CONFLUENCE & INFINITARY REWRITING

A bus tour with twelve stops

Bus driver: Jan Willem Klop

VU University Amsterdam
CWI Amsterdam

June 28, 2013 Eindhoven
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
The subject matter of this book, and how it is situated in some of the last century's developments. For a detailed account of the history of particular the first highlighted subject, see 'History of Lambda-calculus and Combinatory Logic', by Felice Cardone and J. Roger Hindley 2006.

Historical time-flow:

Confluence arose here.

Infinitary rewriting arose here.

Infinite Objects:

1900
- Foundations of Logic and Mathematics
- Theory of Types

1920
- Lambda Calculus, Combinatory Logic

1930
- Typed Lambda Calculi
- String Rewriting Systems (SRSs)
- Formalisation of Computability: Recursive Functions, Recursion Theory
- Algebraic Specifications, Abstract Data Types

1940
- Turing Machines

1970
- Term Rewriting Systems (TRSs)

1978
- Higher-order TRSs

1960–1980
- Functional Programming
- Type Theory, Theorem Provers, Proof Assistants (Automath, Coq, ...)
- Formalization and Verification of Mathematics

1980
- Communicating Processes, Process Algebra, CCS, CSP, ACP, \( \pi \)-calculus, Bigraphs
- Coalgebraic Techniques, Data & Codata, Recursion & Corecursion

2000
- Infinitary TRSs, infinitary Lambda Calculus
- Infinite Sequences, Productivity of Streams

1995–2005
1. REWRITING DICTIONARY

- Normal form
- Reduction cycle; loop if one step
- WN, weakly normalizing
- SN, strongly normalizing; terminating; noetherian
- NF, normal form property
- CR, Church-Rosser
- Equivalent: CR, Church-Rosser
- UN=, unique normal form property wrt =
- UN→, unique normal form property wrt →
- UN, weakly normalizing
- CR, Church-Rosser
- UN, weakly normalizing
- CR, Church-Rosser
- UN, weakly normalizing
- CR, Church-Rosser
- UN, weakly normalizing
Remark 3.2. Beware: The notion of sub.ARS and later of sub.TRS pertains only to shrinking the domain and shrink the reduction relation of the sub.ARS accordingly: It does not cover shrinking the signature by omitting some of the reduction relations.

Added in print: Joerg proposes to adopt Vincent's definition, then we can call the present notion 'closed'.

The following proposition from de Vrijer \[dV87\] is easy but important; it is the essence of the infinitary confluence modulo some set of undefined terms in Chapter xx:

**Proposition 3.1.** Let $A = (A, \rightarrow)$ and $B = (A, \rightarrow')$ be two ARSs. Suppose

\begin{itemize}
  \item[i] $A \subseteq B$
  \item[ii] $\exists \rightarrow' \subseteq \rightarrow$
  \item[iii] $\text{NF}(A) \subseteq \text{NF}(B)$
\end{itemize}

Then $B \not\overset{\text{UN}}{\rightarrow} \Rightarrow A \not\overset{\text{UN}}{\rightarrow}$.
\[ I, K, S, B \]

\[ \omega = \lambda x. xx \]

\[ \Omega = \omega \omega \]

\[ \delta = \lambda xy. y(xy) = SI, \text{Smullyan's Owl} \]

\[ \Delta = \delta^\omega = \delta(\delta(\delta(... = Y\delta \]

\[ Y_0 = \lambda f. (\lambda x. f(xx)) (\lambda x. f(xx)) \text{ Curry's fpc} \]

\[ Y_1 = (\lambda ab. b(aab)) (\lambda ab. b(aab)) \text{ Turing's fpc} \]

\[ Y_0 \delta = Y_1 \]
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Parallel Reduction à la Tait and Martin–Löf

\[ M \rightarrow M \]
\[ M \rightarrow M' \]
\[ \lambda x. M \rightarrow \lambda x. M' \]
\[ M \rightarrow M', \ N \rightarrow N' \]
\[ MN \rightarrow M'N' \]
\[ (\lambda x. M)N \rightarrow M'[x := N'] \]

We use the notation \( \rightarrow \) for parallel reduction. In the style of Tait and Martin–Löf, it is defined by the inductive clauses in Table 10. It characterizes complete developments, in the sense that \( M \rightarrow N \) if and only if there is a complete development from \( M \) to \( N \).

In Aczel [Acz78] the last clause is replaced by:

\[ M \rightarrow \lambda x. M', \ N \rightarrow N' \]
\[ MN \rightarrow M'[x := N'] \]

Now there is a complete \( \beta \)-superdevelopment form \( M \) to \( N \) if and only if \( M \rightarrow N \) according to Aczel's definition.

**Example 12.1.** In the first definition, due to Tait and Martin–Löf, we do not have \( III \rightarrow I \) (with \( I \equiv \lambda x. x \)) ; in Aczel's definition we do.

Likewise \( (\lambda xyz. xyz) abc \rightarrow abc \) and even \( II(\lambda xyz. xyz) abc \rightarrow abc \).

---

Dear [Name],

I would like to mention you a strikingly simple proof of the Church-Rosser theorem for the \( \lambda \)-calculus, due to Martin-Löf in his unpublished paper 'A theory of types', Stockholm 1971. In this paper an extension of the \( \lambda \)-calculus is considered. However the proof of the Church-Rosser theorem immediately carries over to the \( \lambda \)-calculus itself. The idea of the proof arises from cut-elimination properties of certain formal systems. In fact the Church-Rosser theorem is a kind of cut-elimination theorem, the transitivity of \( = \) in the \( \lambda \)-calculus corresponds to the cut.

The trick is to define a relation \( \geq \) between terms in such a way that

1) The transitive closure of \( \geq \) is the (classical) reduction relation \( \rightarrow \).

2) If \( M_1 \geq M_2, M_1 \geq M_3 \), then there exists a term \( M_4 \) such that \( M_2 \geq M_4 \) and \( M_3 \geq M_4 \).

From 1) and 2) the analogue of 2) for \( \geq \) can be derived. From this the Church-Rosser theorem easily follows.

Now \( \geq \) is defined as follows:

\[ M \geq M \]
\[ M \geq M', \ N \geq N' \rightarrow MN \geq M'N' \]
\[ M \geq M', \ N \geq N' \rightarrow (\lambda x M)N \geq (\lambda x M')N' \]
\[ \forall \ M \geq M', \ N \geq N' \rightarrow \text{if } FV(M) \cap \text{BV}(N) = \emptyset \]

\[ (\{x/N\}M \text{ stands for the result of substituting } N \text{ in the free occurrences of } x \text{ in } M ; FV(M) \text{ resp. } \text{BV}(N) \text{ is the set of free resp. bound variables of } M) \]

It is clear that \( \geq \) satisfies 1). A simple inductive proof shows that \( \geq \) also satisfies 2).

