A study for recovering the cut-elimination property in cyclic proof systems by restricting the arity of inductive predicates

Yukihiro Masuoka∗ Daisuke Kimura†

March 14, 2022

The framework of cyclic proof systems provides a reasonable proof system for logics with inductive definitions. It also offers an effective automated proof search procedure for such logics without finding induction hypotheses. Recent researches have shown that the cut-elimination property, one of the most fundamental properties in proof theory, of cyclic proof systems for several logics does not hold. These results suggest that a naive proof search, which avoids the Cut rule, is not enough.

This paper shows that the cut-elimination property still fails in a simple cyclic proof system even if we restrict languages to unary inductive predicates and unary functions, aiming to clarify why the cut-elimination property fails in the cyclic proof systems. The result in this paper is a sharper one than that of the first authors’ previous result, which gave a counterexample using two ternary inductive predicates and a unary function symbol to show the failure of the cut-elimination property in the cyclic proof system of the first-order logic.

1. Introduction

Inductive definition is a way to define mathematical objects based on the induction principle. Several notions, which are essential in both mathematics and computer science, such as natural numbers, lists, and binary trees, are inductively defined. Inductively defined predicates are called inductive predicates. A typical example of inductive predicates is $N(x)$ that means “$x$ is a natural number”. It is given by the following Martin Lof style schemata (called productions) [12]:

$$
\frac{}{N(0)} \quad \text{and} \quad \frac{N(x)}{N(sx)},
$$

where 0 is a constant symbol, and $s$ is a unary function symbol. The first production means that $N(0)$ holds without any assumptions, namely, it says that “0 is a natural number”. The second one means that $N(x)$ implies $N(sx)$, namely, it says that “if $x$ is a natural number, then $sx$ is also a natural number”. It is also assumed that no other rule can be applied to obtain $N(t)$ for any term $t$. Hence, these productions say that the predicate $N$ is defined as the least one that satisfies the following equivalence:

$$
N(x) \Leftrightarrow (x = 0 \lor \exists y.(x = sy \land N(y))).
$$

This equivalence gives an alternative definition of the natural number predicate $N$ instead of giving its productions.

Proof systems for logics with inductive predicates have been studied in the literature [12, 19]. It is known that Gentzen’s sequent calculus LK can be extended with inductive predicates in a uniform way. For example, the following inference rules for the natural number predicate $N$ are generated from the productions of $N$:

$$
\frac{}{\Gamma \vdash \Delta, N(0)}, \quad \frac{\Gamma \vdash \Delta, N(t)}{\Gamma \vdash \Delta, N(st)}, \quad \frac{\Gamma \vdash \Delta, F[0]}{\Gamma \vdash \Delta, F[st]}, \quad \frac{\Gamma, F[x] \vdash \Delta, F[sx]}{\Gamma, \Delta \vdash \Delta} \quad \text{(IND)}.
$$

∗The Graduate University for Advanced Studies, yukihiro.m [at] nii.ac.jp
†Toho University, kmr [at] is.sci.toho-u.ac.jp
where \( \Gamma \) and \( \Delta \) are multisets of formulas, \( x \) is a fresh variable, \( t \) is a term, \( F \) is a formula with a fixed variable \( z \), and \( F[t] \) is the result of substituting \( t \) for \( z \) in \( F \). The last inference rule (IND) corresponds to the induction principle on the natural numbers, and the formula \( F \) is its induction hypothesis. Although the last rule is a reasonable formalization of the induction principle, it causes a difficult problem when we apply a naive proof search algorithm to this proof system because we need to find (or guess) an appropriate induction hypothesis \( F \) in the upper sequents from the lower sequent.

An alternative choice to formulate inductive predicates in sequent calculi is to adopt rules, instead of the (IND) rules, that unfold inductive predicates on the left-hand side of a sequent, according to the equivalence that defines the predicate. Brotherston and Simpson [4] proposed the proof systems (called LKID\( ^\omega \) and CLKID\( ^\omega \)) for classical first-order logic, based on this idea. In their systems, the unfolding rules (called the casesplit rule) for the natural number predicate \( N \) is given as follows:

\[
\frac{\Gamma, t = 0 \vdash \Delta}{\Gamma, N(0) \vdash \Delta} \quad (\text{Case N}),
\]

where \( y \) is a fresh variable.

Although this idea gives a solution to avoid the problem of finding induction hypotheses in proof search, it requires considering infinite proofs, which might have infinite paths. The system LKID\( ^\omega \) is a proof system that admits such infinite proofs which satisfy a condition (called the global trace condition) that ensures the soundness of the system. The cyclic proof system CLKID\( ^\omega \) is a reasonable restriction of LKID\( ^\omega \) that admits only proofs which are regular trees. It is formulated as a finite derivation tree with open assumptions (called buds) and additional edges that connects from each bud to an internal node (called companion of the bud).

The framework of cyclic proof systems gives a general way to formalize logics with inductive definitions (or the least/greatest fixed point operators). It has been proposed several cyclic proof systems other than classical first-order logic, such as logic of bunched implications [3], separation logic [5], and linear logic with the fixed point operators [1, 9].

The cut-elimination property of a proof system states that any provable sequent in the system is also provable without the rule (CUT), which is given below:

\[
\frac{\Gamma \vdash \Delta, F \quad F \vdash \Delta}{\Gamma \vdash \Delta} \quad (\text{CUT}).
\]

This property is one of the most fundamental properties of proof systems because it helps us to investigate a given proof system. For example, some important properties, such as the subformula property and consistency of the proof system, are often obtained using the cut-elimination property. It is also important for the proof search method since it ensures that it is enough to search except for (CUT), which requires to find an appropriate cut formula from possibly infinitary many candidates.

Some infinite proof systems are known to enjoy the cut-elimination property: Brotherston and Simpson proved that the cut-elimination theorem of LKID\( ^\omega \) by showing LKID\( ^\omega \) is sound and cut-free complete to a standard model for inductive predicates [4]. Fortier and Santocanale [10] introduced a cyclic proof system for additive linear logic with the least and greatest fixed point operators, and showed that the rule (CUT) can be eliminated if we admit to lose the regularity of proof-trees. Doumane [9] investigated an infinite proof system \( \mu\mathsf{MALL}^\infty \) for the multiplicative and additive linear logic with the least and greatest fixed point operators, and showed its cut-elimination theorem.

In contrast, the situation for cyclic proof systems is totally different. The open problem about the cut-elimination property of CLKID\( ^\omega \) by Brotherston was negatively solved in the first authors' recent work [13]. The second author showed that the cut-elimination property does not hold in a cyclic proof system of separation logic [11]. Saotome showed the failure of the cut-elimination property for the cyclic proof system of the logic of bunched implications even if we restrict inductive predicates to nullary predicates [15].

There are automated theorem provers based on proof search algorithms of cyclic proof systems [5, 6, 7, 16, 17, 18]. Some of them adopt an additional mechanism that guesses cut formulas. They record sequents which are found during an execution of proof search procedure, and then try to generate possible candidates of cut formulas from the recorded sequents [7, 8, 16, 17]. This technique extends the ability of the provers to find cyclic proofs which may contain (CUT), and also gives an efficient proof search procedure. However, Saotome [14] suggested that there still exist sequents in the symbolic heap separation logic that cannot be found by a normal proof search procedure admitting (CUT) whose cut formulas are assumable from the goal sequent. Recently the framework of cyclic proof-search has been studied from the viewpoint of software verification [20, 21].

In our recent research, we have fixed our attention to the following two natural questions about the cut-elimination property in cyclic proofs.

2
• Can we recover the cut-elimination property of cyclic proof systems (in particular CLKID$^\omega$) by restricting the definitions of inductive predicates?

• Is there a reasonable restriction of (Cut) whose cut formulas can be found in small search space, and that does not lose provability of the original cyclic proof system.

This paper focuses on the first question. We show that the cut-elimination property of CLKID$^\omega$ still fails even if we restrict inductive predicates to two unary predicates ($\epsilon$ and $\omega$) and one unary function symbol ($n$). The proof technique of this paper is a modified and simplified one of the first author’s previous work [13]. Our discussion starts from introducing a simple subsystem (called the base system $B$) that only contains equality and inductive predicates because other logical connectives and quantifiers are not necessary. Then we define a cyclic proof system (called $CB^\omega$) of $B$, and apply the proof technique to our counterexample $TeF(\epsilon) \vdash FsT(\epsilon)$. This counterexample also works for CLKID$^\omega$ since we can easily check that if $TeF(\epsilon) \vdash FsT(\epsilon)$ is cut-free provable in CLKID$^\omega$, then its proof is also a cut-free proof in $CB^\omega$.

The reminder of this paper is structured as follows. Section 2 introduces the base system $B$. Section 3 defines the cyclic proof system $CB^\omega$ of $B$. In Section 4 we show the main theorem by giving our counterexample for the cut-elimination property of $CB^\omega$. Section 5 concludes.

2. The base system

In this section, we present the base system $B$, which only contains equations and inductive definitions. It is a subsystem of the first-order logic with inductive definitions FOL$_{ID}$ [4]. The system $B$ is developed from a language that consists of countable number of variable symbols (denoted by $x, y, z, \ldots$) and arbitrary number of function symbols (denoted by $f$), finite number of ordinary predicate symbols (denoted by $Q_1, \ldots, Q_m$), and finite number of inductive predicate symbols (denoted by $P_1, \ldots, P_n$), where $f, Q_i,$ and $P_j$ have their own arities $arity(f), arity(Q_i),$ and $arity(P_j),$ respectively. A function symbol with arity zero is called a constant symbol. We use $R$ for a meta variable that ranges over both ordinary predicate symbols and inductive predicate symbols.

Terms (denoted by $t$ and $u$) for $B$ are defined by

$$ t ::= x \mid f^t \ldots t, $$

where $n = arity(f)$.

For a unary function symbol $f$, we use an abbreviation $f^nt$ for $f \cdots ft$ ($n$ times of $f$).

We write $x$ for a sequence of variables and $t$ for a sequence of terms. We also write $t(x)$ for $t$ in which the variables $x$ occur. The length of a sequence $t$ is written $|t|$.

A formula (denoted by $\varphi$) of $B$ is defined as follows:

$$ \varphi ::= t = t \mid R(t), $$

where $|t| = arity(R)$. We define free variables as usual, and $FV(\varphi)$ is defined as the set of free variables in $\varphi$.

We write $\varphi[x_0 := t_0, \ldots, x_r := t_r]$ for a formula obtained from a formula $\varphi$ by simultaneously substituting terms $t_0, \ldots, t_r$ for variables $x_0, \ldots, x_r$, respectively. We sometimes write $\theta$ for $x_0 := t_0, \ldots, x_r := t_r$.

