


PAPER

Z property for the shuffling calculus

Koji Nakazawa^{1*} , Ken-etsu Fujita² and Yuta Imagawa³

¹Nagoya University, Nagoya, Japan ²Gunma University, Maebashi, Japan and ³Mitsubishi UFJ Research and Consulting Co., Ltd., Tokyo, Japan

*Corresponding author. Email: knak@i.nagoya-u.ac.jp

(Received 24 February 2021; revised 15 November 2022; accepted 29 November 2022; first published online 10 January 2023)

Abstract

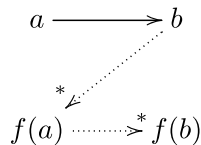
This paper gives a new proof of confluence for Carraro and Guerrieri's call-by-value lambda calculus λ_v^σ with permutation rules. We adapt the compositional Z theorem to λ_v^σ .

Keywords: Confluence; Z theorem; call-by-value lambda calculus

1. Introduction

Confluence is one of the most important properties of rewriting systems. It means that the result of computation does not depend on the order of the computation. In general, it is hard to prove confluence for higher-order rewriting such as (extensions of) the lambda calculus. There is a long history of how to give simple and elegant proofs for confluence of lambda calculi: Church and Rosser's proof with the notion of residuals of redexes, Tait and Martin-Löf's proofs with parallel reduction, and Takahashi's proof with complete developments. Hindley-Rosen's method with commutation of several kinds of reductions is widely applicable for the lambda calculi extended by some reduction rules.

The *Z theorem* by Dehornoy and van Oostrom (2008), van Oostrom (2021) is one of such elegant techniques, and it is widely applicable to confluence proofs in abstract settings. The Z theorem says that confluence of an abstract rewriting system follows from the existence of a mapping f satisfying the following Z property: any one-step reduction $a \rightarrow b$ implies $b \rightarrow^* f(a) \rightarrow^* f(b)$, where \rightarrow^* is the reflexive transitive closure of \rightarrow . It becomes clear why they call it the Z property when one draws the condition as the following diagram.



The mapping f that satisfies the Z property gives an abstract essence of developments without relying on the syntax of terms. If we define $a \xrightarrow{f} b$ by $a \rightarrow^* b \rightarrow^* f(a)$, then the Z property is equivalent to Takahashi's angle property: $a \xrightarrow{f} b$ implies $b \xrightarrow{f} f(a)$. The Z theorem has been applied to some variants of the lambda calculus in Dehornoy and van Oostrom (2008), Komori et al. (2013), Accattoli and Kesner (2012), Nakazawa and Nagai (2014).

The Z property is not only a tool to prove confluence but it is also useful for quantitative analyses of the Church-Rosser property as Fujita (2020) shows. The Z property is strictly stronger than confluence; that is, there are confluent rewrite systems for which no mapping satisfies the Z property. The following example is introduced by Dehornoy and van Oostrom (2008). Let A be the set of integers and $n \rightarrow m$ mean either $(m = n + 1)$ or $(n < 0 \text{ and } m = -n)$, and then, the abstract rewriting system (A, \rightarrow) is confluent but there is no mapping that satisfies the Z property. Hence, it is an interesting problem whether rewriting systems satisfy not only confluence but also the Z property.

Nakazawa and Fujita (2016) have proposed a variant of the Z theorem, called the *compositional Z theorem*. The compositional Z theorem enables us to apply the Z theorem by dividing rewriting systems into two or more subsystems. Compositional Z gives a simple proof of confluence of the lambda-mu calculus with permutation rules for direct sums. The original confluence proof by Ando (2003) for that system is more complex, and one cannot naively apply the original Z theorem because it is hard to directly define a mapping satisfying the Z property for both beta and permutation reductions. Moreover, Honda et al. (2021) adapted the compositional Z theorem to several variants of the lambda-mu calculus with simplification rules.

Carraro and Guerrieri (2014) introduced λ_v^σ , a call-by-value variant of the lambda calculus with strong reduction (reductions under lambda abstractions are allowed) for open terms. If we naively consider call-by-value computations for open terms, there exist terms that are neither a normal form nor a value such as xx , and then, the term $(\lambda y. \lambda z. zz)(xx)(\lambda z. zz)$ cannot be reduced while it morally contains the redex $(\lambda z. zz)(\lambda z. zz)$, which diverges. To avoid such unexpected stuckness of computation, they adopted some permutation rules. In Accattoli and Guerrieri (2016), this calculus is also called the *shuffling calculus* λ_{shuf} and further discussed as one of the call-by-value variants of the lambda calculus for open terms. As with the permutation rules for the direct sums, the permutation rules of λ_v^σ make the proof of confluence harder, and one cannot straightforwardly adapt the ordinary proof techniques resting on parallel reduction or complete developments. Carraro and Guerrieri proved confluence of λ_v^σ using Hindley-Rosen's method, in which they proved that β_v reduction and the permutation rules commute and that the permutation rules are strongly normalizing.

In this paper, we give another proof of confluence for the shuffling calculus λ_v^σ using the compositional Z theorem. This proof provides a case study of applications of the (compositional) Z theorem to extensions of the lambda calculi with some additional reduction rules. Our proof does not depend on any normalization property of the reduction relation, and it differs from the existing proof by Carraro and Guerrieri (2014), where the strong normalization of the permutation reduction is used. As with Nakazawa and Fujita (2016), we cannot naively apply the original Z theorem for λ_v^σ , and the problem can be avoided by the compositional Z. Although the outline of our proof follows the proof of Nakazawa and Fujita (2016), the main difference is that the mapping we consider here may leave redexes of permutation reductions since the calculus has two kinds of permutation rules σ_1 and σ_3 for the call-by-value evaluation context $[]M$ and $V[]$, respectively, for a term M and a value V . Hence, we cannot apply the simplified variant of the compositional Z (Corollary 2.4 in Nakazawa and Fujita 2016).

This paper is the extended version of the conference proceeding, Nakazawa et al. (2017).

2. The Shuffling Calculus λ_v^σ

In this section, we recall the call-by-value lambda calculus λ_v^σ introduced by Carraro and Guerrieri (2014).

