A Short Mechanized Proof of the Church-Rosser Theorem by the Z-property for the $\lambda\beta$-calculus in Nominal Isabelle*

Julian Nagele, Vincent van Oostrom, and Christian Sternagel

University of Innsbruck, Austria
{julian.nagele,vincent.van-oostrom,christian.sternagel}@uibk.ac.at

Abstract

We present a short proof of the Church-Rosser property for the lambda-calculus enjoying two distinguishing features: firstly, it employs the Z-property, resulting in a short and elegant proof; and secondly, it is formalized in the nominal higher-order logic available for the proof assistant Isabelle/HOL.

1 Introduction

Dehornoy proved confluence for the rule of self-distributivity $xyz \to xz(yz)$[1] by means of a novel method [3], the idea being to give a map • that is *monotonic* with respect to $\to^*$ and that yields for each object an *upper bound* on all objects reachable from it in a single step. Later, this method was extracted and dubbed the Z-property [4], and applied to prove confluence of various rewrite systems, in particular the $\lambda\beta$-calculus.

Here we present our Isabelle/HOL [8] formalization of part of the above mentioned work,[2] in particular that the $\lambda\beta$-calculus is confluent since it enjoys the Z-property and that the latter property is equivalent to an abstract version of Takahashi's confluence method [10]. We achieve a rigorous treatment of terms modulo α-equivalence by employing Nominal Isabelle [12], a nominal higher-order logic based on Isabelle/HOL. Our formalization is available from the *archive of formal proofs* [5]. Below, Isabelle code-snippets are in blue and hyperlinked.

2 Nominal λ-terms

In our formalization λ-terms are represented by the following nominal data type, where the annotation "**binds** x **in** t" indicates that the equality of such abstraction terms is up to renaming of x in t:

```
nominal_datatype term =
  Var name
| App term term
| Abs x::name t::term binds x in t
```

For the sake of readability we will use standard notation, i.e., x instead of *Var* x, $s\ t$ instead of *App* $s\ t$, and $\lambda x.\,t$ instead of *Abs* $x\ t$, in the remainder. When defining (recursive) functions on λ-terms, we may have to take care of so-called *freshness constraints*. A freshness constraint is written $x \sharp t$ and states that x does not occur in t, or equivalently, x is fresh for t.

Definition 1. Capture-avoiding substitution is defined recursively by the following equations:

*This work was partially supported by FWF (Austrian Science Fund) projects P27502 and P27528.

[1] Confluence of this single-rule term rewrite system is non-trivial: presently no tool can prove it automatically.

[2] The formalization follows the pen-and-paper proof *exactly*, except for one mistake in Lemma 9 (Rhs).

$$y\,[x := s] = (\textbf{if}\ x = y\ \textbf{then}\ s\ \textbf{else}\ y)$$
$$(t\ u)\,[x := s] = t\,[x := s]\ u\,[x := s]$$
$$y \sharp (x,\ s) \Longrightarrow (\lambda y.\ t)\,[x := s] = \lambda y.\ t\,[x := s]$$

Due to the constraint, the final equation is only applicable when y is fresh for x and s.

In principle it is always possible to rename variables in terms (or any finitely supported structure) apart from a given finite collection of variables. In order to relieve the user of doing so by hand, Nominal Isabelle [12] provides infrastructure for defining nominal functions, giving rise to strong induction principles that take care of appropriate renaming. (However, nominal functions do not come for free: after stating the defining equations, we are faced with proof obligations that ensure pattern-completeness, termination, equivariance, and well-definedness. With the help of some home-brewed Eisbach [6] methods we were able to handle those obligations automatically.) We first consider the Substitution Lemma, cf. [2, Lemma 2.1.16].

Lemma 2. $x \sharp (y,\ u) \Longrightarrow t\,[x := s]\,[y := u] = t\,[y := u]\,[x := s\,[y := u]]$

Proof. In principle the proof proceeds by induction on t. However, for the case of λ-abstractions we additionally want the bound variable to be fresh for s, u, x, and y. With Nominal Isabelle it is enough to indicate that the variables of those terms should be avoided in order to obtain appropriately renamed bound variables. We will not mention this fact again in future proofs.