In the same way the Church-Rosser theorem can be proved when \( n \)-reduction is included.

Sincerely yours,

[Name]

Mathematisch Instituut
Budapestlaan 6
Utrecht- De Uithof
Holland
1924. "Über die Bausteine der mathematischen Logik"

Moses Schönfinkel
Ever since the original proof of the confluence of $\lambda\beta$-reduction in [Church and Rosser, 1936], a general feeling had persisted in the logic community that a shorter proof ought to exist. The work on abstract confluence proofs described in §5.2 did not help, as it was aimed mainly at generality, not at a short proof for $\lambda\beta$ in particular.

For CL, in contrast, the first confluence proof was accepted as reasonably simple; its key idea was to count the simultaneous contraction of a set of non-overlapping redexes as a single unit step, and confluence of sequences of these unit steps was easy to prove, [Rosser, 1935, p.144, Thm. T12].

Then in 1965 William Tait presented a short confluence proof for CL to a seminar on $\lambda$ organized by Scott and McCarthy at Stanford. Its key was a very neat definition of a unit-step reduction by induction on term-structure. Tait’s units were later seen to be essentially the same as Rosser’s, but his inductive definition was much more direct. Further, it could be adapted to $\lambda\beta$. (This possibility was noted at the seminar in 1965, see [Tait, 2003, p.755 footnote]). Tait did not publish his method directly, but in the autumn of 1968 he showed his CL proof to Per Martin-Löf, who then adapted it to $\lambda\beta$ in the course of his work on type theory and included the $\lambda\beta$ proof in his manuscript [Martin-Löf, 1971b, pp.8–11, §2.5].

Martin-Löf’s $\lambda\beta$-adaptation of Tait’s proof was quickly appreciated by other workers in the subject, and appeared in [Barendregt, 1971, Appendix II], [Stenlund, 1972, Ch. 2] and [Hindley et al., 1972, Appendix 1], as well as in a report by Martin-Löf himself, [Martin-Löf, 1972b, §2.4.3].

In $\lambda$, each unit step defined by Tait’s structural-induction method turned out to be a minimal-first development of a set of redexes (not necessarily disjoint). Curry had introduced such developments in [Curry and Feys, 1958, p.126], but had used them only indirectly; Hindley had used them extensively in his thesis, [Hindley, 1969a, p.547,”MCD”], but only in a very abstract setting. They are now usually called parallel reductions, following Masako Takahashi. In [Takahashi, 1989] the Tait-Martin-Löf proof was further refined, and the method of dividing reductions into these unit steps was also applied to simplify proofs of other main theorems on reductions in $\lambda$.

Tait’s structural-induction method is now the standard way to prove confluence in $\lambda$ and CL. However, some other proofs give extra insights into reductions that this method does not, see for example the analysis in [Barendregt, 1981, Chs. 3, 11–12].
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Consideration is given to the equational theory $\lambda\pi$ of lambda calculus extended with constants $\pi$, $\pi_0$, $\pi_1$ and axioms for subjective pairing:

$\pi_0(\pi XY) = X$, $\pi_1(\pi XY) = Y$, $\pi(\pi_0 X)(\pi_1 X) = X$.

The reduction system that one obtains by reading the equations are reductions (from left to right) is not Church-Rosser. Despite this failure, the author obtains a syntactic consistency proof of $\lambda\pi$ and shows that it is a conservative extension of the pure $\lambda$ calculus.

De Vrijer 1989

Extending the lambda calculus with surjective pairing is conservative

Klop, de Vrijer 1989: but UN holds
A Question of Balance (The Moody Blues 1970)

\[
\delta x x \rightarrow_{\delta H} x
\]

\[
Cx \rightarrow \varepsilon(\delta x(Cx))
\]

\[
A \rightarrow CA
\]

\[
A \rightarrow CA \rightarrow \varepsilon(\delta A(CA)) \rightarrow \varepsilon(\delta(CA)(CA)) \rightarrow \varepsilon(CA)
\]

\[C(\varepsilon(CA))\]

**Question:** what about \(\lambda^\infty \beta \delta\) and \(\lambda^\infty \beta \pi\) ?
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence lost
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
A General Church-Rosser Theorem

Peter Aczel

We prove the Church-Rosser theorem in a general framework. Our result easily yields the standard result for the lambda calculus but also has wider application.

§1. The Main Theorem

1.1 An expression system consists of an infinite set of variables and a set of forms. Each form has an arity, i.e., a finite sequence \( k_1, \ldots, k_m (m \geq 0) \) of natural numbers. If \( m = 0 \) the form is a constant. If \( k_1 = \cdots = k_m = 0 \) the form is simple.

Expressions are inductively generated using the two rules:

1) Every variable is an expression.

2) If \( F \) is a form with arity \( k_1, \ldots, k_m (m \geq 0) \) and \( a_1, \ldots, a_m \) are expressions then \( F(\overline{a_i})a_1, \ldots, (\overline{a_m})a_m \) is an expression, where for \( i = 1, \ldots, m \) \( \overline{a_i} \) is a list of \( k_i \) variables.

The expression generated by 2) is said to have form \( F \) and parts \( a_1, \ldots, a_m \). Free and bound occurrences of variables are defined in the usual way, so that occurrences of a variable in the list \( \overline{a_i} \) that are free in \( a_i \) become bound in \( F(\overline{a_i})a_1, \ldots, (\overline{a_m})a_m \). Alphabetic variants of expressions are identified in the standard way.

Below we shall usually write \( F(a_1, \ldots, a_m) \) instead of \( F(\overline{a_i})a_1, \ldots, (\overline{a_m})a_m \). It must be kept in mind that with this abuse of notation a variable that is free in \( a_i \) can become bound in \( F(a_1, \ldots, a_m) \).

Also, an expression \( F(a_1, \ldots, a_m) \) may be the same as an expression \( F(b_1, \ldots, b_m) \) while \( a_i \) is not the same as \( b_i \) for \( i = 1, \ldots, m \).

1.2 We shall be concerned with a partial function on the expressions which we shall call a contraction operation. An expression in the domain of the operation will be called a reducible and its value under the operation will be called the contraction of the reducible. We shall insist that no variable is a reducible. Each contraction operation generates a relation of definitional equality.
The methods to prove confluence of orthogonal higher-order rewriting systems both can be adapted to the case where critical pairs are allowed, but only if they are of the form \((s, s)\). Such a critical pair is said to be trivial. The notion of trivial critical pair is used to define the class of weakly orthogonal higher-order rewriting systems; the definition is analogous to the one for the first-order case.