We use the following notation throughout this paper: syntactic equality is denoted by $=$. For a binary relation R , we use R^+ and R^* for the transitive closure of R and the reflexive transitive closure of R , respectively.

Definition 1. Values and terms of λ_v^σ are given as follows.

$$\begin{aligned} V &::= x \mid \lambda x.M && \text{(values)} \\ M &::= V \mid MM && \text{(terms)} \end{aligned}$$

We define the size of M as the number of occurrences of the symbols in M . We define the free variables of M as usual, and the set of the free variables of M is denoted by $FV(M)$. A relation R on the terms is compatible if $M R M'$ implies $(\lambda x.M) R (\lambda x.M')$, $(MN) R (M'N)$, and $(NM) R (NM')$.

The reduction rules of λ_v^σ are given as follows.

$$\begin{aligned} (\lambda x.M)V &\rightarrow_{\beta_v} M[x := V] \\ (\lambda x.M)NL &\rightarrow_{\sigma_1} (\lambda x.ML)N && (x \notin FV(L)) \\ V((\lambda x.M)N) &\rightarrow_{\sigma_3} (\lambda x.VM)N && (x \notin FV(V)) \end{aligned}$$

Here, V is a value, M , N , and L are terms, and $M[x := V]$ is the usual capture-avoiding substitution. The relations \rightarrow_{β_v} , \rightarrow_{σ_1} , and \rightarrow_{σ_3} are the compatible closure of the reduction relation defined by the corresponding reduction rules, respectively. We define \rightarrow_σ as the union of \rightarrow_{σ_1} and \rightarrow_{σ_3} . We define \rightarrow as the union of \rightarrow_{β_v} and \rightarrow_σ .

This calculus is introduced in Carraro and Guerrieri (2014) to study operational characterization of solvability in call-by-value lambda calculi in particular for open terms. In the original call-by-value lambda calculus λ_v of Plotkin (1975), the term $M = (\lambda yx.xx)(zz)(\lambda x.xx)$ is stuck since zz is not a value, whereas semantically it is equivalent to $(\lambda x.xx)(\lambda x.xx)$ and hence unsolvable. In λ_v^σ , M is reduced to $(\lambda y.(\lambda x.xx)(\lambda x.xx))(zz)$ by the σ -rules, where the redex $(\lambda x.xx)(\lambda x.xx)$ is no longer blocked.

3. Compositional Z

We summarize the *Z theorem* of Dehornoy and van Oostrom (2008) and then extend it for compositional functions, called the *compositional Z* of Nakazawa and Fujita (2016).

Definition 2 ((Weak) Z property). Let (A, \rightarrow) be an abstract rewriting system and \rightarrow_x be another relation on A .

1. A mapping f satisfies the weak Z property for \rightarrow by \rightarrow_x if and only if $a \rightarrow b$ implies $b \rightarrow_x^* f(a) \rightarrow_x^* f(b)$ for any $a, b \in A$.
2. A mapping f satisfies the Z property for \rightarrow if and only if it satisfies the weak Z property for \rightarrow by \rightarrow .

When f satisfies the (weak) Z property, we also say that f is (weakly) Z.

Theorem 3 (Z theorem by Dehornoy and van Oostrom 2008). If there exists a mapping satisfying the Z property for an abstract rewriting system, then it is confluent.

We can often prove that usual complete developments have the Z property.

Theorem 4 (Compositional Z by Nakazawa and Fujita 2016). Let (A, \rightarrow) be an abstract rewriting system, and \rightarrow be $\rightarrow_1 \cup \rightarrow_2$. If there exist mappings $f_1, f_2 : A \rightarrow A$ such that

- (1) f_1 is Z for \rightarrow_1
- (2) $a \rightarrow_1 b$ implies $f_2(a) \rightarrow^* f_2(b)$

- (3) $a \rightarrow^* f_2(a)$ holds for any $a \in \text{Im}(f_1)$
- (4) $f_2 \circ f_1$ is weakly Z for \rightarrow_2 by \rightarrow_1 ,

then $f_2 \circ f_1$ is Z for (A, \rightarrow) , and hence (A, \rightarrow) is confluent.

One of the simplest applications of the compositional Z is for the $\beta\eta$ -reduction on the untyped lambda calculus (although it can be directly proved by the original Z theorem, as Komori et al. (2013) showed). If we let \rightarrow_1 be \rightarrow_η , \rightarrow_2 be \rightarrow_β , f_1 be the complete development of \rightarrow_η , and f_2 be the complete development of \rightarrow_β , then these satisfy the conditions of the compositional Z theorem.

In Nakazawa and Fujita (2016), the compositional Z is applied to the lambda calculus with permutation rules for direct sums. This calculus is a lambda calculus extended with two inclusion operators $\iota_1 M, \iota_2 M$ into direct sums and the case expression (case M with $\iota_1 x_1 \Rightarrow N_1 \mid \iota_2 x_2 \Rightarrow N_2$). For simplicity, this case expression is denoted as $M[x_1.N_1, x_2.N_2]$ in Nakazawa and Fujita (2016). The additional reduction rules are

$$\begin{aligned} (\iota_1 M)[x_1.N_1, x_2.N_2] &\rightarrow_\beta N_1[x_1 := M], \\ (\iota_2 M)[x_1.N_1, x_2.N_2] &\rightarrow_\beta N_2[x_2 := M], \\ E[M[x_1.N_1, x_2.N_2]] &\rightarrow_\pi M[x_1.E[N_1], x_2.E[N_2]], \end{aligned}$$

where E denotes singular destructor contexts defined as $E ::= []L \mid [][y_1.K_1, y_2.K_2] \mid \dots$. The last rule is called the permutation rule, and it makes confluence proofs much harder because of the following critical pair: for the term $P[x_1.N_1, x_2.N_2][y_1.L_1, y_2.L_2]K$, we have