- In the base case $t = z$ for some variable z. If $z = x$ then $t\,[x := s]\,[y := u] = s\,[y := u]$ and $t\,[y := u]\,[x := s\,[y := u]] = s\,[y := u]$, since then $z \neq y$ and thus $z\,[y := u] = z$. Otherwise $z \neq x$. Now if $z = y$, then $t\,[x := s]\,[y := u] = u$ and $t\,[y := u]\,[x := s\,[y := u]] = u$, since $x \sharp u$. If $z \neq y$ then both ends of the equation reduce to z and we are done.
- In case of an application, we conclude by definition and twice the IH.
- Now for the interesting case. Let $t = \lambda z.\ v$ such that $z \sharp (s, u, x, y)$. Then

$$\begin{aligned}(\lambda z.\ v)\,[x := s]\,[y := u] &= \lambda z.\ v\,[x := s]\,[y := u] && \text{since } z \sharp (s, u, x, y)\\ &= \lambda z.\ v\,[y := u]\,[x := s\,[y := u]] && \text{by IH}\\ &= (\lambda z.\ v)\,[y := u]\,[x := s\,[y := u]] && \text{since } z \sharp (s\,[y := u], u, x, y)\end{aligned}$$

where in the last step $z \sharp s\,[y := u]$ follows from $z \sharp (s, u, y)$ by induction on s. □

Definition 3. We define β-reduction inductively by the *compatible* closure [2, Definition 3.1.4] of the β-rule (in its nominal version):

$$x \sharp t \Longrightarrow (\lambda x.\ s)\ t \to_\beta s\,[x := t]$$

The freshness constraint on the β-rule is needed to obtain an induction principle strong enough with respect to avoiding capture of bound variables. The following standard "congruence properties" (cf. [2, Lemma 3.1.6 and Proposition 3.1.16]) will be used freely in the remainder:

$$s \to_\beta^* t \Longrightarrow u \to_\beta^* v \Longrightarrow s\ u \to_\beta^* t\ v$$
$$s \to_\beta^* t \Longrightarrow \lambda x.\ s \to_\beta^* \lambda x.\ t$$
$$s \to_\beta^* s' \Longrightarrow t \to_\beta^* t' \Longrightarrow t\,[x := s] \to_\beta^* t'\,[x := s']$$

They are proven along the lines of their textbook proofs, the first two by induction on the length and the last one by (nominal) induction on t followed by a nested (nominal) induction on the definition of β-steps, using the Substitution Lemma. Furthermore we will make use of the easily proven fact that β-reduction is *coherent* with abstraction:

$$\lambda x.\ s \to_\beta^* t \Longrightarrow \exists\, u.\ t = \lambda x.\ u \wedge s \to_\beta^* u$$

3 Z

We present the Z-property for abstract rewriting, show that it implies confluence, and then instantiate it for the case of (nominal) λ-terms modulo α equipped with β-reduction.

Definition 4. A relation $\to$ on A has the Z-property if there is a map $\bullet : A \to A$ such that $a \to b \Longrightarrow b \to^* a^\bullet \wedge a^\bullet \to^* b^\bullet$.

If $\to$ has the Z-property then it indeed is monotonic, i.e., $a \to^* b$ implies $a^\bullet \to^* b^\bullet$, which is straightforward to show by induction on the length of the former.

Lemma 5. *A relation that has the Z-property is confluent.*

Proof. We show semi-confluence [1]. So assume $a \to^* c$ and $a \to d$. We show $d \downarrow c$ by case analysis on the reduction from a to c. If it is empty there is nothing to show. Otherwise there is a b with $a \to^* b$ and $b \to c$. Then by monotonicity we have $a^\bullet \to^* b^\bullet$. From $a \to d$ we have $d \to^* a^\bullet$ using the Z-property, so in total $d \to^* b^\bullet$. Since by applying the Z-property to $b \to c$ we also get $c \to^* b^\bullet$ we have $d \downarrow c$ as desired. □

There are two natural choices for functions on λ-terms that yield the Z-property for $\to_\beta$, namely the full-development function and the full-superdevelopment function. The former maps a term to the result of contracting all residuals of redexes in it [2, Definition 13.2.7] and the latter also contracts the upward-created redexes, cf. [9, Section 2.7]. While Dehornoy and van Oostrom developed both proofs [4], here we opt for the latter, which requires less case analysis.

