An Introduction to the Clocked Lambda Calculus*

Jörg Endrullis, Dimitri Hendriks, Jan Willem Klop, and Andrew Polonsky

VU University Amsterdam, The Netherlands

Abstract
We give a brief introduction to the clocked $\lambda$-calculus, an extension of the classical $\lambda$-calculus with a unary symbol $\tau$ used to witness the $\beta$-steps. In contrast to the classical $\lambda$-calculus, this extension is infinitary strongly normalising and infinitary confluent. The infinitary normal forms are enriched Lévy–Longo Trees, which we call clocked Lévy–Longo Trees.

1998 ACM Subject Classification D.1.1, D.3.1, F.4.1, F.4.2, I.1.1, I.1.3

Keywords and phrases lambda calculus, convertibility, Böhm Trees

1 The Clocked Lambda Calculus
The classical $\lambda$-calculus [1] is based on the $\beta$-rule
$$(\lambda x. M)N \rightarrow M[x:= N]$$

This calculus is neither infinitary normalising
$$(\lambda x.xx)(\lambda x.xx) \rightarrow (\lambda x.xx)(\lambda x.xx) \rightarrow \ldots ,$$
nor infinitary confluent. To see this, let
$$Y_0 \equiv \lambda f. \omega f \omega f$$
be Curry’s fixed point combinator. The term $Y_0 I$ admits the infinite (strongly convergent) rewrite sequences
$$Y_0 I \rightarrow_\beta (\lambda x. I((xx))(xx)) \rightarrow_\beta \omega$$
$$Y_0 I \rightarrow_\beta (\lambda x. I((xx))(xx)) \rightarrow_\tau^2 \Omega = (\lambda x.xx)(\lambda x.xx)$$

Here infinitary confluence fails: the terms $I^\omega$ and $\Omega$ have no common reduct since they reduce only to themselves (see [2] and [17, Chapter 12]). Even though infinitary confluence fails, the calculus has the property of infinitary unique normal forms. When considering the $\beta$- and $\tau$-rule together, even this property fails, see further [10, 4].

The clocked $\lambda$-calculus [12] consists of the following two rules:
$$(\lambda x. M)N \rightarrow \tau(M[x:= N])$$
$$\tau(M)N \rightarrow \tau(MN)$$

Here every $\beta$-step produces a symbol $\tau$ as a witness of the step. The second rule is used to move the $\tau$’s out of the way of applications and hence potential $\beta$-redexes. We write $\rightarrow_\varnothing$ for the reduction relation of the clocked $\lambda$-calculus.

For a simple example, consider the following reduction:
$$III \rightarrow_\varnothing \tau(l)I \rightarrow_\varnothing \tau(I) \rightarrow_\varnothing \tau(\tau(l))$$

where $l = \lambda x.x$. Note that the second step moves the $\tau$ out of the way of a $\beta$-redex.

* This paper has been published at the Workshop on Infinitary Rewriting 2014. It is a brief introduction to the work [9, 11, 8, 12].
As a second example, let us consider Curry’s fixed point combinator:

\[
Y_0 f \equiv (\lambda f. \omega f f) f \rightarrow_{\beta} \tau(\omega f f) \\
\omega f f \rightarrow_{\beta} \tau(f(\omega f f))
\]

Hence \(Y_0 f\) rewrites to the infinite normal form

\[
Y_0 f \rightarrow_{\rightarrow} \tau(\tau(\tau(\tau(\ldots))))
\]

written without brackets as \(\tau\tau\tau\tau\tau\tau\ldots\).

The clocked \(\lambda\)-calculus enjoys the properties of infinitary confluence, infinitary strong normalization \([15, 18, 5]\) and hence infinitary unique normal forms:

- \(\text{SN}^\infty\): all infinite rewrite sequences are strongly convergent;
- \(\text{CR}^\infty\): \(\forall M, N_1, N_2 (N_1 \leftarrow \rightarrow R M \rightarrow \rightarrow R N_2 \implies N_1 \rightarrow \rightarrow R \cdot \leftarrow \rightarrow R N_2)\);
- \(\text{UN}^\infty\): \(\forall M, N_1, N_2 (N_1 \leftarrow \rightarrow R M \rightarrow \rightarrow R N_2 \text{ and } N_1, N_2 \text{ normal forms } \implies N_1 \equiv N_2)\).

**Lemma 1.** The relation \(\rightarrow_{\rightarrow}\) has the properties \(\text{CR}^\infty, \text{SN}^\infty\) and \(\text{UN}^\infty\).