$$\begin{aligned} \underline{P[x_1.N_1, x_2.N_2][y_1.L_1, y_2.L_2]K} &\rightarrow_\pi P[x_1.N_1[y_1.L_1, y_2.L_2], x_2.N_2[y_1.L_1, y_2.L_2]]K(=M_1) \\ \text{and } \underline{P[x_1.N_1, x_2.N_2][y_1.L_1, y_2.L_2]K} &\rightarrow_\pi P[x_1.N_1, x_2.N_2][y_1.L_1K, y_2.L_2K](=M_2), \end{aligned}$$

where the underlines indicate the redexes of the permutation reductions. Both M_1 and M_2 can be reduced to a common term $M_3 = P[x_1.N_1[y_1.L_1K, y_2.L_2K], x_2.N_2[y_1.L_1K, y_2.L_2K]]$, but in $M_1 \rightarrow^* M_3$ we have to reduce a redex which is created during the reduction sequence and does not occur in M_1 . Because of this, it is hard to apply straightforwardly either the ordinary parallel reduction or the original Z theorem. In Nakazawa and Fujita (2016), the compositional Z theorem is applied to this calculus by dividing the reduction relations into two parts, the β -reduction and the permutations, and defining two mappings, one for the β -reduction and the other for the permutations, satisfying the assumptions of the compositional Z theorem.

4. Confluence of λ_v^σ by the Compositional Z

In this section, we see that λ_v^σ contains similar critical pairs to those in the lambda calculus with direct sums and that the compositional Z solves the problem.

4.1 Difficulties in confluence proof for λ_v^σ

Consider the term $(\lambda x.N)((\lambda y.K)L)P$. Then, we have both

$$\begin{aligned} \underline{(\lambda x.N)((\lambda y.K)L)P} &\rightarrow_{\sigma_1} (\lambda x.NP)((\lambda y.K)L)(=M_1) \\ \text{and } \underline{(\lambda x.N)((\lambda y.K)L)P} &\rightarrow_{\sigma_3} (\lambda y.(\lambda x.N)K)LP(=M_2), \end{aligned}$$

where the underlines indicate the redexes. The terms M_1 and M_2 are reduced to a common term $M_3 = (\lambda y.(\lambda x.NP)K)L$. $M_1 \rightarrow M_3$ is one-step σ_3 reduction, whereas $M_2 \rightarrow^* M_3$ consists of two σ_1 steps:

$$M_2 = \underline{(\lambda y.(\lambda x.N)K)LP} \rightarrow_{\sigma_1} \underline{(\lambda y.(\lambda x.N)KP)L} \rightarrow_{\sigma_1} (\lambda y.(\lambda x.NP)K)L,$$

where the redex of the second σ_1 -step is created at the first step and it does not occur in M_2 . This means that we cannot naively extend the ordinary parallel reduction, by which M_2 is not reduced to M_3 in one step. To define a mapping satisfying the Z property, we have to reduce successive σ -reductions at once. For example, the above example shows that M must be mapped to M_3 or some of its reducts.

In Nakazawa and Fujita (2016), they considered the auxiliary mapping $M@N$ such as

$$(\lambda y.(\lambda x.N)K)L@P = (\lambda y.(\lambda x.N@P)K)L,$$

which yields the normal form for the permutation reduction. Here, there are two kinds of permutation rules σ_1 and σ_3 and we consider the following two auxiliary mappings.

Definition 5. We define the mappings $@_1$ and $@_3$ as follows.

$$\begin{aligned} (\lambda x.M)N@_1P &= (\lambda x.M@_1P)N & V@_3(\lambda x.M)N &= (\lambda x.V@_3M)N \\ M@_1P &= MP \quad (\text{otherwise}) & V@_3M &= VM \quad (\text{otherwise}) \end{aligned}$$

However, similarly to the case of the direct sums, the following naive mapping does not satisfy the Z property.

$$\begin{aligned} x^* &= x \\ (\lambda x.M)^* &= \lambda x.M^* \\ ((\lambda x.M)V)^* &= M^*[x := V^*] \\ (MN)^* &= M^*@_1N^* \quad (M \text{ not a value}) \\ (VN)^* &= V^*@_3N^* \quad (\text{otherwise}) \end{aligned}$$

For $P = (\lambda x.xy)(\lambda z.z)v$ and $Q = (\lambda x.xyv)(\lambda z.z)$, we have $P \rightarrow_{\sigma_1} Q$, but we also have $P^* = (\lambda z.z)y@_1v = (\lambda z.zv)y$ and $Q^* = (\lambda z.z)yv$, and hence, $P^* \not\rightarrow^* Q^*$. In Q^* , the substitution $(xyv)[x := \lambda z.z]$ creates the σ_1 -redex $(\lambda z.z)yv$, and it is not reduced.

4.2 Confluence proof of λ_v^σ by applying the compositional Z theorem

We solve this problem by the compositional Z theorem. To apply it, we divide the reductions of λ_v^σ into σ and β_v , and define the mappings $(\cdot)^S$ and $(\cdot)^B$.

Definition 6. We define the mappings $(\cdot)^S$ and $(\cdot)^B$ as follows.

$$\begin{aligned} x^S &= x & x^B &= x \\ (\lambda x.M)^S &= \lambda x.M^S & (\lambda x.M)^B &= \lambda x.M^B \\ (MN)^S &= M^S@_1N^S \quad (M \text{ not a value}) & ((\lambda x.M)V)^B &= M^B[x := V^B] \\ (VN)^S &= V^S@_3N^S & (MN)^B &= M^BN^B \quad (\text{otherwise}) \end{aligned}$$

In the following, we show that these mappings satisfy the conditions of the compositional Z theorem.

The outline of the proof almost follows the proof in Nakazawa and Fujita (2016). However, in contrast to that proof, the mapping $(\cdot)^S$ does not necessarily collapse all σ -steps; that is, there are terms M and N such that $M \rightarrow_\sigma N$ and $M^S \rightarrow_\sigma^+ N^S$. For example, we have

$$M = (\lambda x.x)(yz)((\lambda v.v)w) \rightarrow_{\sigma_1} (\lambda x.x((\lambda v.v)w))(yz) = N,$$

and

$$M^S = ((\lambda x.x)(yz))^S @_1 ((\lambda v.v)w)^S = (\lambda x.x((\lambda v.v)w))(yz)$$

$$N^S = (\lambda x.x((\lambda v.v)w))^S @_3 (yz)^S = (\lambda x.x@_3((\lambda v.v)w))(yz) = (\lambda x.(\lambda v.xv)w)(yz),$$

The σ_3 -redex $x((\lambda v.v)w)$ in M^S is created in the application of $(\cdot)^S$, and it is not reduced in M^S . Hence, we cannot apply the simplified variant of compositional Z (Corollary 2.4 in Nakazawa and Fujita 2016) for these mappings.