Definition 6. We first define a variant of *App* with built-in β-reduction at the root:

$$
\begin{aligned}
& x \sharp u \Longrightarrow (\lambda x.\, s') \cdot_\beta u = s'[x := u] \\
& x \cdot_\beta u = x\ u \\
& (s\ t) \cdot_\beta u = s\ t\ u
\end{aligned}
$$

An easy case analysis on the first argument shows that this function satisfies the congruence-like property $s \to_\beta^* s' \Longrightarrow t \to_\beta^* t' \Longrightarrow s \cdot_\beta t \to_\beta^* s' \cdot_\beta t'$.

Definition 7. The full-superdevelopment function $\bullet$ on λ-terms is defined as follows:

$$
\begin{aligned}
& x^\bullet = x \\
& (\lambda x.\, t)^\bullet = \lambda x.\, t^\bullet \\
& (s\ t)^\bullet = s^\bullet \cdot_\beta t^\bullet
\end{aligned}
$$

Below, we freely use the fact that $s^\bullet\ t^\bullet \to_\beta^= (s\ t)^\bullet$, which is shown by considering whether or not $s^\bullet$ is an abstraction. The structure of the proof that the $\lambda\beta$-calculus has the Z-property follows that for self-distributivity in that it build on the Self- and Rhs-properties. The former expresses that each term *self*-expands to its full-superdevelopment, and the latter that applying $\bullet$ to the *right-hand side* of the β-rule, i.e., to the result of a substitution, "does more" than applying the map to its components first. Each is proven by structural induction.

Lemma 8 (Self)**.** *For all terms t we have $t \to_\beta^* t^\bullet$.*

Proof. By induction on t using an additional case analysis on $t_1^\bullet$ in the case that $t = t_1\ t_2$. □

Lemma 9 (Rhs)**.** *For all terms t, s and all variables x we have $t^\bullet\, [x := s^\bullet] \to_\beta^* t[x := s]^\bullet$.*

Proof. By induction on t. The cases $t = x$ and $t = \lambda y.\, t'$ are straightforward. If $t = t_1\ t_2$ we continue by case analysis on $t_1^\bullet$.

If $t_1^\bullet = \lambda y.\, u$ then $\lambda y.\, u\,[x := s^\bullet] = t_1^\bullet\,[x := s^\bullet] \rightarrow_\beta^* t_1\,[x := s]^\bullet$ by induction hypothesis. Then, using coherence of β-reduction with abstraction, we can obtain a term v with $t_1\,[x := s]^\bullet = \lambda y.\, v$ and $u\,[x := s^\bullet] \rightarrow_\beta^* v$. We then have $(t_1\ t_2)^\bullet\,[x := s^\bullet] = u\,[y := t_2^\bullet]\,[x := s^\bullet] = u\,[x := s^\bullet]\,[y := t_2^\bullet\,[x := s^\bullet]]$, using the substitution lemma in the last step. Together with $u\,[x := s^\bullet] \rightarrow_\beta^* v$ and the induction hypothesis for t_2 this yields $(t_1\ t_2)^\bullet\,[x := s^\bullet] \rightarrow_\beta^* v\,[y := t_2\,[x := s]^\bullet]$. Since we also have $(t_1\ t_2)\,[x := s]^\bullet = (t_1\,[x := s]\ t_2\,[x := s])^\bullet = v\,[y := t_2\,[x := s]^\bullet]$ we can conclude this case.

If $t_1^\bullet$ is not an abstraction. then from the induction hypothesis we have $(t_1\ t_2)^\bullet\,[x := s^\bullet] = t_1^\bullet\,[x := s^\bullet]\ t_2^\bullet\,[x := s^\bullet] \rightarrow_\beta^* t_1\,[x := s]^\bullet\ t_2\,[x := s]^\bullet \rightarrow_\beta^= (t_1\ t_2)\,[x := s]^\bullet$. □

Lemma 10 (Z)**.** *The full-superdevelopment function* $\bullet$ *yields the Z-property for* $\rightarrow_\beta$*, i.e., we have* $s \rightarrow_\beta t \Longrightarrow t \rightarrow_\beta^* s^\bullet \wedge s^\bullet \rightarrow_\beta^* t^\bullet$ *for all terms* s *and* t*.*

Proof. Assume $s \rightarrow_\beta t$. We continue by induction on the derivation of $\rightarrow_\beta$.