**Definition 3.** A higher-order rewriting system is weakly orthogonal if it is left-linear and all its critical pairs are trivial.

Examples of weakly orthogonal rewriting systems that are not orthogonal are \([a \rightarrow b, f(a) \rightarrow f(b)]\) and \([f(x) \rightarrow f(b), f(a) \rightarrow f(b)]\). Moreover, lambda calculus with both \(\beta\)-and \(\eta\)-reduction is a weakly orthogonal rewriting system.

---

**Orthogonal higher-order rewrite systems are confluent**

**Abstract**

The results about higher-order critical pairs and the confluence of OHRSs provide a firm foundation for the further study of higher-order rewrite systems. It should now be interesting to lift more results and techniques both from term-rewriting and \(\lambda\)-calculus to the level of HRSs. For example termination proof techniques are much studied for TRSs and are urgently needed for HRSs; similarly the extension of our result to weakly orthogonal HRSs or even to Huet’s “parallel closed” systems is highly desirable.

Conversely, a large body of \(\lambda\)-calculus reduction theory has been lifted to CRSs [10] already and should be easy to carry over to HRSs.

Finally there is the need to extend the notion of an HRS to more general left-hand sides. For example the *eta*-rule for the *case*-construct on disjoint unions [15] \(\text{case}(U, \lambda x. F(\text{inl}(x)), \lambda y. G(\text{inr}(y))) \rightarrow F(U)\) is outside our framework, whichever way it is oriented.
Van Oostrom, van Raamsdonk 1994

These rules can be written in the formalism of Combinatory Reduction Systems. They then take the following form:

\[ \text{el}(\text{inl}(Z), [x]Z_0(x), [y]Z_1(y)) \rightarrow Z_0(Z) \]
\[ \text{el}(\text{inr}(Z), [x]Z_0(x), [y]Z_1(y)) \rightarrow Z_1(Z) \]

\[ \lambda \beta \eta \vdash \text{CR} \]
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Modulo unsolvables:

$\lambda \beta \Omega \models CR$

\[ (\lambda x.Z(x))Z' \rightarrow Z(Z') \quad (\beta) \]
\[ M \rightarrow \Omega \quad \text{if } M \neq \Omega \text{ is unsolvable (uns)} \]
\[ \Omega M \rightarrow \Omega \quad (\Omega_i) \]
\[ \lambda x.\Omega \rightarrow \Omega \quad (\Omega_d) \]
In Barendregt 84: section 15.2, 8 pages

$\lambda \beta \eta \Omega \models CR$

later question:
$\lambda^\infty \beta \eta \Omega \models CR^\infty$
Werkweek λ-calculus in de molen te Varik

juni 75.
Fig 15.7: Infinitary reduction graph of the term \( SII \left( SII \right) \), not a closed graph.

The red reduction steps are root steps. All infinite reductions in this graph are divergent. The accumulation or limit points in the euclidean metric, as well as in the tree metric, at the east and south side, are themselves not \( \omega \)-reducts, hence not contained in this \( \omega \)-graph.

Example 15.4. The CL(term \( SII \left( SII \right) \) has the infinite reduction graph displayed in Figure 15.7. Abbreviating \( \omega = SII \) the terms at the nodes of this graph are \( I^n \omega \left( I^m \omega \right) \) for \( n, m \geq 1 \).

Here are some observations:

.i: All the terms in this reduction graph are root (active - but not hypercollapsing).

.ii: There are continuum many infinite reductions contained in this reduction graph; all are divergent; in particular they are root (active).
Statman 1978

instead of $\text{MA} \Rightarrow_\beta N$, write $M \xrightarrow{A} N$

$M \Rightarrow N$: $M$ is more solvable than $N$.

\begin{itemize}
  \item order 0 \quad $\Omega$
  \item order 1 \quad $\lambda.x.\Omega$
  \item order $\infty$ \quad $\Upsilon K \equiv \lambda.x_1.x_2.x_4.$
\end{itemize}

Every countable poset is embeddable in poset of unsolvables.
head normalization theorems

$\Omega_{\text{BeT}}$ (mute terms, no root stable form) = $\square \diamond \text{root}$

$\Omega_{\text{LLT}}$ (no weak head normal form) = $\square \diamond \text{lazy} = \square \text{lazy}$

$\Omega_{\text{BT}}$ (no head normal form, unsolvables) = $\square \diamond \text{head} = \square \text{head}$

$\square \diamond \text{spine} = \square \text{spine}$
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Rewrite, rewrite, rewrite, rewrite, ...*

Nachum DERSHOWITZ†
Department of Computer Science, University of Illinois at Urbana-Champaign, 1304 West Springfield Ave., Urbana, IL 61801–2987, U.S.A.

Stéphane KAPLAN‡
Department of Computer Science, Bar-Ilan University, Ramat-Gan 52100, Israel
Laboratoire de Recherche en Informatique, Université de Paris-Sud, Orsay 91405, France

David A. PLAISTED§
Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27514–3175, U.S.A.

Communicated by
Received
Revised

Abstract. We study properties of rewrite systems that are not necessarily terminating, but allow instead for transfinite derivations that have a limit. In particular, we give conditions for the existence of a limit and for its uniqueness and relate the operational and algebraic semantics of infinitary theories. We also consider sufficient completeness of hierarchical systems.

Is there no limit?
—Job 16:3
\[ F(0) \rightarrow P \rightarrow \ldots \]

\[
\begin{align*}
0 & \quad F \\
S & \quad 0 \\
\end{align*}
\]

\[
\begin{align*}
0 & \quad P \\
S & \quad P \\
\quad S & \quad P \\
\quad S & \quad 0 \\
\end{align*}
\]

Limit: infinite sequence of natural numbers

\[ F(x) \rightarrow P(x, F(S(x))) \]
Transfinite reduction sequence of length $\omega + \omega$

\[ F(x) \rightarrow P(x, F(S(x))) \]
Preliminaries: some tools of the trade

\[ \alpha - \beta = (\alpha \beta) + \alpha \]

\[ \alpha \lambda = \mu < \lambda (\alpha \mu) \text{ if } \lim \lambda \]

Power:

\[ \alpha = (\alpha \beta) + \alpha \]

\[ \alpha \lambda = \mu < \lambda (\alpha \mu) \text{ if } \lim \lambda \]

Example 2.3.

Note that \( \omega + \omega = \omega \cdot 2 \).

Also product is not commutative: \( \omega \cdot \omega = \omega + \omega \).

Exercise 2.11.

Albert Visser: Often one draws ordinals as 'dot diagrams', see Benedikt Löwe's Visualizations of ordinals or telephone poles. Conway and Guy's The book of numbers or David Magor's wikipedia: As such dot diagrams of ordinals suggest some ordinals can be mapped embedded into the segment of real numbers \([-\infty, 0]\) in an order-preserving way.