From here, we prove the conditions of Theorem 4, where we consider \rightarrow_σ as \rightarrow_1 , \rightarrow_{β_v} as \rightarrow_2 , $(\cdot)^S$ as f_1 , and $(\cdot)^B$ as f_2 .

First, we prove auxiliary lemmas concerning the properties of the mappings $@_1$, $@_3$, and $(\cdot)^S$.

Lemma 7. *For any value V and any terms M, N , we have the following.*

- (1) $VM \rightarrow_\sigma^* V@_3M$.
- (2) $MN \rightarrow_\sigma^* M@_1N$.

Proof. They are proved by induction on the size of M .

- (1) If $M = (\lambda x.L)K$, we have

$$\begin{aligned} V((\lambda x.L)K) &\rightarrow_{\sigma_3} (\lambda x.VL)K \\ &\rightarrow_\sigma^* (\lambda x.V@_3L)K && \text{(induction hypothesis)} \\ &= V@_3(\lambda x.L)K. \end{aligned}$$

Otherwise, we have $VM = V@_3M$.

- (2) It is proved similarly. □

Lemma 8. *For any terms M and N , we have the following.*

- (1) $M^S@_1N^S \rightarrow_\sigma^* (MN)^S$.
- (2) $M^SN^S \rightarrow_\sigma^* (MN)^S$.

Proof. (1) If M is not a value, we have $M^S@_1N^S = (MN)^S$. If M is a value, we have $M^S@_1N^S = M^SN^S \rightarrow_\sigma^* M^S@_3N^S = (MN)^S$ by Lemma 7.

- (2) By Lemma 7 and (1), we have $M^SN^S \rightarrow_\sigma^* M^S@_1N^S \rightarrow_\sigma^* (MN)^S$. □

Lemma 9. *For any value V and any terms M, N, K , we have the following.*

- (1) $((\lambda x.M)@_3N)@_1K = (\lambda x.M@_1K)@_3N$.
- (2) $V@_3((\lambda x.M)@_3N) = (\lambda x.V@_3M)@_3N$.

Proof. They are proved by induction on the size of N .

- (1) If $N = (\lambda y.P)Q$, we have

$$\begin{aligned} ((\lambda x.M)@_3((\lambda y.P)Q))@_1K &= (\lambda y.((\lambda x.M)@_3P)@_1K)Q && \text{(def. of } @_1 \text{ and } @_3) \\ &= (\lambda y.(\lambda x.M@_1K)@_3P)Q && \text{(induction hypothesis)} \\ &= (\lambda x.M@_1K)@_3((\lambda y.P)Q) && \text{(def. of } @_3). \end{aligned}$$

Otherwise, by the definitions of $@_1$ and $@_3$, we have $((\lambda x.M)@_3N)@_1K = ((\lambda x.M)N)@_1K = (\lambda x.M@_1K)N = (\lambda x.M@_1K)@_3N$.

(2) It is proved similarly. □

We prove the following lemma. The second property means that the condition (3) of Theorem 4 holds.

Lemma 10. *For any term M , we have the following.*

- (1) $M \rightarrow_{\sigma}^* M^S$.
- (2) $M \rightarrow_{\beta_v}^* M^B$.

Proof. They are proved by induction on the size of M .

- (1) In the case of $M = NK$, we have $NK \rightarrow_{\sigma}^* N^S K^S$ by the induction hypothesis, and $N^S K^S \rightarrow_{\sigma}^* (NK)^S$ by Lemma 8. In the case of $M = x$, it is trivial. In the case of $M = \lambda x.N$, it is proved by the induction hypothesis.
- (2) In the case of $M = NK$, we consider the following cases. If $N = \lambda x.L$ and K is a value, we have

$$\begin{aligned} (\lambda x.L)V &\rightarrow_{\beta_v} L[x := V] \\ &\rightarrow_{\beta_v}^* L^B[x := V^B] && \text{(induction hypothesis)} \\ &= ((\lambda x.L)V)^B. \end{aligned}$$

Otherwise, we have $(NK)^B = N^B K^B$, and $NK \rightarrow_{\beta_v}^* N^B K^B$ holds by the induction hypothesis. In the case of $M = x$, it is trivial. In the case of $M = \lambda x.N$, it is proved by the induction hypothesis. □

The next goal is to prove that the condition (1) of Theorem 4 holds; that is, $(\cdot)^S$ is Z for \rightarrow_{σ} (Propositions 12 and 15).

The following lemma is used to prove the cofinality (Proposition 12): $M \rightarrow_{\sigma} M'$ implies $M' \rightarrow_{\sigma}^* M^S$.

Lemma 11. *For any value V and any terms M, N, K , we have the following.*

- (1) If $x \notin FV(K)$, we have $(\lambda x.MK)N \rightarrow_{\sigma}^* ((\lambda x.M)@_3N)@_1K$.
- (2) If $x \notin FV(V)$, we have $(\lambda x.VM)N \rightarrow_{\sigma}^* V@_3((\lambda x.M)@_3N)$.

Proof. They are proved by induction on the size of N .