Inductive predicate symbols are given with an inductive definition set, which is defined as follows.

Definition 1 (Inductive definition set). A production for $P_j$ is defined as

$$ Q_1(u_1) \cdots Q_h(u_h) \quad P_{j_1}(t_1) \cdots \quad P_{j_m}(t_m). $$

The formulas above the line of a production are called the assumptions of the production. The formula under the line of a production is called the conclusion of the production. An inductive definition set is a finite set of productions.

The unary inductive predicates $TeF$ and $FsT$ given in the next example are important in this paper because they will work as a counterexample for showing the failure of cut-elimination in a cyclic proof system.

Example 2. The productions for $TeF$ and $FsT$ are given as follows:

$$ Q_1(u_1) \cdots Q_h(u_h) \quad P_{j_1}(t_1) \cdots \quad P_{j_m}(t_m). $$
\[
\begin{align*}
\Gamma \vdash \Delta \quad & \text{(Axiom) } (\Gamma \cap \Delta \neq \emptyset) \\
\Gamma, \varphi, \Delta \vdash \Gamma \quad & \text{(Weak) } (\Gamma' \subseteq \Gamma, \Delta' \subseteq \Delta) \\
\frac{\Gamma, \varphi \vdash \Delta}{\Gamma \vdash \Delta} & \text{ (Cut) } \\
\frac{\Gamma \vdash \varphi, \Delta \quad \Gamma \vdash \varphi, \Delta}{\Gamma \vdash \Delta} & \text{ (Subst) } \\
\frac{\Gamma \vdash [x := u, y := t], t = u \vdash \Delta [x := u, y := t]}{\Gamma \vdash \Delta \vdash [x := t, y := u]} & \text{ (= L') } \\
\frac{\Gamma \vdash \Delta \quad \Gamma \vdash \Delta}{\Gamma \vdash \Delta} & \text{ (= R) }
\end{align*}
\]

Figure 1: Inference rules except rules for inductive predicates

\[
\begin{array}{c}
\frac{\text{TeF}(s)}{\text{TeF}(e)} \\
\frac{\text{TeF}(nx)}{\text{TeF}(x)} \\
\frac{\text{FsT}(\theta)}{\text{FsT}(x)} \\
\frac{\text{FsT}(nx)}{\text{FsT}(nx)}
\end{array}
\]

where \(s\) (start) and \(e\) (end) are constant symbols, \(nx\) (“next of \(x\)”) is a unary function symbol.

Intuitively, \(\text{TeF}(t)\) (read “to \(e\) from \(t\)”) means that, for some \(m \geq 0\), “the \(m\)-th next element of \(t\) is \(e\)”, since \(\text{TeF}(t)\) holds for \(t\) such that \(e = n^m t\) for some \(m \geq 0\). Also, \(\text{FsT}(t)\) (read “from \(s\) to \(t\)”) means that, for some \(m \geq 0\), “the \(m\)-th next element from \(s\) is \(t\)”, since \(\text{FsT}(t)\) holds for \(t\) such that \(t = n^m s\) for some \(m \geq 0\). Hence \(\text{TeF}(s)\) and \(\text{FsT}(e)\) are semantically same, that is, they both mean \(e = n^m s\) for some \(m \geq 0\).

The semantics of an inductive predicate is given by the standard least fixed point semantics, namely, the least fixed point of a monotone operator constructed from its productions (see [4]). We skip giving its detailed definition because we do not use semantics in this paper.

**Definition 3** (Sequent). Let \(\Gamma\) and \(\Delta\) be finite sets of formulas in \(B\). A sequent (denoted by \(S\)) of \(B\) is a pair \(\Gamma \vdash \Delta\). The first set \(\Gamma\) is called the antecedent of \(\Gamma \vdash \Delta\) and the second one \(\Delta\) is called the succedent of \(\Gamma \vdash \Delta\).

We use usual abbreviations like \(\Gamma, \varphi \vdash \varphi', \Delta\) for \(\Gamma \cup \{\varphi\} \vdash \{\varphi'\} \cup \Delta\), and \(\Gamma[\theta]\) for \(\{\varphi[\theta] \mid \varphi \in \Gamma\}\). We define \(\text{FV}(\Gamma)\) as the union of free variables of formulas in \(\Gamma\).

### 3. Cyclic proof system \(CB^\omega\) for the base system

In this section, we define a cyclic proof system \(CB^\omega\) for the base system \(B\). To define it, we first define an infinitary proof system \(B^\omega\) for \(B\) in the subsection 3.2. After that, \(CB^\omega\) is defined in the subsection 3.3.

#### 3.1. Inference rules

This section gives the common inference rules for both \(B^\omega\) and \(CB^\omega\). The inference rules except for rules of inductive predicates are given in Figure 1. The sequents above the line of a rule are called the assumptions of the rule. The sequent under the line of a rule is called the conclusion of the rule. The principal formula of a rule is the distinguished formula in its conclusion. The distinguished formulas (\(\varphi\) in Figure 1) of (Cut) are called the cut-formulas.

We note that \(\Gamma, \varphi, \varphi' \vdash \Delta, \varphi', \varphi'\) is identified with \(\Gamma, \varphi \vdash \Delta, \varphi'\), since \(\Gamma \cup \{\varphi, \varphi\} = \Gamma \cup \{\varphi\}\) and \(\Delta \cup \{\varphi', \varphi'\} = \Delta \cup \{\varphi'\}\). Hence we do not have the contraction rule as an explicit inference rule.

From a technical reason, we adopt \((= L')\) instead of \((= L)\), which is given by:

\[
\frac{\Gamma \vdash [x := u, y := t], t = u \vdash \Delta \vdash [x := t, y := u]}{\Gamma \vdash \Delta \vdash [x := u, y := t]} \quad (= L')
\]

These rules are derivable each other without adding extra (Cut): \((= L)\) is derivable by applying (Weak) and \((= L')\), and \((= L')\) is derivable by applying \((= L)\).

We present the two inference rules for inductive predicates. First, for each production

\[
\frac{Q_1(u_1(x)) \cdots Q_n(u_n(x)) \quad P_j(t_1(x)) \cdots P_m(t_m(x))}{P_i(t(x))}
\]

there is the inference rule
\[
\Gamma \vdash Q_1(u_1(u)), \Delta \quad \Gamma \vdash Q_h(u_h(u)), \Delta \quad \Gamma \vdash P_{i_1}(t_1(u)), \Delta \quad \cdots \quad \Gamma \vdash P_{i_m}(t_m(u)), \Delta
\]

\[
\Gamma \vdash P_i(t(u)), \Delta
\]

Next, we define the left introduction rule for the inductive predicate. A case distinction of \( \Gamma, P_i(u) \vdash \Delta \) is defined as a sequent

\[
\Gamma, u = t(y), Q_1(u_1(y)), \ldots, Q_h(u_h(y)), P_{i_1}(t_1(y)), \ldots, P_{i_m}(t_m(y)) \vdash \Delta,
\]

where \( y \) is a sequence of distinct variables of the same length as \( x \) and \( y \not\in \text{FV}(\Gamma \cup \Delta \cup \{P_i(u)\}) \) for all \( y \in y \), and there is a production

\[
Q_1(u_1(x)) \quad \ldots \quad Q_h(u_h(x)) \quad P_{i_1}(t_1(x)) \quad \ldots \quad P_{i_m}(t_m(x))
\]

\[
P_i(t(x))
\]

The inference rule (CASE \( P_i \)) is

\[
\frac{\text{All case distinctions of } \Gamma, P_i(u) \vdash \Delta}{\Gamma, P_i(u) \vdash \Delta} \quad (\text{CASE } P_i)
\]

The formulas \( P_{i_1}(t_1(y)), \ldots, P_{i_m}(t_m(y)) \) in case distinctions are said to be case-descendants of the principal formula \( P_i(u) \).

**Example 4.** The inference rules for the natural number predicate \( \mathbb{N} \) are

\[
\frac{\Gamma \vdash \Delta, N(0)}{(NR_1)} \quad \frac{\Gamma \vdash \Delta, N(t)}{(NR_2)} \quad \frac{\Gamma, t = 0 \vdash \Delta}{\Gamma, N(0) \vdash \Delta} \quad \frac{\Gamma, t = sy, N(y) \vdash \Delta}{\Gamma, N(t) \vdash \Delta} \quad (\text{CASE } N)
\]

where \( y \) is a fresh variable.

**Example 5.** The inference rules for the inductive predicates \( \text{TeF} \) and \( \text{FsT} \) given in Example 2 are as follows:

\[
\frac{\Gamma \vdash \Delta, \text{TeF}(e)}{(\text{TeF } R_1)} \quad \frac{\Gamma \vdash \Delta, \text{TeF}(nt)}{(\text{TeF } R_2)} \quad \frac{\Gamma, t = e \vdash \Delta}{\Gamma, \text{TeF}(e) \vdash \Delta} \quad \frac{\Gamma, t = y, \text{TeF}(ny) \vdash \Delta}{\Gamma, \text{TeF}(t) \vdash \Delta} \quad (\text{CASE } \text{TeF})\text{, where } y \text{ is a fresh variable};
\]

\[
\frac{\Gamma \vdash \Delta, \text{FsT}(s)}{(\text{FsT } R_1)} \quad \frac{\Gamma \vdash \Delta, \text{FsT}(nt)}{(\text{FsT } R_2)} \quad \frac{\Gamma, t = s \vdash \Delta}{\Gamma, \text{FsT}(s) \vdash \Delta} \quad \frac{\Gamma, t = ny, \text{FsT}(y) \vdash \Delta}{\Gamma, \text{FsT}(t) \vdash \Delta} \quad (\text{CASE } \text{FsT})\text{, where } y \text{ is a fresh variable}.
\]

### 3.2. Infinitary proof system \( B^\omega \)

In this subsection, we define an infinitary proof system \( B^\omega \) for \( B \). The inference rules of \( B^\omega \) are the rules displayed in Figure 1 and the rules for inductive predicates given in the previous subsection.

We write \( \langle n_1, \ldots, n_k \rangle \) for the sequence of natural numbers \( n_1, \ldots, n_k \). The length \( |\sigma| \) of a sequent \( \sigma \) is defined by the number of elements in \( \sigma \). Let \( \mathbb{N}^* \) be the set of finite sequences of natural numbers. We write \( \sigma_1 \sigma_2 \) for the concatenation of \( \sigma_1 \) and \( \sigma_2 \) in \( \mathbb{N}^* \). We abbreviate \( \sigma \langle n \rangle \) by \( \sigma n \) for \( \sigma \in \mathbb{N}^* \) and \( n \in \mathbb{N} \).

Let Rule and Seq be the set of names of the inference rules and the set of sequents of \( B^\omega \), respectively.