Given that there are uncountably many real numbers in \([-\infty, 0]\) one might think that there is also room for embedding some uncountable ordinals in \([-\infty, 0]\): However:
Cauchy converging reduction sequence: activity may occur everywhere

Strongly converging reduction sequence, with descendant relations

difference between CC and SC: looping terms

Kennaway-de Vries 1992; De Vrijer, Grabmayer, Endrullis, Hendriks, Simonsen 2012
strong convergence: redex depth to infinity

The sequence is called strongly convergent if the conditions are fulfilled for every limit ordinal $\lambda \leq \alpha$; In this case we write $t(\alpha)$ or $t(\rightarrow \alpha)$ to ex, explicitly indicate the length $\alpha$ of the sequence;

There are several reasons why strong convergence is beneficial; the foremost being that in this way we can define the notion of descendant in the past also called residual over limit ordinals; Also the well-known Parallel Moves Lemma and the Compression Lemma fail for weak convergence; It is further easy to establish that strongly convergent reductions can have any countable length; weakly convergent reductions can have any length: as the one rule TRS with $C \rightarrow C$ demonstrates;

The notion of normal form: which now may be an infinite term: is unproblematic: it is a term without a redex occurrence;
<table>
<thead>
<tr>
<th>Finitary rewriting</th>
<th>Infinitary or transfinite rewriting</th>
</tr>
</thead>
<tbody>
<tr>
<td>finite reduction</td>
<td>strongly convergent reduction</td>
</tr>
<tr>
<td>infinite reduction</td>
<td>divergent reduction (&quot;stagnating&quot;)</td>
</tr>
<tr>
<td>normal form</td>
<td>possibly infinite normal form</td>
</tr>
<tr>
<td>CR: two coinitial finite reductions can be prolonged to a common term</td>
<td>CR(^\infty): two coinitial strongly convergent reductions can be prolonged by strongly convergent reductions to a common term</td>
</tr>
<tr>
<td>UN: two coinitial reductions ending in normals forms, end in the same normal form</td>
<td>UN(^\infty): two coinitial strongly convergent reductions ending in (possibly infinite) normal forms, end in the same normal form</td>
</tr>
<tr>
<td>SN: all reductions lead eventually to a normal form</td>
<td>SN(^\infty): all reductions lead eventually to a possibly infinite normal form, equivalently: there is no divergent reduction</td>
</tr>
<tr>
<td>WN: there is a finite reduction to a normal form</td>
<td>WN(^\infty): there is a strongly convergent reduction to a possibly infinite normal form</td>
</tr>
</tbody>
</table>
zero times infinity

\[ A(x, 0) \rightarrow x \]
\[ A(x, S(y)) \rightarrow S(A(x, y)) \]
\[ M(x, 0) \rightarrow 0 \]
\[ M(x, S(y)) \rightarrow A(M(x, y), x) \]
\[ \infty \rightarrow S(\infty) \]
Basics of infinitary rewriting

\[ \omega \cdot 1 \omega \cdot 2 \omega \cdot 3 \omega \cdot 4 \omega \cdot 5 \omega \cdot 6 \omega \cdot 7 \omega \cdot 8 \omega \cdot 9 \omega \cdot 10 \omega \cdot 11 \omega \cdot 12 \omega \cdot 13 \omega \cdot 14 \omega \cdot 15 \omega \cdot 16 \omega \cdot 17 \omega \cdot 18 \omega \cdot 19 \omega \cdot 20 \]\n
The sequence is called **strongly convergent** if the conditions

\[ 6i7 \text{ and } 6ii7 \]

are fulfilled for every limit ordinal \( \lambda \leq \alpha \); In this case we write \( t(\alpha) \rightarrow t0 \alpha \text{} \). There are several reasons why strong convergence is beneficial; the foremost being that in this way we can define the notion of **descendant** (in the past also called **residual**) over limit ordinals; Also the well-known **Parallel Moves Lemma** and the **Compression Lemma** fail for weak convergence; It is further easy to establish that strongly convergent reductions can have any countable length; weakly convergent reductions can have any length: as the one rule TRS with \( C \rightarrow C \) demonstrates.

The notion of **normal form**; which now may be an infinite term; is unproblematic: it is a term without a redex occurrence.

Example 15.1 (Zero times infinity). Let us discuss all the concepts introduced so far by means of the following reduction rules for addition and multiplication due to...
not $\text{CR}^\infty$

$A(x) \rightarrow x$
$B(x) \rightarrow x$
$C \rightarrow A(B(C))$

---

### Failure of infinitary confluence

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Diagram a" /></td>
<td><img src="image" alt="Diagram b" /></td>
</tr>
</tbody>
</table>

The diagrams illustrate the failure of infinitary confluence for the given rewrite system.
\[S_{xyz} \rightarrow xz(yz)\]
\[K_{xy} \rightarrow x\]
\[\rightarrow @(\rightarrow @(S, x), y), z) \rightarrow @(\rightarrow (x, z), \rightarrow (y, z))\]
\[\rightarrow @(\rightarrow (K, x), y) \rightarrow x\]

*collapsing contexts*

*Failure of infinitary confluence for Combinatory Logic*
Failure of $CR^\infty$

$A(x) \rightarrow x$

$B(x) \rightarrow x$

$C \rightarrow A(B(C))$
Infinitary rewriting

\[ (x) \rightarrow xB(x) \rightarrow xC \rightarrow A(B(C)) \]

The first two rules are so-called collapsing rules, by virtue of their right-hand side being a single variable. Now we have reductions \( C \mapsto A^\omega \) and \( C \mapsto B^\omega \). Figure 15.4 depicts the tiling diagram for these reductions. However, the infinite terms \( A^\omega, B^\omega \) only reduce to themselves; hence CR\(_\infty\) fails.