- (1) If $N = (\lambda y.P)Q$, we have

$$\begin{aligned} (\lambda x.MK)((\lambda y.P)Q) &\rightarrow_{\sigma_3} (\lambda y.(\lambda x.MK)P)Q \\ &\rightarrow_{\sigma}^* (\lambda y.((\lambda x.M)@_3P)@_1K)Q && \text{(induction hypothesis)} \\ &= ((\lambda y.(\lambda x.M)@_3P)Q)@_1K && \text{(definition of } @_1) \\ &= ((\lambda x.M)@_3(\lambda y.P)Q)@_1K && \text{(definition of } @_3). \end{aligned}$$

Otherwise, we have

$$\begin{aligned} (\lambda x.MK)N &\rightarrow_{\sigma}^* (\lambda x.M@_1K)N && \text{(Lemma 7 (2))} \\ &= ((\lambda x.M)N)@_1K && \text{(definition of } @_1) \\ &= ((\lambda x.M)@_3N)@_1K && \text{(definition of } @_3). \end{aligned}$$

(2) If $N = (\lambda y.P)Q$, we have

$$\begin{aligned} (\lambda x.VM)((\lambda y.P)Q) &\rightarrow_{\sigma_3} (\lambda y.(\lambda x.VM)P)Q \\ &\rightarrow_{\sigma}^* (\lambda y.V@_3((\lambda x.M)@_3P)Q) && \text{(induction hypothesis)} \\ &= V@_3(\lambda y.((\lambda x.M)@_3P)Q) && \text{(definition of @}_3\text{)} \\ &= V@_3((\lambda x.M)@_3(\lambda y.P)Q) && \text{(definition of @}_3\text{)}. \end{aligned}$$

Otherwise, we have

$$\begin{aligned} (\lambda x.VM)N &\rightarrow_{\sigma}^* (\lambda x.V@_3M)N && \text{(Lemma 7 (1))} \\ &= V@_3((\lambda x.M)N) && \text{(definition of @}_3\text{)} \\ &= V@_3((\lambda x.M)@_3N) && \text{(definition of @}_3\text{)}. \end{aligned}$$

□

Proposition 12. For any terms M and M' , if $M \rightarrow_{\sigma} M'$ holds, then $M' \rightarrow_{\sigma}^* M^S$ holds.

Proof. By induction on a derivation of $M \rightarrow_{\sigma} M'$.

First, the base cases are proved as follows.

If $M = (\lambda x.P)QK$ and $M' = (\lambda x.PK)Q$, we have

$$\begin{aligned} (\lambda x.PK)Q &\rightarrow_{\sigma}^* (\lambda x.P^S K^S)Q^S && \text{(Lemma 10)} \\ &\rightarrow_{\sigma}^* ((\lambda x.P^S)@_3 Q^S)@_1 K^S && \text{(Lemma 11)} \\ &= ((\lambda x.P)QK)^S && \text{(definition of S)}. \end{aligned}$$

If $M = V((\lambda x.P)Q)$ and $M' = (\lambda x.VP)Q$, it is proved similarly by Lemmas 10 and 11.

Next, we show the induction cases.

If $M = \lambda x.N$, $M' = \lambda x.N'$, and $N \rightarrow_{\sigma} N'$, then, by the induction hypothesis, we have $N' \rightarrow_{\sigma}^* N^S$. Then, we have $\lambda x.N' \rightarrow_{\sigma}^* \lambda x.N^S = (\lambda x.N)^S$.

If $M = PQ$, $M' = P'Q'$, and either $P \rightarrow_{\sigma} P'$ and $Q = Q'$ or $Q \rightarrow_{\sigma} Q'$ and $P = P'$, then, by the induction hypothesis and Lemma 10, we have $P' \rightarrow_{\sigma}^* P^S$ and $Q' \rightarrow_{\sigma}^* Q^S$. Then, we have $P'Q' \rightarrow_{\sigma}^* P^S Q^S \rightarrow_{\sigma}^* (PQ)^S$ by Lemma 8. □

Therefore, we prove some lemmas to prove the monotonicity (Proposition 15): $M \rightarrow_{\sigma} M'$ implies $M^S \rightarrow_{\sigma}^* M'^S$.

Lemma 13. For any values V, W and any terms M, N , we have the following.

- (1) $(M@_1N)[x := V] \rightarrow_{\sigma}^* M[x := V]@_1N[x := V]$.
- (2) $(W@_3M)[x := V] \rightarrow_{\sigma}^* W[x := V]@_3M[x := V]$.

Proof. They are proved by induction on the size of M .

(1) If $M = (\lambda y.P)Q$, we have

$$\begin{aligned} (((\lambda y.P)Q)@_1N)[x := V] &= ((\lambda y.P@_1N)Q)[x := V] && \text{(definition of @}_1\text{)} \\ &= (\lambda y.(P@_1N)[x := V])Q[x := V] \\ &\rightarrow_{\sigma}^* (\lambda y.P[x := V]@_1N[x := V])Q[x := V] && \text{(induction hypothesis)} \\ &= ((\lambda y.P[x := V])Q[x := V])@_1N[x := V] && \text{(definition of @}_1\text{)} \\ &= ((\lambda y.P)Q)[x := V]@_1N[x := V]. \end{aligned}$$

Otherwise, we have $(M@_1N)[x := V] = M[x := V]N[x := V] \rightarrow_{\sigma}^* M[x := V]@_1N[x := V]$ by Lemma 7.

(2) It is proved similarly. □

We prove that the mappings $@_1$ and $@_3$ preserve the reductions \rightarrow_{σ} and \rightarrow_{β_v} in the following sense. This is the main lemma for the monotonicity and is also used to prove that the conditions (2) and (4) of Theorem 4 hold.

Lemma 14. *For any values V, V' and any terms M, M', N, N' , we have the following.*

- (1) *If $M \rightarrow_{\sigma} M'$, then $M@_1N \rightarrow_{\sigma}^* M'@_1N$ and $V@_3M \rightarrow_{\sigma}^* V@_3M'$ hold.*
- (2) *If $M \rightarrow_{\beta_v} M'$, then $M@_1N \rightarrow_{\sigma}^* M'@_1N$ and $V@_3M \rightarrow_{\sigma}^* V@_3M'$ hold.*
- (3) *For $\xi \in \{\beta_v, \sigma\}$, if $N \rightarrow_{\xi} N'$, then $M@_1N \rightarrow_{\xi} M@_1N'$ holds.*
- (4) *For $\xi \in \{\beta_v, \sigma\}$, if $V \rightarrow_{\xi} V'$, then $V@_3M \rightarrow_{\xi} V'@_3M$ holds.*

Proof. They are proved by induction on the size of M .