**Definition 6** (Derivation tree). We define a derivation tree to be a partial function \( \mathcal{D} : \mathbb{N}^* \rightarrow \text{Seq} \times (\text{Rule} \cup \{\text{BUD}\}) \) satisfying the following conditions:

(1) The domain \( \text{dom}(\mathcal{D}) \) of \( \mathcal{D} \) is prefixed-closed, namely, for \( \sigma_1, \sigma_2 \in \mathbb{N}^* \), \( \sigma_1 \sigma_2 \in \text{dom}(\mathcal{D}) \) implies \( \sigma_1 \in \text{dom}(\mathcal{D}) \).

(2) If \( \sigma n \in \text{dom}(\mathcal{D}) \) for \( \sigma \in \mathbb{N}^* \) and \( n \in \mathbb{N} \), then \( \sigma m \in \text{dom}(\mathcal{D}) \) for any \( m \leq n \).

(3) For each \( \sigma \in \text{dom}(\mathcal{D}) \), we write \( (\text{Seq}(\mathcal{D}, \sigma), \text{Rule}(\mathcal{D}, \sigma)) \) for \( \mathcal{D}(\sigma) \). Then the following hold.
(a) If Rule($\mathcal{D}, \sigma$) = (Bud), then $\sigma_0 \notin \text{dom}(\mathcal{D})$.

(b) If Rule($\mathcal{D}, \sigma$) $\neq$ (Bud), $\sigma(n + 1) \notin \text{dom}(\mathcal{D})$, and $\sigma_0, \ldots, \sigma_n \in \text{dom}(\mathcal{D})$, then the following is a rule instance of the rule Rule($\mathcal{D}, \sigma$):

$$\frac{\text{Seq}(\mathcal{D}, \sigma_0) \quad \cdots \quad \text{Seq}(\mathcal{D}, \sigma_n)}{\text{Seq}(\mathcal{D}, \sigma)} \text{ Rule}(\mathcal{D}, \sigma)$$

An element in the domain of a derivation tree is called a node. The empty sequence as a node is called the root. The node $\sigma$ is called a bud if Rule($\mathcal{D}, \sigma$) is (Bud). We write bud($\mathcal{D}$) for the set of buds in $\mathcal{D}$. The node which is not a bud is called an inner node. A derivation tree is called infinite if the domain of the derivation tree is infinite.

We sometimes identify a node $\sigma$ with the sequent Seq($\mathcal{D}, \sigma$).

**Definition 7** (Path). We define a path in a derivation tree $\mathcal{D}$ to be a (possibly infinite) sequence $(\sigma_i)_{0 \leq i < \alpha}$ of nodes in dom($\mathcal{D}$) such that $\sigma_{i+1} = \sigma_i n$ for some $n \in \mathbb{N}$ and $\alpha \in \mathbb{N}_0 \cup \{\omega\}$, where $\mathbb{N}_0$ is the set of positive natural numbers and $\omega$ is the least infinite ordinal. A finite path $\sigma_0, \sigma_1, \ldots, \sigma_n$ is called a path from $\sigma_0$ to $\sigma_n$.

The length of a finite path $(\sigma_i)_{0 \leq i < \alpha}$ is defined as $\alpha$. We define the height of a node as the length of the path from the root to the node.

We sometimes write $(\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}$ for the path $(\sigma_i)_{0 \leq i < \alpha}$ in a derivation tree $\mathcal{D}$ if $\text{Seq}(\mathcal{D}, \sigma_i) = \Gamma_i \vdash \Delta_i$.

**Definition 8** (Trace). For a path $(\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}$ in a derivation tree $\mathcal{D}$, we define a trace following $(\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}$ to be a sequence of formulas $(\tau_i)_{0 \leq i < \alpha}$ such that the following hold:

1. $\tau_i$ is an inductive predicate in $\Gamma_i$.
2. If $\Gamma_i \vdash \Delta_i$ is the conclusion of (SUBST) with $\theta$, then $\tau_i$ is $\tau_{i+1}[\theta]$.
3. If $\Gamma_i \vdash \Delta_i$ is the conclusion of ($= L'$) with the principal formula $t = u$ and $\tau_i$ is $\varphi[x := t, y := u]$, then $\tau_{i+1}$ is $\varphi[x := u, y := t]$.
4. If $\Gamma_i \vdash \Delta_i$ is the conclusion of (CASE $P_i$), then either
   - $\tau_i$ is the principal formula of the rule and $\tau_{i+1}$ is a case-descendant of $\tau_i$, or
   - $\tau_{i+1}$ is the same as $\tau_i$.

In the former case, $\tau_i$ is said to be a progress point of the trace.

5. If $\Gamma_i \vdash \Delta_i$ is the conclusion of any other rules and $i + 1 < \alpha$, then $\tau_{i+1}$ is $\tau_i$.

**Definition 9** (Global trace condition). If a trace has infinitely many progress points, we call the trace an infinitely progressing trace. If there exists an infinitely progressing trace following a tail of the path $(\Gamma_i \vdash \Delta_i)_{i \geq k}$ with some $k \geq 0$ for every infinite path $(\Gamma_i \vdash \Delta_i)_{i \geq 0}$ in a derivation tree, we say the derivation tree satisfies the global trace condition.

**Definition 10** ($\mathcal{B}\omega$ pre-proof). A (possibly infinite) derivation tree $\mathcal{D}$ without buds is called a $\mathcal{B}\omega$ pre-proof. The sequent Seq($\mathcal{D}, \langle \rangle$) at the root node is called the conclusion of $\mathcal{D}$.

**Definition 11** ($\mathcal{B}\omega$ proof). A $\mathcal{B}\omega$ pre-proof that satisfies the global trace condition is called a $\mathcal{B}\omega$ proof.

The global trace condition was originally introduced as a sufficient condition for the soundness of LKID$\omega$ with respect to the standard models in Brotherston’s paper [2, 4]. It also ensures the soundness of $\mathcal{B}\omega$, since a $\mathcal{B}\omega$ proof can be transformed to a LKID$\omega$ proof by replacing ($= L'$) by ($= L$). Brotherston also showed the cut-free completeness of LKID$\omega$ for the standard models. The cut-free completeness of $\mathcal{B}\omega$ follows from this result: a cut-free LKID$\omega$ proof of a sequent in $\mathcal{B}\omega$ can contain only the rules (AXIOM), (WEAK), (SUBST), ($= L$), ($= R$), and the rules for inductive predicates. Hence the cut-free proof can be transformed to a cut-free $\mathcal{B}\omega$ proof by replacing ($= L$) by ($= L'$) with (WEAK).
3.3. Cyclic proof system $\text{CB}^\omega$

In this section, we introduce a cyclic proof system $\text{CB}^\omega$.

**Definition 12** (Companion). For a finite derivation tree $D$, we define the companion for a bud $\sigma_{bud}$ as an inner node $\sigma$ in $D$ with $\text{Seq}(D, \sigma) = \text{Seq}(D, \sigma_{bud})$.

**Definition 13** ($\text{CB}^\omega$ pre-proof). We define a $\text{CB}^\omega$ pre-proof to be a pair $(D, C)$ such that $D$ is a finite derivation tree and $C$ is a function mapping each bud to its companion. The sequent at the root node of $D$ is called the conclusion of the proof.

**Definition 14** (Tree-unfolding). Let $P$ be a $\text{CB}^\omega$ pre-proof $(D, C)$. A tree-unfolding $T(P)$ of $P$ is recursively defined by

$$T(P)(\sigma) = \begin{cases} D(\sigma), & \text{if } \sigma \in \text{dom}(D) \setminus \text{bud}(D), \\ T(P)(\sigma_1\sigma_2), & \text{if } \sigma \notin \text{dom}(D) \setminus \text{bud}(D) \text{ with } \sigma = \sigma_1\sigma_2, \sigma_1 \in \text{bud}(D) \text{ and } \sigma_3 = C(\sigma_1), \end{cases}$$

Note that a tree-unfolding is a $\text{B}^\omega$ pre-proof.

**Definition 15** ($\text{CB}^\omega$ proof). A $\text{CB}^\omega$ pre-proof $P$ of a sequent $S$ is called a $\text{CB}^\omega$ proof of $S$ if its tree-unfolding $T(P)$ satisfies the global trace condition. A cut-free $\text{CB}^\omega$ proof is a $\text{CB}^\omega$ proof that does not contain (Cut). A sequent $S$ is said to be (cut-free) provable in $\text{CB}^\omega$ if a (cut-free) $\text{CB}^\omega$ proof of $S$ exists.

A $\text{CB}^\omega$ pre-proof in which each companion is an ancestor of the corresponding bud is called cycle-normal. The following proposition says that $\text{CB}^\omega$ satisfies the cycle-normalization property.

**Proposition 16.** For a $\text{CB}^\omega$ pre-proof $P$, we have a $\text{CB}^\omega$ cycle-normal pre-proof $P'$ such that $T(P) = T(P')$.

This property was already shown by Brotherston in a general setting that includes $\text{CB}^\omega$ [2]. Besides it, a shorter proof for $\text{CLKID}^\omega$ was given in [13]. It can be applied straightforwardly to the current setting. (We give the proof in the appendix for the reviewer’s convenience.)

4. A counterexample with only unary inductive predicates to cut-elimination in $\text{CB}^\omega$

In this section, we prove the following theorem, which is the main theorem. Let $\text{TeF}$ and $\text{FsT}$ be the inductive predicates defined in Example 2.

**Theorem 17.** The following statements hold:

(1) $\text{TeF}(s) \vdash \text{FsT}(e)$ is provable in $\text{CB}^\omega$.

(2) $\text{TeF}(s) \vdash \text{FsT}(e)$ is not cut-free provable in $\text{CB}^\omega$.

This theorem means that $\text{TeF}(s) \vdash \text{FsT}(e)$ is a counterexample with only unary inductive predicates to cut-elimination in $\text{CB}^\omega$.

A $\text{CB}^\omega$ proof of $\text{TeF}(s) \vdash \text{FsT}(e)$ is the derivation tree given in Figure 2, where $(\dagger)$ indicates the pairing of the companion with the bud and the underlined formulas denotes the infinitely progressing trace for the tails of the infinite path (some applying rules and some labels of rules are omitted for limited space). Thus, Theorem 17 (1) is correct.

In this section, we henceforth prove Theorem 17 (2).

4.1. The outline of the proof of Theorem 17 (2)

Before proving the theorem, we outline our proof of Theorem 17 (2).

Assume there exists a cut-free $\text{CB}^\omega$ proof of $\text{TeF}(s) \vdash \text{FsT}(e)$. By the cycle-normalization property of $\text{CB}^\omega$, there exists a cut-free cycle-normal $\text{CB}^\omega$ proof of $\text{TeF}(s) \vdash \text{FsT}(e)$. Let $(D_{cf}, C_{cf})$ be the $\text{CB}^\omega$ proof.

