B.: A complete, co-inductive syntactic theory of sequential control and state. In: Proceedings of 34th POPL, pp. 161–172. ACM Press (2007)
33. Strachey, C., Wadsworth, P.: Continuations: A mathematical semantics for handling full jumps. Technical Report PRG-11, Oxford University Computing Laboratory, Programming Research Group (1974)
34. Strachey, C., Wadsworth, P.: Continuations: a mathematical semantics for handling full jumps. High. Order Symb. Comput. **13**(1/2), 135–152 (2000)
35. Sullivan, G.T.: An Extensional MetaLanguage with I/O and a Dynamic Store. PhD thesis, Northeastern University (1996)
36. Talcott, C.: Reasoning about functions with effects. In: Higher Order Operational Techniques in Semantics, pp. 347–390. Cambridge University Press (1998)
37. Thielecke, H.: Categorical Structure of Continuation Passing Style. PhD thesis, University of Edinburgh (1997). Also available as technical report ECS-LFCS-97-376
38. Weijland, W.P.: Synchrony and Asynchrony in Process Algebra. PhD thesis, University of Amsterdam (1989)