- (1) If $M = (\lambda x.P)Q$, it is not a σ -redex, so M' is $(\lambda x.P')Q'$ for some P' and Q' such that either $P \rightarrow_{\sigma} P'$ and $Q = Q'$ or $Q \rightarrow_{\sigma} Q'$ and $P = P'$. Then, we have

$$\begin{aligned} ((\lambda x.P)Q)@_1N &= ((\lambda x.P@_1N)Q) && \text{(definition of } @_1) \\ &\rightarrow_{\sigma}^* ((\lambda x.P'@_1N)Q') && \text{(inductive hypothesis)} \\ &= ((\lambda x.P')Q')@_1N && \text{(definition of } @_1). \end{aligned}$$

Otherwise, we have $M@_1N = MN \rightarrow_{\sigma} M'N \rightarrow_{\sigma}^* M'@_1N$ by Lemma 7.

The case $@_3$ is proved similarly.

- (2) If $M = (\lambda x.P)V$ and $M' = P[x := V]$, we have

$$\begin{aligned} ((\lambda x.P)V)@_1N &= ((\lambda x.P@_1N)V) && \text{(definition of } @_1) \\ &\rightarrow_{\beta_v} (P@_1N)[x := V] \\ &\rightarrow_{\sigma}^* P[x := V]@_1N && \text{(Lemma 13 and } x \notin FV(N)). \end{aligned}$$

If $M = (\lambda x.P)Q$, $M' = (\lambda x.P')Q'$, and either $P \rightarrow_{\beta_v} P'$ and $Q = Q'$ or $Q \rightarrow_{\beta_v} Q'$ and $P = P'$, we have

$$\begin{aligned} ((\lambda x.P)Q)@_1N &= ((\lambda x.P@_1N)Q) && \text{(definition of } @_1) \\ &\rightarrow^* ((\lambda x.P'@_1N)Q') && \text{(induction hypothesis)} \\ &= ((\lambda x.P')Q')@_1N && \text{(definition of } @_1). \end{aligned}$$

Otherwise, we have $M@_1N = MN \rightarrow_{\beta_v} M'N \rightarrow_{\sigma}^* M'@_1N$ by Lemma 7.

The case $@_3$ is proved similarly.

- (3) If $M = (\lambda x.P)Q$, we have

$$\begin{aligned} ((\lambda x.P)Q)@_1N &= (\lambda x.P@_1N)Q && \text{(definition of } @_1) \\ &\rightarrow_{\xi} (\lambda x.P@_1N')Q && \text{(inductive hypothesis)} \\ &= ((\lambda x.P)Q)@_1N' && \text{(definition of } @_1). \end{aligned}$$

Otherwise, we have $M@_1N = MN \rightarrow_{\xi} MN' = M@_1N'$.

- (4) It is proved similarly. □

Proposition 15. For any terms M and M' , if $M \rightarrow_{\sigma} M'$ holds, then $M^S \rightarrow_{\sigma}^* M'^S$ holds.

Proof. By induction on a derivation of $M \rightarrow_{\sigma} M'$.

First, the base cases are proved as follows.

If $M = (\lambda x.P)QK$ and $M' = (\lambda x.PK)Q$, we have

$$\begin{aligned} ((\lambda x.P)QK)^S &= ((\lambda x.P^S)@_3 Q^S)@_1 K^S \\ &= (\lambda x.P^S@_1 K^S)@_3 Q^S && \text{(Lemma 9 (1))} \\ &\rightarrow_{\sigma}^* (\lambda x.(PK)^S)@_3 Q^S && \text{(Lemmas 8 (1) and 14 (4))} \\ &= ((\lambda x.PK)Q)^S. \end{aligned}$$

If $M = V((\lambda x.P)Q)$ and $M' = (\lambda x.VP)Q$, we have

$$\begin{aligned} (V((\lambda x.P)Q))^S &= V^S@_3((\lambda x.P^S)@_3 Q^S) \\ &= (\lambda x.V^S@_3 P^S)@_3 Q^S && \text{(Lemma 9 (2))} \\ &= ((\lambda x.VP)Q)^S. \end{aligned}$$

Next, the induction cases are proved as follows.

If $M = \lambda x.N$, $M' = \lambda x.N'$, and $N \rightarrow_{\sigma} N'$, by the induction hypothesis, we have $N^S \rightarrow_{\sigma}^* N'^S$. Then, we have $(\lambda x.N)^S = \lambda x.N^S \rightarrow_{\sigma}^* \lambda x.N'^S = (\lambda x.N')^S$.

If $M = PQ$, $M' = P'Q'$, and either $P \rightarrow_{\sigma} P'$, $Q = Q'$ or $Q \rightarrow_{\sigma} Q'$, $P = P'$, by the induction hypothesis, we have $P^S \rightarrow_{\sigma}^* P'^S$ and $Q^S \rightarrow_{\sigma}^* Q'^S$. If P is not a value, we have

$$\begin{aligned} (PQ)^S &= P^S@_1 Q^S \\ &\rightarrow_{\sigma}^* P'^S@_1 Q'^S && \text{(Lemma 14 (1) and (3))} \\ &\rightarrow_{\sigma}^* (P'Q')^S && \text{(Lemma 8)}. \end{aligned}$$

If P is a value, we have

$$\begin{aligned} (PQ)^S &= P^S@_3 Q^S \\ &\rightarrow_{\sigma}^* P'^S@_3 Q'^S && \text{(Lemma 14 (1) and (4))} \\ &= (P'Q')^S && (P' \text{ is a value}). \end{aligned}$$

□

We prove the condition (2) of Theorem 4 (Proposition 18).

Lemma 16. For any value V and any term M , we have the following.

- (1) $M^S[x := V^S] \rightarrow_{\sigma}^* M[x := V]^S$.
- (2) $M^B[x := V^B] \rightarrow_{\beta_V}^* M[x := V]^B$.