We note that both terms are reduction loops, i.e. reduction cycles of length one. This is not a coincidence. We will see later that any term \( t \) that is not CR\(_\infty\), must be 'close to' a looping term; in fact it must have a looping term in its family, defined as the set of subterms of the reducts.

\[ \text{ABC-counterexample in perspective: euclidean distance} = \text{tree distance} \]
**Example 2.4.** The ‘ABC-example’ that we saw in the preceding example also works in the much more important rewrite system Combinatory Logic CL, with the usual three basic combinators I, K, S and their corresponding reductions rules (see, e.g., Barendregt [2]), and also in infinitary λ-calculus that we will consider in more detail in the next section. The figure on the right, with the infinite collapsing tower of two different collapsing contexts \( K \Box K \) and \( K \Box S \) shows how the ABC-counterexample can be simulated using a fixed-point construction in those calculi. To see that this is indeed a CR\( ^\infty \)-counterexample, note that \( \mu x.K(KxS)K \rightarrow \mu x.KxS \) and also \( \mu x.K(KxS)K \rightarrow \mu x.KxK \), while \( \mu x.KxS \) and \( \mu x.KxK \) only reduce to themselves (in any countable ordinal number of steps, by the way).
Ketema-Simonsen, with $\mathcal{U}N^\infty$ as corollary.
For all terms $t$ in an orthogonal TRS, we have

$$Fam(t) \cap HC = \emptyset \ \Rightarrow \ \text{CR}^\infty(t)$$
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
$\lambda^\infty : \text{not PML}^\infty$

$\omega_I \equiv (\lambda x. I(xx))$

$\omega \equiv \lambda x. xx$

$YI \rightarrow \omega_I \omega_I$

$I^\omega \equiv$

For infinitary lambda calculus Parallel Moves Lemma PML$^\infty$ fails, hence also CR$^\infty$
Let $M \equiv \lambda x.Kxw$ and $Z = YM$. Then

curious: in the limit, $w$ is fixed as a free variable, all reducts are open terms
A SIMPLE PROOF

Curry’s fpc

Turing’s fpc
\( Y_0: \lambda f. (x.f(xx)(\lambda x.f(xx))) \)

\( Y_1: (\lambda ab. b(aab)) (\lambda ab. b(aab)) \)

\( Y_0 \text{(SI)} \rightarrow Y_1 \)

Exercise. Prove that \( Y_0 \not\equiv_\beta Y_1 \)
\[
Yx \rightarrow\rightarrow x(Yx) \rightarrow\rightarrow x^2(Yx) \rightarrow^\omega x^\omega \equiv x(x(x(x(\ldots)))
\]

\[
BY \equiv (\lambda abc. a(bc)) Y \equiv_\infty Y^\omega
\]

\[
BYS \equiv (\lambda abc. a(bc)) YS
\]

\[
\lambda bc. Y(bc)
\]

\[
\lambda bc. (bc)^\omega \equiv \lambda cz. (cz)^\omega
\]
playing with infinite lambda terms: infinite fixed point combinators

twinkle = Δ = δ^ω = δ(δ(δ(δ ... 

Δx ≡ δΔx → β → β x(Δx)

(SS)^ωSSSI, another infinite fpc

ΔΔ is an interesting term. We have

ΔΔ → Δ^ω → (Δ^ω)^ω → ((Δ^ω)^ω)^ω → ...

See Figure 8. Somewhat surprisingly, ΔΔ does have a normal form, viz. μx.xx; and moreover ΔΔ has the property SN^∞. To see that μx.xx is indeed the normal form, one may consider the reduction

ΔΔ → (Δ^ω)^ω = Δ^ω((Δ^ω)^ω) → (Δ^ω)^ω((Δ^ω)^ω) → ...

and check that the reductions involved do not employ root redexes. (Only in the reduction ΔΔ → Δ^ω a root step is present; in the ‘later’ reductions there are no root steps.) In fact we have a strongly convergent reduction

ΔΔ → Δ^ω → (Δ^ω)^ω → ((Δ^ω)^ω)^ω → ... → μx.xx

The term ΔΔ has uncountably many reducts. It has reductions of any countable ordinal length. It is SN^∞ with μx.xx as its unique normal form. This normal form is in fact a Berarducci tree. The example of ΔΔ was also mentioned in [4]. SN^∞ can be proved as follows: We have CR^∞ as there are no collapsing rules in this TRS, which is a fragment (sub-TRS) of CL. Since there is a normal form, we have WN^∞. Hence, SN^∞ follows by the equivalence SN^∞ ↔ WN^∞ as global properties of TRSs.
The infinitary $\beta$-reduction $\rightarrow^\infty_\beta$ has the infinitary normal form property $\text{NF}_\infty^\infty$, that is, for all $M, N \in \text{Ter}_\infty^\infty(\lambda)$ with $N$ a normal form and $M \ (\leftarrow^\beta \cup \rightarrow^\infty_\beta)^* \ N$ we have $M \rightarrow^\infty_\beta N$. In a picture:

```
M ←――→ * N  a normal form
```

Actually the following property is sufficient:

```
M ―――→ N  a normal form
```

We obtain infinitary unique normal forms $\text{UN}_\infty^\infty$ as a direct corollary.
playing with infinite lambda terms: looping lambda terms

Fig. 17.3: An infinite looping λ-term.

Fig. 17.4: Another counterexample to CR∞ of λ∞β-calculus.

Example 17.2. Now we consider the ARS A = (Nω, →) consisting of the streams of extended natural numbers N = N ∪ {∞}. The reduction relation is again the addition of two consecutive stream entries, now with the understanding that n + ∞ = ∞ + n = ∞. Now consider the stream ∞111..., corresponding in fact to the infinite looping term in Figure 17.3. Also the reduction graph of this looping term is isomorphic to that of the stream as mentioned. That it is non-CR∞ is a nice and nontrivial exercise, left to the reader.

Todo: The following figure has still to be described and motivated in the text.
**Theorem 13.2.6.** In infinitary $\lambda$-calculus, a term is root looping if and only if it is of one of the following forms:

(i) $\Omega$

(ii) $|^{\omega}$

(iii) $BB$ where $B$ is the infinite solution of $B = \lambda x. xB$,

(iv) $(\lambda v_0.(\lambda v_1.(\lambda v_2. \ldots) t_2) t_1) t_0$ such that $t_i$ is obtained from $t_{i+1}$ by replacing $v_0$ by $t_0$ and all variables $v_{j+1}$ by $v_j$. We call such a term a cascade.

![Diagram of cascades](image-url)

**Figure 13.2:** The shape of cascades; here $\pi$ stands for replacing all variables $v_j$ by $v_{j+1}$ followed by replacing an arbitrary (possibly infinite) number of occurrences of $t_0$ by $v_0$. 
different ways to count depth

001-depth 1
{l,d}-steps don’t count

101-depth 4
{l}-steps don’t count

111-depth 7
all steps count
typical terms in the three domains
Berarducci Trees

BeT(Ω₃) =

BeT(Ω₃I) =

BeT(YΩ₃) =

not easy  easy  easy for closed normal forms;
open problem for general terms
Restoring infinitary confluence by quotienting undefined terms

It is a pitfall to think that all normal forms from $\lambda_\infty^{\beta}\Omega$ are $BT$'s. To see what is the difference, we formulate the following theorem. Of course one can characterize the normal forms from $\lambda_\infty^{\beta}\Omega$ calculus in a negative way, by stating that they do not contain the pattern of a $\beta$-redex; but this does not give insight in their structure, from what components they are built.