### 2 Clocked Lévy–Longo Trees

The unique infinitary normal forms with respect to \(\rightarrow_{\rightarrow}\) are clocked Lévy–Longo Trees \([9, 11, 12]\), that is, Lévy–Longo Trees (a variant of Böhm Trees) enriched with symbols \(\tau\) witnessing the \(\beta\)-steps performed in the reduction to the normal form. We write \(\text{LLT}_{\beta}(M)\) for the unique infinite normal form of \(M\).

Consider the well-known fixed point combinators of Curry and Turing, \(Y_0\) and \(Y_1\):

\[
Y_0 \equiv \lambda f. \omega f f \\
\omega f \equiv \lambda x. f(xx) \\
Y_1 \equiv \eta \eta \\
\eta \equiv \lambda x f. f(xxf)
\]

Figure 1 displays the clocked Lévy–Longo Trees of \(Y_0 f\) (left) and \(Y_1 f\) (right) where we write \(\tau^n(t)\) for \(\tau(\tau(\ldots(\tau(t)\ldots)\ldots)\ldots)\). For \(Y_0 f\) we have seen the reduction to the infinite normal form above, and a similar computation leads to the clocked Lévy–Longo Tree of \(Y_1 f\). The \(\tau\)'s in the clocked Lévy–Longo Tree witness the number of head reduction steps needed to normalise the corresponding subterm to weak head normal form.
Discriminating Lambda Terms

For more details on the results in this section, we refer to [12, 11].

We define $\rightarrow_\tau$ by the rule

$$\tau(M) \rightarrow M$$

and use $=_\tau$ to denote the equivalence closure of $\rightarrow_\tau$. For $M, N \in \text{Ter}^\infty(\lambda \tau)$, we define

(i) $M \geq_\tau N$, $M$ is globally improved by $N$ iff $\text{LLT}_\tau(M) \rightarrow_\tau \text{LLT}_\tau(N)$;
(ii) $M =e_\tau N$, $M$ eventually matches $N$ iff $\text{LLT}_\tau(M) =_\tau \text{LLT}_\tau(N)$.

For example, $\Upsilon_0 f$ globally improves $\Upsilon_1 f$ ($\Upsilon_0 f \geq_\tau \Upsilon_1 f$) as can be seen from the clocked Lévy–Longo Trees of $\Upsilon_0 f$ and $\Upsilon_1 f$ in Figure 1.

Theorem 2. Clocks are accelerated under reduction, that is, if $M \rightarrow N$, then the reduct $N$ improves $M$ globally, that is, $\text{LLT}_\tau(M) \rightarrow_\tau \text{LLT}_\tau(N)$.

As a consequence we obtain the following method for discriminating $\lambda$-terms:

Theorem 3. Let $M$ and $N$ be $\lambda$-terms. If $N$ cannot be improved globally by any reduct of $M$, then $M \not= _\beta N$.

In [11] we use this theorem to answer the following question of Selinger and Plotkin [16]: Is there a fixed point combinator $Y$ such that $A Y \equiv Y (\lambda z.f zz) =_\beta Y (\lambda x.Y (\lambda y.f xy)) \equiv B Y$

or in other notation:

$$\mu z.f zz =_\beta \mu x.\mu y.f xy,$$

with the usual definition $\mu x.M(x) = Y (\lambda x.M(x))$. The terms $A Y$ and $B Y$ have the same Böhm Trees, namely the solution of $T = f TT$. Clocked Lévy–Longo Trees can be employed to show that such fixed point combinators do not exist, see [11]. For deciding equality of $\mu$-terms with the usual unfolding rule $\mu z.M(z) = M[z := \mu z.M(z)]$, see [6].

For a large class of $\lambda$-terms the clocks are invariant under reduction, that is, the clocked Lévy–Longo Trees coincide up to insertion and removal of a finite number of $\tau$'s.

Definition 4 (Simple terms). A redex $(\lambda x.M)N$ is called:
(i) linear if $x$ has at most one occurrence in $M$;
(ii) call-by-value if $N$ is a normal form; and
(iii) simple if it is linear or call-by-value.

A $\lambda$-term $M$ is simple if (a) it has no weak head normal form, or the head reduction to whnf contracts only simple redexes and is of one of the following forms: (b) $M \rightarrow_h \lambda x.M'$ with $M'$ a simple term, or (c) $M \rightarrow_h y M_1 \ldots M_m$ with $M_1, \ldots, M_m$ simple terms.