The key concepts for the proof are a root-like sequent, a switching point, and an unfinished path. To define these concepts, we define the relation $\equiv_\Gamma$ for a finite set of formulas $\Gamma$ to be the smallest congruence relation on terms containing $t_1 = t_2 \in \Gamma$ (Definition 18) and the index of $\text{TeF}(t)$ in a sequent $\Gamma \vdash \Delta$ (Definition 20).
The index of $\text{TeF}(t)$ is $m - n$ if there uniquely exists $m - n$ such that $n, m \in \mathbb{N}$, and $n^m \equiv \tau \cdot n^n\cdot s$. The index of $\text{TeF}(t)$ is $\perp$ if $n^m t \not\equiv \tau \cdot n^n\cdot s$ for any $m, n \in \mathbb{N}$. The index of $\text{TeF}(t)$ may be undefined, but the index is always defined in a special sequent, called a root-like sequent (Definition 21). A switching point is defined as a node that is the conclusion of (CASE TeF) with the principal formula whose index is $\perp$ (Definition 23). An unfinished path is defined as a path $(\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}$ of $T(\mathcal{D}_{cf}, \mathcal{C}_{cf})$ such that $\Gamma_0 \vdash \Delta_0$ is a root-like sequent and $\Gamma_i \vdash \Delta_i$ is a switching point if $\Gamma_{i+1} \vdash \Delta_{i+1}$ is the left assumption of $\Gamma_i \vdash \Delta_i$ (Definition 24). Then, the following statements hold:

1. The root is a root-like sequent;
2. Every sequent in an unfinished path is a root-like sequent (Lemma 25);
3. There exists a switching point on an infinite unfinished path (Lemma 27); and
4. The rightmost path from a root-like sequent is infinite (Lemma 29).

At last, we show there exist infinite nodes in the derivation tree $\mathcal{D}_{cf}$. Because of (1) and (4), the rightmost path from the root is an infinite unfinished path. By (3), there exists a switching point on the path. Let $\bar{\sigma}_0$ be the node of the smallest height among such switching points. Let $\sigma_0$ be the left assumption of $\bar{\sigma}_0$. By (2), the sequent of $\sigma_0$ is a root-like sequent. By (4), the rightmost path from $\sigma_0$ is infinite. Therefore, there exists a bud $\mu_0$ in the rightmost path from $\sigma_0$. By (3) and the definition of $\bar{\sigma}_0$, there exists a switching point between $\sigma_0$ and $\mu_0$. Let $\bar{\sigma}_1$ be the node of the smallest height among such switching points. The nodes $\bar{\sigma}_0$ and $\bar{\sigma}_1$ are distinct by their definitions. By repeating this process as in Figure 3, we get a set of infinite nodes $\{\bar{\sigma}_i | i \in \mathbb{N}\}$. It is a contradiction since the set of nodes of $\mathcal{D}_{cf}$ is finite.
4.2. The proof of Theorem 17 (2)

We show Theorem 17 (2). Assume there exists a cut-free CBF’ proof of TeF(s) ⊢ FsT(e) for contradiction. By the cycle-normalization property of CBF’, there exists a cut-free cycle-normal CBF’ proof of TeF(s) ⊢ FsT(e).

We write (Dcf, Ccf) for a cut-free cycle-normal CBF’ proof of TeF(s) ⊢ FsT(e).

Remark. Let Γ ⊢ ∆ be a sequent in (Dcf, Ccf). By induction on the height of sequents in Dcf, we can easily show the following statements:

1. Γ consists of only atomic formulas with =, TeF.
2. ∆ consists of only atomic formulas with FsT.
3. A term in Γ and ∆ is of the form n^a, n^a e, or n^a x with some variable x.
4. The possible rules in (Dcf, Ccf) are (WEAK), (SUBST), (= L’), (FsT R1), (FsT R2), and (CASE TeF) (See Example 5).

Without loss of generality, we assume terms in this section are of the form n^a, n^a e, or n^a x with some variable x.

Definition 18 ( ⪰Γ ). For a set of formulas Γ, we define the relation ⪰Γ to be the smallest congruence relation on terms which satisfies the condition that t_1 = t_2 ∈ Γ implies t_1 ⪰Γ t_2.

Intuitively, ⪰Γ represents the equal in any models of Γ.

Definition 19 ( ~Γ ). For a set of formulas Γ and terms t_1, t_2, we define t_1 ~Γ t_2 by n^a t_1 ⪰Γ n^a t_2 for some n, m ∈ N.

Note that ~Γ is a congruence relation.

Definition 20 (Index). For a finite set Γ and TeF(t) ∈ Γ, we define the index of TeF(t) in Γ as follows:

1. If t ≠ Γ s, then the index of TeF(t) in Γ is ⊥, and
2. if there uniquely exists m – n such that n, m ∈ N, and n^a t ⪰Γ n^a s, then the index of TeF(t) in Γ is m – n (namely the uniqueness means that n^a t ⪰Γ n^a s for n, m ∈ N implies m – n = m’ – n’).

Note that if there exists n_0, m_0, n_1, m_1 ∈ N such that n^a t ⪰Γ n^a s, n^a t ⪰Γ n^a s and m_0 – n_0 ≠ m_1 – n_1, then the index of TeF(t) in Γ is undefined.

Definition 21 (Root-like sequent). The sequent Γ ⊢ ∆ is said to be a root-like sequent if the following conditions hold:

1. if s ≠ Γ e,
2. t ≠ Γ s for any FsT(t) ∈ ∆, and
3. if n^a s ⪰Γ n^a s, then n = m.

A root-like sequent does not occur as a conclusion of (FsT R1) by the first and second conditions. The third condition guarantees the existence of an index, as shown in the following lemma.

Lemma 22. If Γ ⊢ ∆ is a root-like sequent, the index of any TeF(t) in Γ is defined.

Proof. Let TeF(t) ∈ Γ. If t ≠ Γ s, then the index is ⊥.

Assume t ~Γ s. By Definition 19, there exist n_0 and m_0 such that n^a t ⪰Γ n^a s. To show the uniqueness, assume n^a t ⪰Γ n^a s for n_1 and m_1. Since n^a t ⪰Γ n^a t and n^a n_0 + m_0 ⪰Γ n^a n_1 + m_0, we have n^a n_0 + m_0 ⪰Γ n^a n_1 + m_0. From (3) of Definition 21, m_0 + n_1 = m_1 + n_0. Thus, m_0 – n_0 = m_1 – n_1.

Definition 23 (Switching point). A node σ in a derivation tree is called a switching point if the rule with the conclusion σ is (CASE TeF) and the index of the principal formula for the rule in the conclusion is ⊥.

We call the assumption of (CASE TeF) whose form is Γ, t = x, TeF(nx) ⊢ ∆ the right assumption of the rule. The other assumption is called the left assumption of the rule.
Definition 24 (Unfinished path). A path \((\Gamma, \Delta)_{0 \leq i < \alpha}\) in \(T(D_{cf}, C_{cf})\) with some \(\alpha \in \mathbb{N} \cup \{\omega\}\) is said to be an unfinished path if the following conditions hold:

1. \(\Gamma_0 \vdash \Delta_0\) is a root-like sequent, and
2. if the rule for \(\Gamma_i \vdash \Delta_i\) is (CASE TeF) and \(\Gamma_{i+1} \vdash \Delta_{i+1}\) is the left assumption of the rule, then \(\Gamma_i \vdash \Delta_i\) is a switching point.

Lemma 25. Every sequent in an unfinished path is a root-like sequent.

Sketch of proof. Let \((\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}\) be an unfinished path. We prove the statement by the induction on \(i\).

For \(i = 0\), \(\Gamma_0 \vdash \Delta_0\) is a root-like sequent by Definition 24.

For \(i > 0\), we can prove the statement by considering cases according to the rule with the conclusion \(\Gamma_{i-1} \vdash \Delta_{i-1}\). For more details, see Appendix B.2. \(\square\)

Lemma 26. For an unfinished path \((\Gamma_i \vdash \Delta_i)_{0 \leq i < \alpha}\) and a trace \((\tau_k)_{k \geq 0}\) following \((\Gamma_i \vdash \Delta_i)_{i \geq p}\), if \(\kappa_k\) is the index of \(\tau_k\), the following statements holds:

1. If \(\kappa_k = 0\), then \(\kappa_{k+1} = 1\).
2. If the rule with the conclusion \(\Gamma_{p+k} \vdash \Delta_{p+k}\) is (Weak) or (Subst), then \(\kappa_{k+1} = \kappa_k\) or \(\kappa_{k+1} = 0\).
3. If the rule with the conclusion \(\Gamma_{p+k} \vdash \Delta_{p+k}\) is \(= L'\) or (FsT R2), then \(\kappa_{k+1} = \kappa_k\).
4. Assume the rule with the conclusion \(\Gamma_{p+k} \vdash \Delta_{p+k}\) is (CASE TeF).
   a. If \(\Gamma_{p+k+1} \vdash \Delta_{p+k+1}\) is the left assumption of the rule, then \(\kappa_{k+1} = \kappa_k\).
   b. If \(\Gamma_{p+k+1} \vdash \Delta_{p+k+1}\) is the right assumption of the rule and \(\kappa_k\) is not a progress point of the trace, then \(\kappa_{k+1} = \kappa_k\).
   c. If \(\Gamma_{p+k+1} \vdash \Delta_{p+k+1}\) is the right assumption of the rule and \(\kappa_k\) is a progress point of the trace, then \(\kappa_{k+1} = \kappa_k + 1\).

Sketch of proof. Let \(\tau_k \in \text{TeF}(\kappa_k)\).

1. It suffices to show that \(\tau_{k+1} \not\models \Gamma_{p+k+1}\) if \(\tau_k \not\models \Gamma_{p+k}\). We can prove it by considering cases according to the rule with the conclusion \(\Gamma_{p+k} \vdash \Delta_{p+k}\). For more details, see Appendix B.3.

2. (3) (4) Straightforward. For more details, see Appendix B.3. \(\square\)

Lemma 27. For an infinite unfinished path \((\Gamma_i \vdash \Delta_i)_{i \geq 0}\) in \(T(D_{cf}, C_{cf})\), there exists \(l \in \mathbb{N}\) such that the following conditions hold:

1. \(\Gamma_l \vdash \Delta_l\) is a switching point in \(T(D_{cf}, C_{cf})\), and
2. \(\Gamma_{l+1} \vdash \Delta_{l+1}\) is the right assumption of the rule with the conclusion \(\Gamma_l \vdash \Delta_l\).