Now we see that the components with infinite spine are not possible in a $BT$ tree. On the other hand, the normal forms from $\lambda_\infty^{\beta}\Omega$ calculus are $BT$'s, Berarducci trees. Below we will use this fact.

It is interesting to consider the question what $BT$'s are actually realizable by finite $\lambda$-terms, i.e., which of them are finitely generated. Note that we can compose continuum many $BT$'s with their building blocks as given in Figure 18.9, or equivalently, as normal forms of infinitary $\lambda_\infty^{\beta}\Omega$ calculus. This question is answered in [Bar8], Theorem [)-[-], in the way one would expect: all and only the computably enumerable $BT$'s are finitely generated, of course provided they have only finitely many free variables. Interestingly, this characterization is much more subtle for the $\lambda_I$ version of the $BT$'s; it then requires moreover the computability of a variable indicator, see [Bar8], Theorem [)-[-5]. It would be interesting to do this exercise also for the case of $LLT$ and $BT$-

Theorem 18.1. The normal forms from $\lambda_\infty^{\beta}\Omega$-calculus are built (coinductively) from the four building block types as in Figure 18.9, namely a variable, $\text{hnf-contexts}$, the Omnivore, and $d_*\omega$-terms.

![Building blocks for infinitary lambda normal forms](image)
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Definition 20. Let $M \in \text{Ter}_\infty^{\lambda \perp}$ then we define the Bohm Tree $\text{BT}(M)$, the Levy-Longo Tree $\text{LLT}(M)$, and the Berarducci Tree $\text{BeT}(M)$ coinductively by

$$\text{BT}(M) = \begin{cases} \lambda \bar{x}.y \text{BT}(M_1) \ldots \text{BT}(M_m) & \text{if } M \text{ has hnf } \lambda \bar{x}.y M_1 \ldots M_m, \\ \bot & \text{otherwise.} \end{cases}$$

$$\text{LLT}(M) = \begin{cases} \lambda x.\text{LLT}(M') & \text{if } M \text{ has whnf } \lambda x.M', \\ \bot & \text{otherwise.} \end{cases}$$

$$\text{BeT}(M) = \begin{cases} y & \text{if } M \rightarrow y, \\ \lambda x.\text{BeT}(N) & \text{if } M \rightarrow \lambda x.N, \\ \text{BeT}(M_1) \text{BeT}(M_2) & \text{if } M \rightarrow M_1 M_2 \text{ such that } M_1 \text{ is of order 0,} \\ \bot & \text{in all other cases (i.e., when } M \text{ is mute).} \end{cases}$$

Coinductive definition of BT, LLT, BeT
The strategic redexes: root, lazy, head, and spine.

The spine of a \( \lambda \)-term, finite or infinite, is the maximal \( \beta \)-branch (redexes whose pattern is on the spine are spine redexes). The uppermost one is the head redex. It is the root redex if its root is that of the whole term.

In the BT sense, there may be several redexes at depth \( \ell \); the spine redexes; the uppermost one in the syntactic sense is the head redex. In the \( \lambda \)-LTT sense, there is at most a unique redex at depth \( \ell \), which is the lazy redex. In the \( \beta \)-LTT sense, there is at most one; unique; redex at depth \( \ell \), the root redex.

An elegant characterization of depth-' redexes is due to Fer-Jan de Vries. Depending on which of the derivation rules \( d \), \( l \), \( r \) is adopted, the inference systems given in Table 18(3) allows just the redexes of \( dlr \)-depth \( \ell \) to be contracted; e.g., with rules \( \beta \), \( d \), \( l \) we have spine reduction; with \( \beta \), \( l \) we have lazy reduction; and with only \( \beta \) we have root reduction. The normal forms for these three notions of reduction are the hnf's; the whnf's; and the non-redexes; respectively.

(\[ \lambda x. M (\lambda x. N) \rightarrow M (\lambda x. N) \]\(\beta\)
(\[ \lambda x. M \rightarrow \lambda x. N \]\(d\)
(\[ M Z \rightarrow N Z \]\(l\)
(\[ Z M \rightarrow Z N \]\(r\)

Table 18(3): Characterizing redexes at depth \( \ell \). The rules \( d \), \( l \), \( r \) are also known as \( \xi \), \( \nu \), \( \mu \)..

The typical redexes:

redex is root \( \Rightarrow \) lazy \( \Rightarrow \) head \( \Rightarrow \) spine
notions of undefinedness, with a caveat

3.2.5. Lambda theories
The syntactic analysis of finite and infinitary λ-calculus sheds more light on some of the main models of λ-calculus. It is long known that the theory of $P_{\omega}$ is that of $BT$ equality. It is interesting that we can split up this equality in two 'orthogonal' components: on the one hand there is equating all unsolvables terms $M$ with $BT M = \Omega$ called the theory $H$ in [u]; on the other hand, there is the 'infinite expansion' given by the theory of $\lambda_\omega \beta$. The supremum of both theories is the theory of $P_\omega$. Figure 19 gives the partial order of these theories for the three different frameworks. The $B$ in that figure is the theory of $BT$ equality described first in [uo Section 2.1]. This can be seen as a precursor of our $\lambda_\infty \beta \Omega$'s are there applied to each other by first taking their projections up to depth $n$ and then applying these finite $BT$'s to each other and finally taking the limit. It would be an interesting student assignment to prove the equivalence with the more direct setup via the present $\lambda_\infty \beta \Omega$ calculus.

3.2.6. Restoring infinitary confluence by quotienting undefined terms
It is a pitfall to think that all normal forms from $\lambda_\infty \beta$ calculus are $BT$'s. To see what is the difference, we formulate the following theorem. Of course, one can characterize the normal forms from $\lambda_\infty \beta$ calculus in a negative way, by stating that they do not contain the pattern of a $\beta$ predex; but this does not give insight in their structure, from what components they are built. Figure 18. Now we see that the components with infinite spine are not possible in a Bohm tree. On the other hand, the normal forms from $\lambda_\infty \beta$ calculus are $BeT$'s, Berarducci trees. Below we will use this fact. It is interesting to consider the question what $BT$'s are actually realizable by finite $\lambda$ terms. iq which of them are finitely generated? Note that we can compose continuum many $BT$'s with their building blocks as given in Figure 19. or equivalently, as normal forms of infinitary $\lambda_\infty \beta \Omega$ calculus. This question is answered in [uo Theorem 3.7] in the way one would expect; all and only
lambda theories compared

**Question:**
Do we have LLT and BeT versions of $P_\omega$?

**Question:**
can we interpret $\lambda^\infty \beta \Omega$ in $P_\omega$?
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
clock behaviour of fpc in Böhm sequence of fpc’s

\[ Y_0, Y_0 \delta, Y_0 \delta \delta, Y_0 \delta \delta \delta, Y_0 \delta \delta \delta \delta, \ldots \]

\[ Y_3 \equiv Y_0 \delta \delta \delta \rightarrow^7_h \lambda a.a(\omega_\delta \omega_\delta \delta \delta a) \]

\[ \lambda a.a \square \]

\[ \omega_\delta \omega_\delta \delta \delta a \rightarrow^7_h a(\omega_\delta \omega_\delta \delta \delta a) \]

\[ a \square \]
reduces to reducts of has the property that So every redex j

The clocked Böhm trees of strictly intermediate between one position in the

how they are computed in what 'tempo'

is given by inspecting the general circumstances

\( \lambda \)

Proof.