Note that this definition is inherently coinductive; this is similar to the definition of Böhm Trees in [1]. The infinitary rewrite relation itself can also be defined coinductively, see further [3, 13, 7].

Theorem 5. Let $N$ be a reduct of a simple term $M$. Then $N$ eventually matches $M$ (i.e., $\text{LLT}_\tau(M) =_\tau \text{LLT}_\tau(N)$).

For simple terms, the discrimination method can be simplified as follows:
4 An Introduction to the Clocked Lambda Calculus

Theorem 6. If simple terms \( M, N \) do not eventually match \( \text{LLT}_\Theta(M) \neq \tau \text{LLT}_\Theta(N) \), then they are not \( \beta \)-convertible, that is, \( M \neq \beta N \).

Example 7. We show that the fixed point combinators \( Y_0, Y_1, Y_2, \ldots \) of the Böhm sequence are all inconvertible. For \( n \geq 1 \), define
\[
Y_n = \eta \eta \delta^{n-1}
\]
where
\[
MN^{n-0} = M \\
MN^{n+1} = MNN^n
\]
The clocked Lévy–Longo Trees of \( Y_0x \) and \( Y_1x \) are shown in Figure 1. We now determine the clocked Lévy–Longo Trees of \( Y_nx \) for \( n \geq 2 \):
\[
Y_n \equiv \eta \eta \delta^{n-1}x
\]
\[
\rightarrow \phi \tau(\lambda f.f(\eta \eta f))\delta^{n-1}x
\]
\[
\rightarrow \phi \tau(\lambda f.f(\eta \eta f))\delta \delta^{n-2}x
\]
\[
\rightarrow \phi \tau(\tau(\eta \eta \delta))\delta \delta^{n-2}x
\]
\[
\rightarrow \phi \tau^2(\delta(\eta \eta \delta))\delta \delta^{n-2}x
\]
\[
\rightarrow \phi \tau^3(\delta(\eta \eta \delta))\delta \delta^{n-3}x
\]
\[
\vdots
\]
\[
\rightarrow \phi \tau^{2n-2}(\delta(\eta \eta \delta)^{n-1}x)
\]
\[
\rightarrow \phi \tau^{2n}(x(\eta \eta \delta^{n-1}x))
\]
None of these steps duplicate a redex, hence \( Y_n \) is a simple term. We have
\[
\text{LLT}_\Theta(Y_nx) \equiv \tau^{2n}(x \text{LLT}_\Theta(Y_nx))
\]
Observe that all of the clocked Lévy–Longo Trees \( \text{LLT}_\Theta(Y_nx) \) differ in an infinite number of \( \tau \)'s. By Theorem 6 it follows that all terms in the Böhm sequence are inconvertible.

4 Atomic Clocked Lambda Calculus

The clocked \( \lambda \)-calculus can be enhanced to not only recording whether head reduction steps have taken place, but also where they took place. We use \( \{\lambda, L, R, \tau\}^* \) for the positions.

The atomic clocked \( \lambda \)-calculus consists of the rules
\[
(\lambda x.M)N \rightarrow \tau(M[x := N])
\]
\[
\tau_p(M)N \rightarrow \tau_{lp}(MN)
\]
The atomic clocks further strengthen the discrimination power of method Lévy–Longo Trees.

Let \( S = \lambda abc.ac(bc) \). For \( k, n_1, \ldots, n_k \in \mathbb{N} \) define a fixed point combinator \( Y^{(n_1, \ldots, n_k)} \) by
\[
Y^{(n_1, \ldots, n_k)} = G_k[\ldots G_1[Y_0] \ldots]
\]
where \( G_n = [[SS]^{n-1}I] \).

As fixed point combinators, they all have the same Lévy–Longo Tree \( \lambda x.x(x(x(\ldots))) \). However, using atomic clocked Lévy–Longo Trees we have shown in [11] that all these fixed point combinators are different, all of them are inconvertible: \( \vec{n} \neq \vec{m} \) implies \( Y^{\vec{n}} \neq \beta Y^{\vec{m}} \).
5 Future Work

We have employed the (atomic) clocked $\lambda$-calculus for proving that $\lambda$-terms are not convertible by showing that they have a different tempo in reducing to their infinite normal form. The method is however not yet strong enough to answer questions like: Is there a fixed point combinator $Y$ such that

$$Y = \beta \delta Y$$

$$Y = \beta Y \delta$$

where $\delta = \lambda ab. b(ab)$? R. Statman conjectured that no such fixed point combinator exists. However, this is still an open problem\footnote{B. Intrigila \cite{11} gave a proof that no such fixed point combinator exists. The proof however contains a serious gap, see further \cite{12}.}. It would be interesting to see whether methods in the flavour of the clocked $\lambda$-calculus could contribute to a solution. Note that every fixed point combinator fulfils the first equation: $Y = \delta Y$ if and only if $Y$ is a fixed point combinator, that is, all fixed point combinators are fixed points of $\delta$.