Proof. Since \((\Gamma_l \vdash \Delta_l)_{l \geq 0}\) is an infinite path and \(T(D_{cf}, C_{cf})\) satisfies the global trace condition, there exists an infinitely progressing trace following a tail of the path. Let \((\tau_k)_{k \geq 0}\) be an infinitely progressing trace following \((\Gamma_l \vdash \Delta_l)_{l \geq p}\). Let \(d_k\) be the index of \(\tau_k\) in \(\Gamma_{p+k}\).

We show that there exists \(l \in \mathbb{N}\) such that \(d_l = \perp\). The set \(\{d_k | k \geq 0\}\) is finite since the set of sequents in \((\Gamma_l \vdash \Delta_l)_{l \geq 0}\) is finite and we have a unique index of an atomic formula with TeF in \(\Gamma_l \vdash \Delta_l\). Since \((\tau_k)_{k \geq 0}\) is an infinitely progressing trace following \((\Gamma_l \vdash \Delta_l)_{l \geq p}\), if there does not exist \(k' \in \mathbb{N}\) such that \(d_{k'} = \perp\), Lemma 26 implies that \(\{d_k | k \geq 0\}\) is infinite. Thus, there exists \(k' \in \mathbb{N}\) such that \(d_{k'} = \perp\).

Since \((\tau_k)_{k \geq 0}\) is an infinitely progressing trace following \((\Gamma_l \vdash \Delta_l)_{l \geq p}\), there exists a progress point \(\tau_l\) with \(l > k'\). By Lemma 26, \(d_l = \perp\). Since \(\tau_k\) is a progress point, \(\Gamma_{p+k} \vdash \Delta_{p+k}\) is a switching point and \(\Gamma_{p+k+1} \vdash \Delta_{p+k+1}\) is the right assumption of the rule. \(\square\)

Definition 28 (Rightmost path). For a derivation tree \(D\) and a node \(\sigma\) in \(D\), we define the rightmost path from the node \(\sigma\) as the path \((\sigma_i)_{0 \leq i < \alpha}\) satisfying the following conditions:

1. The node \(\sigma_0\) is \(\sigma\).
2. If \(\sigma_i\) is the conclusion of (CASE TeF), the node \(\sigma_{i+1}\) is the right assumption of the rule.
(3) If $\sigma_i$ is the conclusion of the rules (Weak), (Subst), ($=^L$), or ($\text{FsT R}_2$), the node $\sigma_{i+1}$ is the assumption of the rule.

**Lemma 29.** The rightmost path from a root-like sequent in $T(D_{\text{cf}}, C_{\text{cf}})$ is infinite.

**Proof.** By Definition 24, the rightmost path from a root-like sequent in $T(D_{\text{cf}}, C_{\text{cf}})$ is an unfinished path. By Lemma 25, every sequent on the path is a root-like sequent. By Definition 21, ($\text{FsT R}_1$) does not occur in the path. Thus, the path is infinite. $\square$

**Remark.** For an infinite path in $T(D_{\text{cf}}, C_{\text{cf}})$, the corresponding path in $D_{\text{cf}}$ has a bud.

We now have the lemmas to prove Theorem 17 (2).

**Proof of Theorem 17 (2).** We show that there exists a sequence $(\bar{\sigma}_i)_{i \in \mathbb{N}}$ of switching points in $D_{\text{cf}}$ which satisfies the following conditions:

(i) The height of $\bar{\sigma}_i$ is greater than the height of $\bar{\sigma}_{i-1}$ in $D_{\text{cf}}$ for $i > 0$.

(ii) For any node $\sigma$ on the path from the root to $\bar{\sigma}_i$ in $D_{\text{cf}}$ excluding $\bar{\sigma}_i$, $\sigma$ is a switching point if and only if the child of $\sigma$ on the path is the left assumption of the rule (CASE $\text{TeF}$).

We construct $(\bar{\sigma}_i)_{i \in \mathbb{N}}$ and show (i) and (ii) by induction on $i$.

We consider the case $i = 0$.

The rightmost path in $T(D_{\text{cf}}, C_{\text{cf}})$ from the root is an infinite unfinished path since $\text{TeF}(s) \vdash \text{FsT}(e)$ is a root-like sequent and there exists no node which is the left assumption of (CASE $\text{TeF}$) on the path. By Lemma 27, there exists a switching point on the path. Hence, there exists a switching point on the rightmost path from the root in $D_{\text{cf}}$. Let $\bar{\sigma}_0$ be the switching point of the smallest height among such switching points. (i) and (ii) follow immediately for $\bar{\sigma}_0$.

We consider the case $i > 0$.

Let $\alpha$ be the left assumption of the rule with the conclusion $\bar{\sigma}_{i-1}$. Because of (ii), the path from the root to $\bar{\sigma}_{i-1}$ is also an unfinished path. Since $\bar{\sigma}_{i-1}$ is a switching point, the path from the root to $\alpha$ is also an unfinished path. By Lemma 25, $\alpha$ is a root-like sequent. By Lemma 29, the rightmost path from $\alpha$ in $T(D_{\text{cf}}, C_{\text{cf}})$ is infinite. Therefore, there is a bud on the rightmost path in $D_{\text{cf}}$ from $\alpha$. Let $\mu$ be the bud.

Let $\pi_1$ be the path from the root to $\mu$ in $D_{\text{cf}}$ and $\pi_2$ be the path from $C_{\text{cf}}(\mu)$ to $\mu$ in $D_{\text{cf}}$. We define the path $\pi$ in $T(D_{\text{cf}}, C_{\text{cf}})$ as $\pi_1 \pi_2^{\omega}$. Let $(\sigma_i)_{0 \leq i}$ be $\pi$. Because of (ii), $\pi$ is an unfinished path. By Lemma 27, there is a switching point $\sigma_1$ and $\sigma_{i+1}$ is the right assumption of the rule. Hence, there is a switching point on $\pi_1 \pi_2$ in $D_{\text{cf}}$ such that its child on $\pi_1 \pi_2$ is the right assumption of the rule. Define $\bar{\sigma}_i$ as the switching point of the smallest height among such switching points.

We show $\bar{\sigma}_i$ satisfies the conditions (i) and (ii).

(i) By the definition of $\bar{\sigma}_i$, $\bar{\sigma}_i$ is on $\pi_1$. By the condition (ii), $\bar{\sigma}_i$ is not on the path from the root to $\bar{\sigma}_{i-1}$. Hence, the height of $\bar{\sigma}_i$ is greater than that of $\bar{\sigma}_{i-1}$.

(ii) Let $\sigma$ be a node on the path from the root to $\bar{\sigma}_i$ excluding $\bar{\sigma}_i$. We can assume $\sigma$ is on the path from $\bar{\sigma}_{i-1}$ to $\bar{\sigma}_i$ excluding $\bar{\sigma}_i$ by the induction hypothesis.

The “only if” part: Assume that $\sigma$ is a switching point. By the definition of $\bar{\sigma}_i$, we see that $\sigma$ is $\bar{\sigma}_{i-1}$. The child of $\bar{\sigma}_{i-1}$ on the path from the root to $\bar{\sigma}_i$ is $\alpha$, which is the left assumption of the rule.

The “if” part: Assume that the child of $\sigma$ on the path is the left assumption of the rule. Since there is not the left assumption of a rule on the path from $\alpha$ to $\bar{\sigma}_i$, we see that $\sigma$ is $\bar{\sigma}_{i-1}$. Thus, $\sigma$ is a switching point.

We complete the construction and the proof of the properties.

Because of (i), $\bar{\sigma}_0, \bar{\sigma}_1, \ldots$ are all distinct in $D_{\text{cf}}$. Thus, $\{\bar{\sigma}_i | i \in \mathbb{N}\}$ is infinite. It is a contradiction since the set of nodes in $D_{\text{cf}}$ is finite. $\square$

### 4.3. Corollaries of Theorem 17

By Theorem 17, we have some corollaries. We write $\text{CLKID}^\omega$ for the cyclic proof system for first-order logic with inductive definitions proposed by Brotherston and Simpson [4].

**Corollary 30.** The following statements hold:

(i) $\text{TeF}(s) \vdash \text{FsT}(e)$ is provable in $\text{CLKID}^\omega$.

(ii) $\text{TeF}(s) \vdash \text{FsT}(e)$ is not cut-free provable in $\text{CLKID}^\omega$. 

$\square$
No 

Assume there exists a

The derivation tree in Figure

Companion in the right-most path if there exists an infinitely progressing trace. However, since there is the left-contraction rule. The proof technique in [13] and this paper is more complicated than in [11] since there is the left-contraction rule. The proof technique in [11] is to show that there does not exist a companion in the right-most path if there exists an infinitely progressing trace. However, since there is the left-contraction rule, we have an infinitely progressing trace in the right-most path, as in Figure 4.

Figure 4: There can be an infinitely progressing trace in the rightmost path

| nullary predicates | unary predicates | binary predicates | N-ary (N ≥ 3) predicates |
|-------------------|-----------------|-----------------|--------------------------|
| Classical Logic   | ? (perhaps Yes) | No' (This paper)| No' (This paper)         |
| Separation Logic  | No [15]         | No' (by [15])   | No' [13]                 |
| Bunched Logic     | No [15]         | No [15]         | No [15]                  |

Proof. (i) The derivation tree in Figure 2 is also a CLKIDω proof of the sequent. Thus, we have the statement.

(ii) Assume there exists a CLKIDω cut-free proof of Tef(s) ⊢ Fst(e). We write CLKIDω for the cyclic proof system obtained by replacing (= L) with (= L') in CLKIDω. Since (= L) is derivable in CLKIDω, there exists a CLKIDω cut-free proof of Tef(s) ⊢ Fst(e). Let (Δcf, Ccf) be such a proof. Since the rules which can occur in (Δcf, Ccf) are (Weak), (Subst), (= L'), (Fst R1), (Fst R2), and (Case Tef), we understand (Δcf, Ccf) as a cut-free CBω proof of Tef(s) ⊢ Fst(e). It contradicts Theorem 17 (2).

Corollary 30 means that Tef(s) ⊢ Fst(e) is a counterexample to cut-elimination in CLKIDω, and therefore we have the following corollary.

Corollary 31. We do not eliminate the cut rule in CLKIDω if we restrict predicates in the language to unary predicates.

5. Conclusions and discussion

We have shown that Tef(s) ⊢ Fst(e) is a counterexample with only unary inductive predicates to cut-elimination in CBω. This counterexample implies that we cannot eliminate the cut rule in first-order logic with inductive definitions if we restrict predicates in the language to unary predicates and =.

The proofs for counterexamples to cut-elimination in [13] and this paper is more complicated than in [11] since there is the left-contraction rule. The proof technique in [11] is to show that there does not exist a companion in the right-most path if there exists an infinitely progressing trace. However, since there is the left-contraction rule, we have an infinitely progressing trace in the right-most path, as in Figure 4. Comparing our proof technique with the proof technique in [15] is reserved for future work.