Proof. (1) By induction on the size of M . If $M = PQ$ and P is not a value, we have

$$\begin{aligned} (PQ)^S[x := V^S] &= (P^S@_1 Q^S)[x := V^S] \\ &\rightarrow_{\sigma}^* P^S[x := V^S]@_1 Q^S[x := V^S] && \text{(Lemma 13)} \\ &\rightarrow_{\sigma}^* P[x := V]^S@_1 Q[x := V]^S && \text{(induction hypothesis and Lemma 14 (1), (3))} \\ &= (P[x := V]Q[x := V])^S && \text{(Lemma 8 (1))} \\ &= (PQ)[x := V]^S. \end{aligned}$$

If P is a value, it is proved similarly. The other cases are straightforward.

(2) By induction on the size of M . If $M = (\lambda y.P)W$, we have

$$\begin{aligned}
 ((\lambda y.P)W)^B[x := V^B] &= P^B[y := W^B][x := V^B] && \text{(definition of B)} \\
 &= P^B[x := V^B][y := W^B[x := V^B]] \\
 &\rightarrow_{\beta_v}^* P[x := V]^B[y := W[x := V]^B] && \text{(induction hypothesis)} \\
 &= ((\lambda y.P[x := V])W[x := V])^B && \text{(definition of B)} \\
 &= ((\lambda y.P)W)[x := V]^B.
 \end{aligned}$$

If $M = xW$ and $V = \lambda y.P$, we have

$$\begin{aligned}
 (xW)^B[x := (\lambda y.P)^B] &= (\lambda y.P^B)W^B[x := \lambda y.P^B] \\
 &\rightarrow_{\beta_v} P^B[y := W^B[x := \lambda y.P^B]] \\
 &\rightarrow_{\beta_v}^* P^B[y := W[x := \lambda y.P]^B] && \text{(induction hypothesis)} \\
 &= ((\lambda y.P)W[x := \lambda y.P])^B \\
 &= (xW)[x := \lambda y.P]^B.
 \end{aligned}$$

The other cases are straightforward. □

Lemma 17. For any terms M and N , we have $M^B N^B \rightarrow_{\beta_v}^* (MN)^B$.

Proof. If M is a value, we have $M^B N^B = (xN)^B[x := M^B] \rightarrow_{\beta_v}^* (xN)[x := M]^B = (MN)^B$ by Lemma 16. If M is not a value, we have $M^B N^B = (MN)^B$. □

Proposition 18. For any terms M and M' , if $M \rightarrow_{\sigma} M'$ holds, then $M^B \rightarrow_{\sigma}^* M'^B$ holds.

Proof. By induction on a derivation of $M \rightarrow_{\sigma} M'$.

First, the base cases are proved as follows.

If $M = (\lambda x.P)VK$ and $M' = (\lambda x.PK)V$, we have

$$\begin{aligned}
 ((\lambda x.P)VK)^B &= P^B[x := V^B]K^B && \text{(definition of B)} \\
 &= (P^B K^B)[x := V^B] \\
 &\rightarrow_{\beta_v}^* (PK)^B[x := V^B] && \text{(Lemma 17)} \\
 &= ((\lambda x.PK)V)^B && \text{(definition of B)}.
 \end{aligned}$$

If $M = (\lambda x.P)QK$, $M' = (\lambda x.PK)Q$, and Q is not a value, we have

$$\begin{aligned}
 ((\lambda x.P)QK)^B &= ((\lambda x.P^B)Q^B K^B) && \text{(definition of B)} \\
 &\rightarrow_{\sigma} ((\lambda x.P^B K^B)Q^B) \\
 &\rightarrow_{\beta_v}^* (\lambda x.(PK)^B)Q^B && \text{(Lemma 17)} \\
 &= ((\lambda x.PK)Q)^B && \text{(definition of B)}.
 \end{aligned}$$

The case $M = V((\lambda x.P)Q)$ and $M' = (\lambda x.VP)Q$ is proved similarly by Lemma 17.

Next, the induction cases are proved as follows.

If $M = \lambda x.N$, $M' = \lambda x.N'$, and $N \rightarrow_{\sigma} N'$, by the induction hypothesis, we have $N^B \rightarrow_{\sigma}^* N'^B$. Then, we have $(\lambda x.N)^B = \lambda x.N^B \rightarrow_{\sigma}^* \lambda x.N'^B = (\lambda x.N')^B$.

If $M = (\lambda x.P)V$, $M' = (\lambda x.P')V'$, and either $P \rightarrow_{\sigma} P'$ and $V = V'$ or $V \rightarrow_{\sigma} V'$ and $P = P'$, we have

$$\begin{aligned} ((\lambda x.P)V)^B &= P^B[x := V^B] && \text{(definition of B)} \\ &\rightarrow^* P^B[x := V'^B] && \text{(induction hypothesis)} \\ &= ((\lambda x.P')V')^B && \text{(definition of B)}. \end{aligned}$$

The other cases are straightforward. □

The last subgoal is to prove the condition (4) of Theorem 4, that is, $((\cdot)^S)^B$ is weakly Z for \rightarrow_{β_v} by \rightarrow (Proposition 20).

Lemma 19. *For any terms M and M' , we have the following.*

- (1) *If $M \rightarrow_{\beta_v} M'$ holds, then $M^S \rightarrow^* M'^S$ holds.*
- (2) *If $M \rightarrow_{\beta_v} M'$ holds, then $M^B \rightarrow^* M'^B$ holds.*

Proof. (1) By induction on a derivation of $M \rightarrow_{\beta_v} M'$.

In the base case, we suppose $M = (\lambda x.P)V$ and $M' = P[x := V]$. We have $((\lambda x.P)V)^S = (\lambda x.P^S)@_3 V^S = (\lambda x.P^S)V^S \rightarrow_{\beta_v} P^S[x := V^S] \rightarrow^*_{\beta_v} P[x := V]^S$ by Lemma 16.

The induction cases are proved as follows.