An example of a term that is not simple is

The idea is that we will extract from a

\( Y \)

by an application of Intrigila's theorem stating that for no f

\( \beta \)

\( \delta \)

Because

\( k \)

\( \omega \)

\( \eta \)

\( \beta \)

\( \delta \)

\( \approx \)

Figure 21: Clocked Böhm trees of \( Y_0 f \) and \( Y_1 f \)

Figure 14:3: Clocked Böhm trees of \( BY_0 \) and \( BY_0 S \).
Clocked Lambda Calculus

\[(\lambda x.M)N \rightarrow \tau(M[x:=N])\]
\[\tau(M)N \rightarrow \tau(MN)\]

The \(\tau\)'s are ticks of the clock (measure of efficiency).
Properties: orthogonal, SN\(\infty\), CR\(\infty\), UN\(\infty\)

Normal forms are clocked Lévy–Longo trees:

\[nf(Y_0f) \equiv \tau^2\]
\[nf(Y_1f) \equiv \tau^2\]

\(nf(Y_0f)\) and \(nf(Y_1f)\) are different because they have different clocks.
\(Y_0 \neq Y_1\)
clocked lambda theories

Exercise.

(i) in $\lambda^\infty \beta$ there is only one Ogre, Omnivore;

(ii) in $\lambda \beta$ there are infinitely many, i.p. all $Y_nK$ are different
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
\[ \text{PS} \rightarrow \varepsilon \quad \text{SP} \rightarrow \varepsilon \]

where \( \varepsilon \) is the empty word. This system has two trivial critical pairs:

\[ \text{P} \leftarrow \text{PSP} \rightarrow \text{P} \quad \text{S} \leftarrow \text{SPS} \rightarrow \text{S}, \]

and hence is weakly orthogonal.

Now consider the term \( \psi \) defined as follows:

\[ \psi = \text{P} \text{SSPPPSSSSPPPPPSSSSSSS} \ldots \]

\[ S^\omega \leftrightarrow \psi \rightarrow P^\omega \]
Fig. 17.6: Counterexample to $\text{UN}^\infty$ in $\lambda^\infty \beta \eta$.

Question:

$\lambda^\infty \beta \eta \Omega \models \text{CR}^\infty$
1. Introduction
2. Lambda and CL: basic confluence
3. Surjective Pairing: confluence lost
4. Confluence of a higher order
5. Lambda with black holes: confluence
6. Confluence lost in infinity
7. The threefold path
8. Black holes to the rescue
9. The rhythm of lambda terms
10. Getting rid of ordinals
11. Infinity and eta: total breakdown
12. A lambda universe
Other bus tours of the RTA-IWC-WIR company

Abstract rewriting and confluence, decreasing diagrams, De Bruijn - Van Oostrom

Alternative set-up of infinitary rewriting, Kahrs, ideal completion (Bahr), coinductive definition, Endrullis, Polonsky et al.

Infinitary Rewriting Coinductively

\[ \rightarrow^* = \mu x. \forall y. (\rightarrow \varepsilon \cup \bar{x})^* \circ \bar{y} \]

\[ \bar{R} = \{ \langle f(s_1, \ldots, s_n), f(t_1, \ldots, t_n) \rangle \mid s_1 R t_1, \ldots, s_n R t_n \} \cup \text{id} \]
Fig. 18.18: Large random $\lambda$-term, viewed as a mini-cosmos, evolving non-deterministically by local changes due to $\beta$-steps; their patterns are the red configurations. In the final result the place and nature of the normalized parts of the structure, as well as the singularities formed by the unsolvable terms, the black holes, is 'predestined', independent of the actual evolution path to the normal form, an infinite $\lambda\beta\Omega$-term.
S. Abramsky and C.-H. L. Ong.
Full Abstraction in the Lazy Lambda Calculus.

Peter Aczel.
A general Church-Rosser theorem.

and

Highlights in Infinitary Rewriting and Lambda Calculus.

Syntactic Definitions of Undefined: On Defining the Undefined.

Jürgen Avenhaus.
*Reduktionssysteme - Rechnen und Schliessen in gleichungsdefinierten Strukturen*.

F. Baader and T. Nipkow.
Term Rewriting and All That.

H. P. Barendregt.
The Type Free Lambda Calculus.

H. P. Barendregt.
The Lambda Calculus, its Syntax and Semantics, volume 103 of Studies in Logic and The Foundations of Mathematics.

Degrees, reductions and representability in the lambda calculus.
Technical report, Department of Mathematics, Utrecht University, 1976.

Needed Reduction and Spine Strategies for the Lambda Calculus.

H. P. Barendregt and J. W. Klop.
Applications of Infinitary Lambda Calculus.
*Lambda Calculus with Types.*  

A. Berarducci.  
Infinite $\lambda$-Calculus and Non-Sensible Models.  

A. Berarducci and M. Dezani-Ciancaglini.  
Infinite $\lambda$-calculus and types.  

A. Berarducci and B. Intrigila.  
Church–Rosser $\lambda$-theories, Infinite $\lambda$-calculus and Consistency Problems.  

I. Bethke.  
Lamda Calculus.  
Chapter 10 in [97].

I. Bethke, J. W. Klop, and de Vrijer, R.C.
Descendants and Origins in Term Rewriting.

M. Bezem, J. W. Klop, and V. van Oostrom.
Advanced ARS Theory.

C. Böhm, editor.

F. Cardone and J.R. Hindley.
History of lambda-calculus and combinatory logic.

A. Church and J.B. Rosser.
Some Properties of Conversion.

Th. Coquand.
Infinite Objects in Type Theory.
R. C. de Vrijer.

*Surjective Pairing and Strong Normalization: Two Themes in Lambda Calculus.*

N. Dershowitz and J.-P. Jouannaud.

Rewrite systems.

N. Dershowitz, S. Kaplan, and D. A. Plaisted.

Rewrite, Rewrite, Rewrite, Rewrite, Rewrite, Rewrite, . . . .