If $s \rightarrow_\beta t$ is a root step then $s = (\lambda x.\, s')\ t'$ and $t = s'\,[x := t']$ for some s' and t'. Then $s^\bullet = s'^\bullet\,[x := t'^\bullet]$ and thus $t \rightarrow_\beta^* s^\bullet$ using Lemma 8 twice, so $s^\bullet \rightarrow_\beta^* t^\bullet$ by Lemma 9.

The case where the step happens below an abstraction follows from the induction hypothesis.

If the step happens in the left argument of an application then $s = s'\ u$ and $t = t'\ u$. From the induction hypothesis and Lemma 8 we have $t'\ u \rightarrow_\beta^* s'^\bullet\ u^\bullet \rightarrow_\beta^= (s\ u)^\bullet$. That also $(s'\ u)^\bullet \rightarrow_\beta^* (t'\ u)^\bullet$ follows directly from the induction hypothesis. The case where the step happens in the right argument of an application is symmetric. □

4 Perspective

This note originated from the bold and vague claim of Dehornoy and van Oostrom [4] that the confluence proof for the $\lambda\beta$-calculus by establishing the Z-property for the full-superdevelopment map, is *the shortest*. We present a brief qualitative and quantitative analysis of this claim.

Three major methods in the literature for showing confluence of the $\lambda\beta$-calculus are:

$$\text{complete developments} \models \diamond \Longrightarrow \text{complete, full-developments} \models \angle \Longleftarrow \text{full-developments} \models \mathrm{Z}$$

From left to right, that complete developments have the diamond ($\diamond$) property is due to Tait and Martin–Löf [2, Section 3.2], that complete developments have the angle ($\angle$) property with respect to the full-development function is due to Takahashi [10] (cf. [11, Proposition 1.1.11]), and that full-developments have the Z-property is due to [4]. From the fact that the second method needs the concepts of both the others, it stands to reason that its formalization is not the shortest, as confirmed by a formalization of Nipkow [7] and our quantitative analysis below.

Our proof varies on the above picture along yet another dimension, replacing developments (due to Church and Rosser, cf. [2, Definition 11.2.11]) by superdevelopments (due to Aczel, cf. [9, Section 2.7]). Where full-developments give a “tight” upper bound on the single-step reducts of a given term, full-superdevelopments do not, and one may hope for a simplification of the analysis because of it. This is confirmed by our quantitative analysis below. One may vary along this dimension as well: *any* map $\bullet$ having the Z-property suffices as we show now.

Definition 11. A relation $\rightarrow$ on A has the *angle* property for a map $\bullet$ from A to A, and relation $\mathrel{\circ\!\!\!\twoheadrightarrow}$ on A, if $\rightarrow \subseteq \mathrel{\circ\!\!\!\twoheadrightarrow} \subseteq \rightarrow^*$ and $a \mathrel{\circ\!\!\!\twoheadrightarrow} b$ implies $b \mathrel{\circ\!\!\!\twoheadrightarrow} a^\bullet$.

Lemma 12. *A relation* $\rightarrow$ *has the Z-property for map* $\bullet$ *if and only if it has the angle property for map* $\bullet$ *and some relation* $\mathrel{\circ\!\!\!\twoheadrightarrow}$*.*

Proof. First assume that $\mathrel{-\!\!\!\circ\!\!\!\!\rightarrow}$ has the angle property for map $\bullet$ and relation $\mathrel{-\!\!\!\circ\!\!\!\!\rightarrow}$. To show that $\to$ has Z assume $a \to b$. Then by assumption we also have $a \mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} b$ and hence $b \mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} a^\bullet$ and $a^\bullet \mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} b^\bullet$, by applying the angle property twice, which together with $\mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} \subseteq \to^*$ yields Z.