Table 1 shows the results we obtained about the cut-elimination property of each cyclic proof system for some logics. “No” means that the cut-elimination property does not hold. “No’” means that the cut-elimination property does not hold if there are constants and a unary function symbol. The second and third column results in the “Separation Logic” row are easily obtained from Saotome’s result [15] because the counterexample for the cyclic proof system of bunched logic also works for separation logic.

Why does not the cut-elimination property hold in cyclic proof systems? The reason is not yet wholly understood, but we discuss it briefly. The proofs for counterexamples to cut-elimination in [11, 15, 13] and this paper have one thing in common. It is to contradict the finiteness of the sequent occurring in the cut-free proof of each counterexample if it exists. The more important fact is that the cut-elimination property of LKIDω, which is obtained by expanding the shape of each proof figure into an infinitary tree, holds. These facts suggest that the reason the cut-elimination property does not hold in cyclic proof systems is the finiteness of occurring sequents in each proof.

Now, we discuss the “Classical Logic” row in Table 1. It suggests the reason the cut-elimination property does not hold might be a unary function symbol in the language. Then, we conjecture that the cut-elimination
property of $\text{CLKID}^\omega$ hold if there is no unary function symbol, and therefore the cut-elimination property of $\text{CLKID}^\omega$ hold if restricting the arity of predicates to nullary.

By the way, can we restrict cut formulas in $\text{CLKID}^\omega$ without changing provability? Saotome et al. [14] suggest that we cannot restrict the cut formulas to formulas presumable from the goal sequent in the cyclic proof system for symbolic-heaps, a fragment of separation logic. The cut formulas in Figure 2 are presumable. Also, the cut formula in a $\text{CLKID}^\omega$ proof of the counterexample in [13] is presumable. Can we restrict the cut formulas to presumable formulas? If the answer to the problem is “Yes”, there may be an efficient proof search in $\text{CLKID}^\omega$. Research into solving the problem is in progress.

**Acknowledgments**

We are grateful to Koji Nakazawa and Kenji Saotome for giving us helpful suggestions from the early stage of this work. We would also like to thank Makoto Tatsuta for giving us valuable comments. This work was supported by the Research Institute for Mathematical Sciences, an International Joint Usage/Research Center located in Kyoto University.

**References**

[1] D. Baelde and A. Doumane and A. Saurin. *Infinitary proof theory: the multiplicative additive case*. Proceedings of the 25th EACSL Annual Conference on Computer Science Logic (CSL 2016), LIPIcs vol. 62, pp. 42:1–42:17, 2016.

[2] J. Brotherston. *Sequent Calculus Proof Systems for Inductive Definitions*. PhD thesis, University of Edinburgh, 2006.

[3] J. Brotherston. *Formalised Inductive Reasoning in the Logic of Bunched Implications*. Proceedings of Static Analysis Symposium (SAS’14), LNCS vol. 4634, pp. 87–103, 2007.

[4] J. Brotherston and A. Simpson. *Sequent calculi for induction and infinite descent*. Journal of Logic and Computation, 21(6):1177–1216, 2011.

[5] J. Brotherston, D. Distefano, and R. L. Petersen. *Automated Cyclic Entailment Proofs in Separation Logic*. Proceedings of the 23rd International Conference on Automated Deduction (CADE-23), LNCS vol. 6803, pp. 131–146, 2011.

[6] J. Brotherston, N. Gorogiannis, and R. L. Petersen. *A Generic Cyclic Theorem Prover*. Proceedings of the 10th Asian Symposium on Programming Languages and Systems (APLAS 2012), LNCS vol. 7705, pp. 350–367, 2012.

[7] D. Chu, J. Jaffar, and M. Trinh. *Automatic Induction Proofs of Data-Structures in Imperative Programs*. Proceedings of the 36th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI’15), pp. 457–466, 2015.

[8] The Cyclist Framework and Provers. [http://www.cyclist-prover.org/](http://www.cyclist-prover.org/)

[9] A. Doumane. *On the infinitary proof theory of logics with fixed points*. PhD thesis, University of Paris 7, 2017.

[10] J. Fortier and L. Santocanale. *Cuts for circular proofs: semantics and cut-elimination*. Proceedings of Computer Science Logic (CSL 2013), LIPIcs vol. 23, pp. 248–262, 2013.

[11] D. Kimura, K. Nakazawa, T. Terauchi, and H. Unno. *Failure of cut-elimination in cyclic proofs of separation logic*. Computer Software, 37(1):39–52, 2020.

[12] P. Martin Lof. *Haupstatz for the intuitionistic theory of iterated inductive definitions*. Proceedings of the Second Scandinavian Logic Symposium, Studies in Logic and the Foundations of Mathematics, vol. 63, pp. 179–216, 1971.

[13] Y. Masuoka and M. Tatsuta. *Counterexample to cut-elimination in cyclic proof system for first-order logic with inductive definitions*. arXiv: [https://arxiv.org/abs/2106.11798](https://arxiv.org/abs/2106.11798), 2021.
[14] K. Saotome, K. Nakazawa, and D. Kimura. *Restriction on Cut in Cyclic Proof System for Symbolic Heaps*. Proceedings of Functional and Logic Programming (FLOPS2020), pp. 88–105, 2020.

[15] K. Saotome, K. Nakazawa, and D. Kimura. *Failure of cut-elimination in the cyclic proof system of bunched logic with inductive propositions*. Proceedings of the 6th International Conference on Formal Structures for Computation and Deduction (FSCD2021), LIPIcs vol. 195, pp. 11:1–11:14, 2021.

[16] Q. T. Ta, T. C. Le, S. C. Khoo, and W. -N. Chin. *Automated Mutual Explicit Induction Proof in Separation Logic*. Proceedings of the 21st International Symposium of Formal Methods (FM ’16), LNCS vol. 9995, pp. 659–676, 2016.

[17] Q. T. Ta, T. C. Le, S. C. Khoo, and W. -N. Chin. *Automated lemma synthesis in symbolic-heap separation logic*. Proceedings of the 45th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL ’18), Vol. 2, Article No.9, 2018.

[18] M. Tatsuta, K. Nakazawa, and D. Kimura. *Completeness of cyclic proofs for symbolic heaps with inductive definitions*. Proceedings of the 17th Asian Symposium on Programming Languages and Systems (APLAS 2019), LNCS vol. 11893, pp. 367–387, 2019.

[19] A. Tiu and A. Momigliano. *Cut elimination for a logic with induction and co-induction*. Journal of Applied Logic, vol. 10 (4) pp. 330–367, 2012.

[20] G. Tellez and J. Brotherston. *Automatically Verifying Temporal Properties of Programs with Cyclic Proof*. Journal of Automated Reasoning, vol. 64(3), pp. 555–578, 2020.

[21] T. Tsukada and H. Unno. *Software Model-Checking as Cyclic-Proof Search*. arXiv: https://arxiv.org/abs/2111.05617 (to appear in POPL 2022)
A. The proof of the cycle normalization of $\mathcal{CB}^{\omega}$ (Proposition 16)

Proof of Proposition 16. Let $(D, C)$ be $P$, and $D_1$ be the tree unfolding of $P$. We write $\sigma \preceq \sigma'$ when $\sigma$ is an initial segment of $\sigma'$. If $\sigma$ is a strict initial segment of $\sigma'$, we write $\sigma \prec \sigma'$. We define $D(\sigma)$ by $D(\sigma)(1) = D(\sigma_1)$, $S_1$ as $\{\sigma' \mid \sigma' \preceq \sigma \in S\}$, and $S_2$ as $\{\sigma' \mid \sigma' \prec \sigma \in S\}$.

Then define $S_1$ and $S_2$ by:

$$S_1 = \{\sigma \in \text{dom}(D_1) \mid \exists \sigma' \prec \sigma (D(\sigma') = D(\sigma_1)), \forall \sigma_1, \sigma_2 \prec \sigma (D(\sigma_1) \neq D(\sigma_2)), \forall n \exists \sigma_1 \preceq \sigma (\sigma_1 \in \text{dom}(D_1), |\sigma_1| \geq n)\},$$

$$S_2 = \{\sigma \in \text{dom}(D_1) \mid \sigma \neq 0 \neq \text{dom}(D_1), \forall \sigma' \preceq \sigma (\sigma' \in S_1)\}.$$

$S_1$ is the set of nodes such that the node is on some infinite path and the node is of the smallest height on the path among nodes, each of which has some inner node of the same subtree. $S_2$ is the set of leaf nodes of finite paths which are not cut by $S_1$.

Define $D'$ by

$$D'(\sigma) = \begin{cases} D_1(\sigma) & \text{if } \sigma \in (S_1)^* \cup S_2, \\ (\Gamma \vdash \Delta, \text{(BUD)}) & \text{if } \sigma \in S_1 \text{ and } D_1(\sigma) = (\Gamma \vdash \Delta, R). \end{cases}$$

Define $C'$ by $C'(\sigma) = \sigma$ for $\sigma \in \text{bud}(D)$ where $\sigma' \prec \sigma$ and $D(\sigma') = D(\sigma)$. We can show that $\text{dom}(D')$ is finite as follows. Since $\text{dom}(D') = S_1 \cup S_2$, we have $\text{dom}(D') \subseteq \text{dom}(D_1)$. Since $D_1$ is finite-branching, $D'$ is so. Assume $\text{dom}(D')$ is infinite to show contradiction. By König’s lemma, there is some infinite path $(\sigma_i)$ such that $\sigma_i \in \text{dom}(D')$. Since $D_1$ is regular, the set $\{D(\sigma)\}$ is finite. Hence there are $j < k$ such that $D(\sigma_j) = D(\sigma_k)$. Take the smallest $k$ among such $k$’s. Then $\sigma_k \in S_1$. Hence $\sigma_{k+1} \notin S_1$. Hence $\sigma_{k+1} \notin \text{dom}(D')$, which contradicts.

Then $(D', C')$ is a CLKID$^\omega$ cycle-normal pre-proof.

Define $D'_1$ as the tree-unfolding of $(D', C')$.

We can show $D_1 = D'_1$ on $\text{dom}(D'_1)$ as follows.

Case 1 where for any $\sigma' \preceq \sigma, \sigma' \notin S_1$. $D'_1(\sigma) = D'(\sigma) = D_1(\sigma)$.

Case 2 where there is some $\sigma_1 \preceq \sigma$ such that $\sigma_1 \in S_1$. Let $\sigma_1 \sigma_2$ be $\sigma$ and $\sigma_3$ be $C'(\sigma_1)$. Then $D_1(\sigma) = D'(\sigma_1) = D'(\sigma_2) = D'(\sigma_3) = D'_1(\sigma_3 \sigma_2) = D'_1(\sigma_3 \sigma_2)$ by the induction hypothesis, it is $D'_1(\sigma_1 \sigma_2)$ by definition of $D'_1$, and it is $D'_1(\sigma)$.