M. Dezani-Ciancaglini, P. Severi, and F.-J. de Vries.

Infinitary Lambda Calculus and Discrimination of Berarducci Trees.

R. Di Cosmo and D. Kesner.

Simulating expansions without expansions.
J. Endrullis and R. de Vrijer.
Reduction Under Substitution.

Levels of Undecidability in Rewriting.

Productivity of Stream Definitions.

Proving Infinitary Normalization.

Unique Normal Forms in Infinitary Weakly Orthogonal Rewriting.

J. Endrullis and D. Hendriks.
Lazy Productivity via Termination.

Modular Construction of Fixed Point Combinators and Clocked Böhm Trees.

Clocked lambda calculus.
2013.
To appear.

J. Endrullis and A. Polonsky.
A Characterization of Reduction 1-Cycles in Infinitary Lambda Calculus.
In *TYPES*, 2012.
Jörg Endrullis, Clemens Grabmayer, Jan Willem Klop, and Vincent van Oostrom.
On equal -terms.

Jörg Endrullis, Helle Hvid Hansen, Dimitri Hendriks, Andrew Polonsky, and Alexandra Silva.
A coinductive treatment of infinitary rewriting.

Enno Folkerts.
Invertibility in lambda-eta.

C. Hankin.

J. R. Hindley and P. Seldin.
*Lambda-Calculus and Combinators.*

S. Kahrs.
Infinitary Rewriting: Foundations Revisited.

J. R. Kennaway.
On transfinite abstract reduction systems.

J. R. Kennaway and F.-J. de Vries.
Infinitary Rewriting, chapter 12.
in [97].

An Infinitary Church–Rosser Property for Non-Collapsing Orthogonal Term Rewriting Systems.

From Finite to Infinite Lambda Calculi.


J. Ketema.
Comparing Böhm-Like Trees.

J. Ketema.
Counterexamples in Infinitary Rewriting with Non-fully-extended Rules.

J. Ketema.
Reinterpreting Compression in Infinitary Rewriting.

J. Ketema and J. G. Simonsen.
Infinitary Combinatory Reduction Systems.
J. Ketema and J. G. Simonsen.

J. Ketema and J. G. Simonsen.

J. Ketema and J. G. Simonsen.

J. W. Klop.

J. W. Klop.
Term Rewriting Systems.
J. W. Klop.
New Fixed Point Combinators from Old.

J. W. Klop and R. C. de Vrijer.
Extended term rewriting systems.

J. W. Klop and R. C. de Vrijer.
Infinitary Normalization.

J. W. Klop, V. van Oostrom, and F. van Raamsdonk.
Combinatory reduction systems: introduction and survey.

J. W. Klop, V. van Oostrom, and F. van Raamsdonk.

Jan Willem Klop and Roel C. de Vrijer.
Unique normal forms for lambda calculus with surjective pairing. 

A. Kurz, D. Petrisan, P. Severi, and F.-J. de Vries.
An Alpha-Corecursion Principle for the Infinitary Lambda Calculus. 

B. Lercher.
Lambda-Calculus Terms That Reduce To Themselves. 

J.-J. Lévy.
An Algebraic Interpretation of the $\lambda\beta K$-Calculus, and an Application of a Labelled $\lambda$-Calculus. 

S. Lucas.
Transfinite Rewriting Semantics for Term Rewriting Systems.
R. Mayr and T. Nipkow.
Higher-order rewrite systems and their confluence.

A. Middeldorp.
Call by Need Computations to Root-Stable Form.

A. Middeldorp, V. van Oostrom, F. van Raamsdonk, and R. C. de Vrijer, editors.

R. Nakajima.
Infinite normal forms for the λ-calculus.

E. Ohlebusch.
*Advanced Topics in Term Rewriting*. 
C.-H. L. Ong.

*The Lazy Lambda Calculus: an Investigation into the Foundations of Functional Programming.*

V. van Oostrom.

*Confluence for Abstract and Higher-Order Rewriting.*

V. van Oostrom.

Higher-order families.

V. van Oostrom.

Normalisation in Weakly Orthogonal Rewriting.

S. Ronchi Della Rocca.

*The Parametric Lambda Calculus: A Metamodelfor Computation.*
A. Salibra.
Topological incompleteness and order incompleteness of the lambda calculus.

D. Sangiorgi.
The lazy lambda calculus in a concurrency scenario.

D. S. Scott.
Some Philosophical Issues Concerning Theories of Combinators.

P. Severi and F.-J. de Vries.
A Lambda Calculus for $D_\infty$.

P. Severi and F.-J. de Vries.
An Extensional Böhm Model.
P. Severi and F.-J. de Vries. 
Continuity and Discontinuity in Lambda Calculus. 

P. Severi and F.-J. de Vries. 
Order Structures for Böhm-like Models. 

P. Severi and F.-J. de Vries. 
Decomposing the Lattice of Meaningless Sets in the Infinitary Lambda Calculus. 

P. Severi and F.-J. de Vries. 
Weakening the Axiom of Overlap in Infinitary Lambda Calculus. 

P. Severi and F.-J. de Vries. 
Meaningless Sets in Infinitary Combinatory Logic.
J. G. Simonsen.
On Confluence and Residuals in Cauchy Convergent Transfinite Rewriting.

J. G. Simonsen.
Weak Convergence and Uniform Normalization in Infinitary Rewriting.

R. Smullyan.
*To Mock a Mockingbird, and Other Logic Puzzles.*

R. Statman.
The Word problem for Smullyan’s Lark Combinator is Decidable.

Richard Statman.
Every countable poset is embeddable in the poset of unsolvable terms.


**Terese.**

*Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science.*

**V. van Oostrom and F. van Raamsdonk.**

Weak orthogonality implies confluence: the higher-order case.


**F. van Raamsdonk.**

Higher-order rewriting.

In *Term Rewriting Systems*, Chapter 11, pages 588–667.

**F.-J. de Vries.**

Böhm Trees, Bisimulations and Observations in Lambda Calculus.


**C. P. Wadsworth.**
The relation between computational and denotational properties for Scott’s $D_{\infty}$-models of the lambda-calculus.


C. P Wadsworth.
Approximate reduction and lambda calculus models.


J. Waldmann.
The Combinator S.


H. Zantema.
Normalization of Infinite Terms.

In *Proc. 19th Int. Conf. on Rewriting Techniques and Applications (RTA 2008)*, number 5117 in Lecture Notes in Computer Science, pages 441–455, 2008.

H. Zantema.
Well-Definedness of Streams by Transformation and Termination.

*Logical Methods in Computer Science, 6(3), 2010.*

H. Zantema and M. Raffelsieper.
Proving Productivity in Infinite Data Structures.