If $M = PQ$, $M' = P'Q'$, P is not a value, and either $P \rightarrow_{\beta_v} P'$ and $Q = Q'$ or $Q \rightarrow_{\beta_v} Q'$ and $P = P'$, we have $P^S \rightarrow^* P'^S$ and $Q^S \rightarrow^* Q'^S$ by the induction, and then

$$\begin{aligned} (PQ)^S &= P^S@_1 Q^S \\ &\rightarrow^* P'^S@_1 Q'^S && \text{(Lemma 14)} \\ &\rightarrow^*_{\sigma} (P'Q')^S && \text{(Lemma 8)}. \end{aligned}$$

If P is a value, it is proved similarly.

The case $\lambda x.P \rightarrow_{\beta_v} \lambda x.P'$ is straightforward by the induction hypothesis.

- (2) It is straightforwardly proved by induction on a derivation of $M \rightarrow_{\beta_v} M'$, using Lemma 16. □

Proposition 20. *The mapping $((\cdot)^S)^B$ is weakly Z for \rightarrow_{β_v} by \rightarrow .*

Proof. In this proof, we write M^{SB} for $(M^S)^B$.

First, we prove that $M \rightarrow_{\beta_v} M'$ implies $M' \rightarrow^* M^{SB}$ by induction on a derivation of $M \rightarrow_{\beta_v} M'$.

In the base case, we suppose $M = (\lambda x.P)V$ and $M' = P[x := V]$. We have $M^{SB} = ((\lambda x.P^S)@_3 V^S)^B = ((\lambda x.P^S)V^S)^B = P^{SB}[x := V^{SB}]$, and $P[x := V] \rightarrow^* P^{SB}[x := V^{SB}]$ holds by Lemma 10.

The induction cases are proved as follows.

If $M = PQ$, $M = P'Q'$, and either $P \rightarrow_{\beta_v} P'$ and $Q = Q'$ or $Q \rightarrow_{\beta_v} Q'$ and $P = P'$, we have $P' \rightarrow^* P^{SB}$ and $Q' \rightarrow^* Q^{SB}$ by the induction hypothesis and Lemma 10, and then we have $P'Q' \rightarrow^* P^{SB}Q^{SB} \rightarrow^* (PQ)^{SB}$ by Lemmas 8 and 17.

The case $\lambda x.P \rightarrow_{\beta_v} \lambda x.P'$ is straightforward by the induction hypothesis.

Secondly, we prove that $M \rightarrow_{\beta_v} M'$ implies $M^{SB} \rightarrow^* M'^{SB}$. If $M \rightarrow_{\beta_v} M'$ holds, we have $M^S \rightarrow^* M'^S$ by Lemma 19 (1), and then, we have $M^{SB} \rightarrow^* M'^{SB}$ by Proposition 18 and Lemma 19 (2). □

We have proved the conditions of Theorem 4, and hence, we give a confluence proof of λ_v^{σ} .

Theorem 21. *The mapping $((-)^S)^B$ is Z for \rightarrow of λ_v^σ , and hence λ_v^σ is confluent.*

Proof. It is sufficient to prove the conditions of Theorem 4. (1) is proved by Propositions 12 and 15. (2) is proved by Proposition 18. (3) is proved by Lemma 10. (4) is proved by Proposition 20. \square

5. Conclusion

This paper gives a new proof of confluence using the compositional Z theorem for λ_v^σ , a variant of the call-by-value lambda calculus, which contains permutation reductions. This is a new example of an application of the compositional Z theorem. Compared with the existing proof with Hindley-Rosen's method in Carraro and Guerrieri (2014), our proof does not depend on any normalization property of reduction relations.

Acknowledgements. This work was partially supported by Grants-in-Aid for Scientific Research KAKENHI 18K11161 (to Koji Nakazawa) and 17K05343 and 20K03711 (to Ken-etsu Fujita).

References

- Accattoli, B. and Guerrieri, G. (2016). Open call-by-value. In: *Asian Symposium on Programming Languages and Systems (APLAS 2016)*, Lecture Notes in Computer Science, vol. 10017, 206–226.
- Accattoli, B. and Kesner, D. (2012). The permutative λ -calculus. In: *Proceedings of the International Conference on Logic Programming and Automated Reasoning (LPAR 2012)*, Lecture Notes in Computer Science, vol. 7180, 15–22.
- Ando, Y. (2003). Church-Rosser property of a simple reduction for full first-order classical natural deduction. *Annals of Pure and Applied Logic* **119** 225–237.
- Carraro, A. and Guerrieri, G. (2014). A semantical and operational account of call-by-value solvability. In: *Foundations of Software Science and Computation Structures (FoSSaCS 2014)*, Lecture Notes in Computer Science, vol. 8412, Springer, 103–118.
- Dehornoy, P. and van Oostrom, V. (2008). Z, proving confluence by monotonic single-step upperbound functions. In: *Logical Models of Reasoning and Computation (LMRC-08)*.
- Fujita, K. (2020). A formal system of reduction paths for parallel reduction. *Theoretical Computer Science* **813** 327–340.
- Honda, Y., Nakazawa, K. and Fujita, K. (2021). Confluence proofs of lambda-mu-calculi by Z theorem. *Studia Logica* **109** 917–936.
- Komori, Y., Matsuda, N. and Yamakawa, F. (2013). A simplified proof of the church-rosser theorem. *Studia Logica* **102** (1) 175–183.
- Nakazawa, K. and Fujita, K. (2016). Compositional Z: Confluence proofs for permutative conversion. *Studia Logica* **104** 1205–1224.
- Nakazawa, K., Fujita, K. and Imagawa, Y. (2017). Z for call-by-value. In: *Proceedings of the 6th International Workshop on Confluence*, 57–61.
- Nakazawa, K. and Nagai, T. (2014). Reduction system for extensional lambda-mu calculus. In: *25th International Conference on Rewriting Techniques and Applications joint with the 12th International Conference on Typed Lambda Calculi and Applications (RTA-TLCA 2014)*, Lecture Notes in Computer Science, vol. 8560, 349–363.
- Plotkin, G. (1975). Call-by-name, call-by-value and the λ -calculus. *Theoretical Computer Science* **1** 125–159.
- van Oostrom, V. (2021). Z; syntax-free developments. In: *6th International Conference on Formal Structures for Computation and Deduction (FSCD 2021)*, Leibniz International Proceedings in Informatics, vol. 195, 24:1–24:22.