We show $\text{dom}(D_1) \subseteq \text{dom}(D'_1)$ as follows. By induction on $|\sigma|$, we will show $\sigma \in \text{dom}(D_1)$ implies $\sigma \in \text{dom}(D'_1)$. If $\sigma \in S_1 \cup S_2$, then $\sigma \in \text{dom}(D'') - S_1$. Hence $\sigma \in \text{dom}(D'_1)$. If there is some $\sigma_1 \prec \sigma$ such that $\sigma_1 \in S_1$, then by letting $\sigma = \sigma_1 \sigma_2$ and $\sigma_3 = C'(\sigma_1)$, $D_1(\sigma) = D_1(\sigma_3 \sigma_2)$ by definition of $C'$, by the induction hypothesis for $\sigma_3 \sigma_2$ it is $D'_1(\sigma_3 \sigma_2)$, and it is $D'_1(\sigma)$ by definition of $D'_1$. Thus we have shown $\text{dom}(D_1) \subseteq \text{dom}(D'_1)$ holds. Hence $D_1 = D'_1$ holds.

B. The proofs of Lemma 25 and Lemma 26

In this appendix, we show Lemma 25 and Lemma 26. We assume terms in this appendix are of the form $n^x$, $nP$, or $nP x$ with some variable $x$. For terms $t_1$ and $t_2$, we write $t_1 \equiv t_2$ if $t_1$ is the same as $t_2$. For sets of formulas $\Gamma_1$ and $\Gamma_2$, we write $\Gamma_1 \equiv \Gamma_2$ if $\Gamma_1$ is the same as $\Gamma_2$.

B.1. The lemmas for Lemma 25 and Lemma 26

We show the lemmas for Lemma 25 and Lemma 26.

Lemma 32. Let $\Gamma$ be a set of formulas and $\theta$ be a substitution.

(i) For any terms $t_1$ and $t_2$, $t_1[\theta] \equiv t_2[\theta]$ if $t_1 \equiv t_2$.

(ii) For any terms $t_1$ and $t_2$, $t_1 \not\equiv t_2$ if $t_1[\theta] \not\equiv t_2[\theta]$.

Proof. (i) We prove the statement by induction on the definition of $\equiv$. We only show the base case. Assume $t_1 = t_2 \in \Gamma$. Then, $t_1[\theta] = t_2[\theta] \in [\theta]$. Thus, $t_1[\theta] \equiv t_2[\theta]$.

(ii) By Definition 19 and (i), we have the statement.

Lemma 33. Let $\Gamma$ be a set of formulas, $u_1, u_2$ be terms, $v_1, v_2$ be variables, $\Gamma_1 \equiv (\Gamma|v_1 := u_1, v_2 := u_2), u_1 = u_2$, and $\Gamma_2 \equiv (\Gamma|v_1 := u_2, v_2 := u_1), u_1 = u_2$.
(i) For any terms $t_1$ and $t_2$, $\Gamma[t_1 := u_1, v_2 := u_2] \not\models_{\Gamma_2} \Gamma[t_2 := v_1, u_2 := u_1]$. If $t_1[v_1 := u_1, v_2 := u_2] \not\models_{\Gamma_1} t_2[v_2 := v_1, u_2 := u_1]$, if $t_1[v_1 := u_1, v_2 := u_2] \not\models_{\Gamma_1} t_2[v_2 := v_1, u_2 := u_1]$.

(ii) For any terms $t_1$ and $t_2$, $\Gamma[t_1 := u_1, v_2 := u_1] \not\models_{\Gamma_2} \Gamma[t_2 := v_1, u_2 := u_1]$. If $t_1[v_1 := u_1, v_2 := u_2] \not\models_{\Gamma_1} t_2[v_2 := v_1, u_2 := u_1]$, if $t_1[v_1 := u_1, v_2 := u_2] \not\models_{\Gamma_1} t_2[v_2 := v_1, u_2 := u_1]$.

**Proof.** (i) We prove the statement by induction on the definition of $\models_{\Gamma_2}$. We only show the base case. Assume $t_1[v_1 := u_1, v_2 := u_1] \not\models_{\Gamma_2} \Gamma[t_2 := v_1, u_2 := u_1]$ to show $t_1[v_1 := u_1, v_2 := u_2] \not\models_{\Gamma_1} t_2[v_2 := v_1, u_2 := u_2]$. If $t_1[v_1 := u_1, v_2 := u_1] = t_2[v_1 := u_1, v_2 := u_1]$ is $u_1 = u_2$, then $t_1 = t_2$ is $v_1 = v_2$, $v_2 = u_2$, $u_1 = v_1$, or $u_1 = u_2$. Therefore, $t_1[v_1 := u_1, v_2 := u_2] \models_{\Gamma_1} t_2[v_1 := u_1, v_2 := u_2]$.

Assume $t_1[v_1 := u_2, v_2 := u_1] = t_2[v_1 := u_2, v_2 := u_1]$ is not $u_1 = u_2$. By case analysis, we have $t_1 = t_2 \in \Gamma$. Hence, $t_1[v_1 := u_1, v_2 := u_2] = t_2[v_1 := u_1, v_2 := u_2] \in \Gamma_1$. Therefore, we have $t_1[v_1 := u_1, v_2 := u_2] \models_{\Gamma_1} t_2[v_1 := u_1, v_2 := u_2]$.

(ii) By Definition 19 and (i), we have the statement.

**Lemma 34.** For a set of formulas $\Gamma$, the following statements are equivalent:

(i) $u_1 \models_{\Gamma} u_2$.

(ii) There exists a finite sequence of terms $(t_i)_{0 \leq i \leq n}$ with $n \geq 0$ such that $t_0 \equiv u_1$, $t_n \equiv u_2$ and $t_i = t_{i+1} \in [\Gamma]$ for $0 \leq i < n$, where $[\Gamma] = \{n^it_1 = n^it_2 | n \in N \text{ and either } t_1 = t_2 \in \Gamma \text{ or } t_2 = t_1 \in \Gamma\}$.

**Proof.** (i) $\Rightarrow$ (ii): Assume $u_1 \models_{\Gamma} u_2$ to prove (ii) by induction on the definition of $\models_{\Gamma}$. We consider cases according to the clauses of the definition.

Case 1. If $u_1 = u_2 \in \Gamma$, then we have $u_1 = u_2 \in [\Gamma]$. Thus, we have (ii).

Case 2. If $u_1 \equiv u_2$, then we have (ii).

Case 3. We consider the case where $u_2 \models_{\Gamma} u_1$. By the induction hypothesis, there exists a finite sequence of terms $(t_i)_{0 \leq i \leq n}$ such that $t_0 \equiv u_2$, $t_n \equiv u_1$ and $t_i \equiv t_{i+1} \in [\Gamma]$ with $0 \leq i < n$. Let $t'_n \equiv t_n-1$. The finite sequence of terms $(t'_i)_{0 \leq i \leq n}$ satisfies $t'_0 \equiv u_1$, $t'_n \equiv u_2$ and $t'_i \equiv t'_{i+1} \in [\Gamma]$. Thus, we have (ii).

Case 4. The case where $u_2 \models_{\Gamma} u_3, u_3 \models_{\Gamma} u_2$. By the induction hypothesis, there exist two finite sequences of terms $(t_i)_{0 \leq i \leq n}$, $(t'_i)_{0 \leq i \leq m}$ such that $t_0 \equiv u_1, t_n \equiv t'_0 \equiv u_3, t'_n \equiv u_2, t_i \equiv t_{i+1} \in [\Gamma]$ and $t'_j = t'_{j+1} \in [\Gamma]$ with $0 \leq i < n, 0 \leq j < m$. Define $\hat{t}_k$ as $t_k$ if $0 \leq k < n$ and $t'_{k-n}$ if $n \leq k \leq n + m$. The finite sequence of terms $(\hat{t}_i)_{0 \leq i \leq n}$ satisfies $\hat{t}_0 \equiv u_1$, $\hat{t}_n \equiv u_2$ and $\hat{t}_k \equiv \hat{t}_{k+1} \in [\Gamma]$. Thus, we have (ii).

Case 5. We consider the case where $u_1 \models_{\Gamma} u_2$. $u_1 \equiv u[v := \hat{u}_1]$ and $u_2 \equiv u[v := \hat{u}_2]$. By the induction hypothesis, there exists a finite sequence of terms $(t_i)_{0 \leq i \leq n}$ with $n \in N$ such that $t_0 \equiv u_1, t_n \equiv u_2, t_i \equiv t_{i+1} \in [\Gamma]$ with $0 \leq i < n$. Assume $v$ does not occur in $u$. In this case, we have $u_1 \equiv u[v := \hat{u}_1] \equiv u \equiv u[v := \hat{u}_2] \equiv u_2$. Hence, (ii) holds.

Assume $v$ occurs in $u$. In this case, we have $u \equiv n^mv$ for some natural numbers $m$. Let $t'_i = n^mt_i$ for $0 \leq i \leq n$. The finite sequence of terms $(t'_i)_{0 \leq i \leq n}$ satisfies $t'_0 \equiv u_1, t'_n \equiv u_2$ and $t'_i \equiv t'_{i+1} \in [\Gamma]$. (ii) $\Rightarrow$ (i): Assume (ii) to show (i). By the assumption, there exists a finite sequence of terms $(t_i)_{0 \leq i \leq n}$ with $n \in N$ such that $t_0 \equiv u_1, t_n \equiv u_2$ and $t_i \equiv t_{i+1} \in [\Gamma]$ with $0 \leq i < n$. If $t_i = t_{i+1} \in [\Gamma]$, then $t_i = t_{i+1}$ is $n^it_1 = n^it_2$, where $t_1 = t_2 \in \Gamma$ or $t_2 = t_1 \in \Gamma$. Therefore, $t_i \models_{\Gamma} t_{i+1}$. Because of the transitivity of $\models_{\Gamma}$, we have $u_1 \models_{\Gamma} u_2$.

For a term $t$, we define $\text{Var}(t)$ as a variable or a constant in $t$.

**Lemma 35.** For a set of formulas $\Gamma_1$ and $\Gamma_2 \equiv (\Gamma_1, u = u')$, if $\text{Var}(u')$ do not occur in $\Gamma_1, u, t, t'$, then $t \models_{\Gamma_2} t'$ implies $t \models_{\Gamma_1} t'$.