Now assume $\to$ has the Z-property. We define the $\bullet$-*development*[3] relation by $a \mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} b$ if $a \to^* b$ and $b \to^* a^\bullet$. Then $\to \subseteq \mathrel{-\!\!\!\circ\!\!\!\!\rightarrow} \subseteq \to^*$ follows from the definition of $\mathrel{-\!\!\!\circ\!\!\!\!\rightarrow}$ and the Z-property. The angle itself directly follows from the definition of $\mathrel{-\!\!\!\circ\!\!\!\!\rightarrow}$ and monotonicity of $\bullet$. □

We turn to the quantitative analysis of the claim of [4]. Formalizing confluence of the $\lambda\beta$-calculus has a long history for which we refer the reader to [7]. We compare our formalization in Isabelle to two other such, Nipkow's formalization in Isabelle/HOL [7] (as currently distributed with Isabelle) and Urban and Arnaud's formalization in Nominal Isabelle.[4] There are two major differences of the present proof to Nipkow's formalization. On the one hand Nipkow uses de Brujin indices to represent λ-terms. This considerably increases the size of the formal theories – almost 200 lines of the roughly 550 line development are devoted to setting up terms and the required manipulations on indices. Our development is 300 lines (60 of which are used for our ad hoc Eisbach methods). The second difference is the actual technique used to show confluence: Nipkow proceeds by establishing the diamond property for complete developments. Urban and Arnaud proceed by establishing the angle property for multisteps with respect to the full-development function. This results in a 100 line increase compared to our formalization.

References

[1] F. Baader and T. Nipkow. *Term Rewriting and All That.* CUP, 1998.

[2] H. Barendregt. *The Lambda Calculus: Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, 2nd revised edition, 1984.

[3] P. Dehornoy. *Braids and Self-Distributivity*, volume 192 of *Progress in Mathematics*. Springer, 2000. doi:10.1007/978-3-0348-8442-6.

[4] P. Dehornoy and V. van Oostrom. Z, proving confluence by monotonic single-step upperbound functions. In *LMRC*, 2008. www.phil.uu.nl/~oostrom/publication/talk/lmrc060508.pdf.

[5] B. Felgenhauer, J. Nagele, V. van Oostrom, and C. Sternagel. The Z property. *AFP*, June 2016. https://www.isa-afp.org/entries/Rewriting_Z.shtml, Formal proof development.

[6] D. Matichuk, T. Murray, and M. Wenzel. Eisbach: A proof method language for Isabelle. *J. Autom. Reasoning*, 56(3):261–282, 2016. doi:10.1007/s10817-015-9360-2.

[7] T. Nipkow. More Church-Rosser proofs. *J. Autom. Reasoning*, 26(1):51–66, 2001. doi:10.1023/A:1006496715975.

[8] T. Nipkow, L. C. Paulson, and M. Wenzel. *Isabelle/HOL - A Proof Assistant for Higher-Order Logic*, volume 2283 of *LNCS*. Springer, 2002. doi:10.1007/3-540-45949-9.

[9] F. van Raamsdonk. *Confluence and Normalisation for Higher-Order Rewriting.* PhD thesis, Vrije Universiteit Amsterdam, 1996.

[10] M. Takahashi. Parallel reductions in λ-calculus. *Inform. Comput.*, 118(1):120–127, 1995. doi:10.1006/inco.1995.1057.

[11] Terese. *Term Rewriting Systems*, volume 55 of *CTTCS*. CUP, 2003.

[12] C. Urban and C. Kaliszyk. General bindings and alpha-equivalence in Nominal Isabelle. *LMCS*, 8(2):14:1–14:35, 2012. doi:10.2168/LMCS-8(2:14)2012.

[3] For the full-development map $\bullet$ such *syntax-free* $\bullet$-developments may differ from the usual ones, e.g. for $(\lambda y.\, I)\ ((\lambda x.\, x\ x)\ I)$. We conjecture that on terminating, non-erasing and non-collapsing λ-terms they coincide.

[4] http://www.inf.kcl.ac.uk/staff/urbanc/cgi-bin/repos.cgi/nominal2/file/d79e936e30ea/Nominal/Ex/CR.thy