Furthermore, we are interested to investigate further applications of the clocked $\lambda$-calculus. For example, the clocks can be used as a measure of efficiency.

References

1 H.P. Barendregt. The Lambda Calculus. Its Syntax and Semantics, volume 103 of Studies in Logic and The Foundations of Mathematics. North-Holland, revised edition, 1984.
2 H.P. Barendregt and J.W. Klop. Applications of Infinitary Lambda Calculus. Information and Computation, 207(5):559–582, 2009.
3 C. Coquand and T. Coquand. On the Definition of Reduction for Infinite Terms. Comptes Rendus de l’Académie des Sciences. Série I, 323(5):553–558, 1996.
4 J. Endrullis, C. Grabmayer, D. Hendriks, J. W. Klop, and V. van Oostrom. Unique Normal Forms in Infinitary Weakly Orthogonal Rewriting. In Proc. 21st Int. Conf. on Rewriting Techniques and Applications (RTA 2010), volume 6, pages 85–102. Schloss Dagstuhl, 2010.
5 J. Endrullis, C. Grabmayer, D. Hendriks, J.W. Klop, and R. de Vrijer. Proving Infinitary Normalization. In TYPES 2008, volume 5497 of LNCS, pages 64–82. Springer, 2009.
6 J. Endrullis, C. Grabmayer, J.W. Klop, and V. van Oostrom. On Equal $\mu$-Terms. Theoretical Computer Science, 412(28):3175–3202, 2011.
7 J. Endrullis, H.H. Hansen, D. Hendriks, A. Polonsky, and A. Silva. A coinductive treatment of infinitary rewriting. CoRR, abs/1306.6224, 2013.
8 J. Endrullis, D. Hendriks, J. W. Klop, and A. Polonsky. Clocks for functional programs. In The Beauty of Functional Code, pages 97–126. Springer Berlin Heidelberg, 2013.
9 J. Endrullis, D. Hendriks, and J.W. Klop. Modular Construction of Fixed Point Combinators and Clocked Böhm Trees. In Proc. Symp. on Logic in Computer Science (LICS 2010), pages 111–119, 2010.
10 J. Endrullis, D. Hendriks, and J.W. Klop. Highlights in Infinitary Rewriting and Lambda Calculus. Theoretical Computer Science, 464:48–71, 2012.
11 J. Endrullis, D. Hendriks, J.W. Klop, and A. Polonsky. Discriminating Lambda-Terms using Clocked Böhm Trees. Logical Methods in Computer Science, 2012. In print.
12 J. Endrullis, D. Hendriks, J.W. Klop, and A. Polonsky. Clocked Lambda Calculus. Mathematical Structures in Computer Science, 2013. Accepted for publication.
13 J. Endrullis and A. Polonsky. Infinitary Rewriting Coinductively. In *Proc. Types for Proofs and Programs (TYPES 2012)*, volume 19, pages 16–27. Schloss Dagstuhl, 2013.

14 B. Intrigila. Non-Existent Statman’s Double Fixed Point Combinator Does Not Exist, Indeed. *Information and Computation*, 137(1):35–40, 1997.

15 J.W. Klop and R.C. de Vrijer. Infinitary Normalization. In *We Will Show Them: Essays in Honour of Dov Gabbay*, volume 2, pages 169–192. College Publ., 2005. Techn. report: [http://www.cwi.nl/ftp/CWIreports/SEN/SEN-R0516.pdf](http://www.cwi.nl/ftp/CWIreports/SEN/SEN-R0516.pdf).

16 G.D. Plotkin, 2007. Personal communication at the symposium for H. Barendregt’s 60th birthday.

17 Terese. *Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2003.

18 H. Zantema. Normalization of Infinite Terms. In *Proc. 19th Int. Conf. on Rewriting Techniques and Applications (RTA 2008)*, number 5117, pages 441–455, 2008.