**Proof.** Assume $t \models_{\Gamma_2} t'$ and $t \not\models_{\Gamma_1} \text{Var}(u')$. By Lemma 34, there exists a sequence $(t_j)_{0 \leq j \leq n}$ with $n \in N$ such that $t_0 \equiv t, t_n \equiv t'$ and $t_j = t_{j+1} \in [\Gamma_2]$ with $0 \leq j < n$. We show $t \models_{\Gamma_1} t'$ by induction on $n$.

For $n = 0$, we have $t \models_{\Gamma_1} t'$ immediately.

We consider the case where $n > 0$.

If $t_j \equiv n^mu'$ for $0 \leq j \leq n$ and $m \in N$, then $t_j = t_{j+1} \in [\Gamma_1]$ with $0 \leq i < n$. By Lemma 34, we have $t \models_{\Gamma_1} t'$.
Assume that there exists \( j_0 \) with \( 0 \leq j_0 \leq n \), such that \( t_{j_0} \equiv n^m u' \) for \( m \in \mathbb{N} \). Since any formula of \( \Gamma_2 \) in which \( u' \) occurs is either \( n^m u = n^{m'} u' \) or \( n^{m'} u = n^m u' \) with \( l \in \mathbb{N} \) and \( \text{Var}(u') \) do not occur in \( t, t' \), we have \( t_{j_0-1} \equiv t_{j_0+1} \equiv n^{m'} u' \). Define \( t_k \) as \( t_k \) if \( 0 \leq k < j_0 \) and \( t_{k+1} \equiv t' \) if \( k \leq n - 1 \). Then, \( t_0 \equiv t, t_{n-1} \equiv t' \) and \( t_k = t_{k+1} \in \Gamma_2 \) with \( 0 \leq k < n - 1 \). By the induction hypothesis, we have \( t \not\equiv \Gamma_1 t' \).

**Lemma 36.** For a set of formulas \( \Gamma_1 \) and \( \Gamma_2 \equiv (\Gamma_1, u = u') \), if \( t \not\equiv \Gamma_1 u, u \) and \( t \not\equiv \Gamma_1 u', \) then \( t \not\equiv \Gamma_2 t' \) implies \( t \not\equiv \Gamma_1 t' \).

**Proof.** Assume \( t \not\equiv \Gamma_1 u, \) \( t \not\equiv \Gamma_1 u' \), and \( t \not\equiv \Gamma_2 t' \). By Lemma 34, there exists a sequence \( (t_i)_{0 \leq i \leq m} \) with \( m \in \mathbb{N} \) such that \( t_0 \equiv t, t_{m} \equiv t' \) and \( t_i = t_{i+1} \in \Gamma_2 \) with \( 0 \leq i < m \).

If \( t_i \not\equiv n^m u \) and \( t_i \not\equiv n^m u' \) for all \( 0 \leq i \leq n \) and any \( l \in \mathbb{N} \), then \( t_i = t_{i+1} \in \Gamma_1 \) with all \( 0 \leq i < m \). By Lemma 34, we have \( t \not\equiv \Gamma_1 t' \).

Assume that there exists \( i \) with \( 0 \leq i \leq n \), such that \( t_i \equiv n^m u \) or \( t_i \equiv n^m u' \) for some \( l \in \mathbb{N} \). Let \( i_0 \) be the least number among such \( i \)'s. Since \( i_0 \) is the least, we have \( t_i = t_{i+1} \in \Gamma_1 \) for all \( 0 \leq i < i_0 \). By Lemma 34, we have \( t \not\equiv \Gamma_1 n^m u \) or \( t \not\equiv \Gamma_1 n^m u' \). This contradicts \( t \not\equiv \Gamma_1 u \) and \( t \not\equiv \Gamma_1 u' \).

**B.2. The proof of Lemma 25**

We show Lemma 25.

Let \( (\Gamma_1, \Delta_i)_{0 \leq i \leq m} \) be an unbalanced path. We prove the statement by the induction on \( i \).

For \( i = 0 \), \( \Gamma_0 \not\equiv \Delta_0 \) is a root-likle sequent by Definition 24.

For \( i = 0 \), we consider cases according to the rule with the conclusion \( \Gamma_{i-1} \vdash \Delta_{i-1} \).

**Case 1.** The case \((\text{Weak})\).

1. By the induction hypothesis \((1)\), we have \( s \not\equiv \Gamma_{i-1} e \). By \( \Gamma_i \subseteq \Gamma_{i-1} \), we have \( s \not\equiv \Gamma_i e \).
2. Let \( F s T(t) \in \Delta_i \). By \( \Delta_i \subseteq \Delta_{i-1} \), we have \( F s T(t) \in \Delta_{i-1} \). By the induction hypothesis \((2)\), \( t \not\equiv \Gamma_{i-1} s \). By \( \Gamma_i \subseteq \Gamma_{i-1} \), we have \( t \not\equiv \Gamma_i s \).
3. Assume \( n^m s \equiv \Gamma_{i-1} n^m s \). By \( \Gamma_i \subseteq \Gamma_{i-1} \), we have \( n^m s \equiv \Gamma_{i-1} n^m s \). By the induction hypothesis \((3)\), \( n = m \).

**Case 2.** The case \((\text{Subst}) \) with a substitution \( \theta \).

1. By the induction hypothesis \((1)\), we have \( s \not\equiv \Gamma_{i-1} e \). By Lemma 32 \((ii)\), we have \( s \not\equiv \Gamma_i e \).
2. Let \( F s T(t) \in \Delta_i \). By \( \Delta_{i-1} \equiv \Delta_i \theta \), \( F s T(t \theta) \in \Delta_{i-1} \). By the induction hypothesis \((2)\), \( t \theta \not\equiv \Gamma_{i-1} s \). By Lemma 32 \((ii)\), \( t \not\equiv \Gamma_i s \).
3. Assume \( n^m s \equiv \Gamma_{i-1} n^m s \). By Lemma 32 \((i)\), we have \( n^m s \equiv \Gamma_{i-1} n^m s \). By the induction hypothesis \((3)\), \( n = m \).

**Case 3.** The case \((= L')\).

Let \( u_1 = u_2 \) be the principal formula of the rule. There exists \( \Gamma \) and \( \Delta \) such that

\[
\begin{align*}
\Gamma_{i-1} &\equiv (\Gamma[v_1 := u_1, v_2 := u_2], u_1 = u_2), \\
\Delta_{i-1} &\equiv (\Delta[v_1 := u_1, v_2 := u_2], u_1 = u_2), \\
\Gamma_i &\equiv (\Gamma[v_1 := u_2, v_2 := u_1], u_1 = u_2), \\
\Delta_i &\equiv (\Delta[v_1 := u_2, v_2 := u_1], u_1 = u_2).
\end{align*}
\]

1. By the induction hypothesis \((1)\), we have \( s \not\equiv \Gamma_{i-1} e \). By Lemma 33 \((ii)\), we have \( s \not\equiv \Gamma_i e \).
2. Let \( F s T(t) \in \Delta_i \). By the definition of \( \Delta_i \), there exists a term \( \theta \) such that \( t \equiv \theta[v_1 := u_2, v_2 := u_1] \). Then, \( F s T(\theta[v_1 := u_1, v_2 := u_2]) \in \Delta_{i-1} \). By the induction hypothesis \((2)\), \( \theta[v_1 := u_1, v_2 := u_2] \not\equiv \Gamma_{i-1} s \). By Lemma 33 \((ii)\), \( \theta[v_1 := u_2, v_2 := u_1] \not\equiv \Gamma_i s \). Thus, \( t \not\equiv \Gamma_i s \).
3. Assume \( n^m s \equiv \Gamma_i, n^m s \). By Lemma 33 \((i)\), we have \( n^m s \equiv \Gamma_{i-1} n^m s \). By the induction hypothesis \((3)\), \( n = m \).

**Case 4.** The case \((\text{Case TEF}) \) with the right assumption \( \Gamma_i \vdash \Delta_i \).

Let \( T e F(t) \) be the principal formula of the rule. There exists \( \Pi \) such that \( \Gamma_{i-1} \equiv (\Pi, T e F(t)) \) and \( \Gamma_i \equiv (\Pi, t = x, T e F(\bar{x})) \) for a fresh variable \( x \).

1. If \( s \not\approx \Gamma_i e \), then we have \( s \not\approx \Gamma_{i-1} e \) by Lemma 35. It contradicts the induction hypothesis \((1)\). Thus, \( s \not\approx \Gamma_i e \).
2. Let \( F s T(t') \in \Delta_i \). If \( t' \not\approx \Gamma_i s \), then we have \( t' \not\approx \Gamma_{i-1} s \) by Lemma 35. It contradicts the induction hypothesis. Thus, \( t' \not\approx \Gamma_i s \).
3. Assume \( n^m s \equiv \Gamma_{i-1} n^m s \). By Lemma 35, \( n^m s \equiv \Gamma_{i-1} n^m s \). By the induction hypothesis \((3)\), \( n = m \).

**Case 5.** The case \((\text{Case TEF}) \) with the left assumption \( \Gamma_i \vdash \Delta_i \). In this case, \( \Gamma_{i-1} \vdash \Delta_{i-1} \) is a switching point.
Let $\text{TeF}(t)$ be the principal formula of the rule. There exists $\Pi$ such that $\Gamma_{i-1} \equiv (\Pi, \text{TeF}(t))$ and $\Gamma_i \equiv (\Pi, t = \varepsilon)$.

Since $\Gamma_{i-1} \vdash \Delta_{i-1}$ is a switching point, we have $t \not\vdash_{\Gamma_{i-1}} s$. By the induction hypothesis (1), $s \not\vdash_{\Gamma_{i-1}} e$.

(1) Assume $s \vdash_{\Gamma_{i}} e$ for contradiction. By $t \not\vdash_{\Gamma_{i-1}} s$, $s \not\vdash_{\Gamma_{i-1}} e$ and Lemma 36, we have $s \not\vdash_{\Gamma_{i}} e$. It contradicts the induction hypothesis (1). Thus, $s \not\vdash_{\Gamma_{i}} e$.

(2) Let $\text{FsT}(t') \in \Delta_i$. Assume $t' \not\vdash_{\Gamma_{i}} s$ for contradiction. By $t \not\vdash_{\Gamma_{i-1}} s$, $s \not\vdash_{\Gamma_{i-1}} e$ and Lemma 36, we have $t' \not\vdash_{\Gamma_{i}} s$. It contradicts the induction hypothesis (2). Thus, $t' \not\vdash_{\Gamma_{i}} s$.

(3) Assume $n^s s \equiv_{\Gamma_{i}} n^m s$. By $t \not\vdash_{\Gamma_{i-1}} s$, $s \not\vdash_{\Gamma_{i-1}} e$ and Lemma 36, we have $n^s s \equiv_{\Gamma_{i-1}} n^m s$. By the induction hypothesis (3), $n = m$.

Case 6. The case ($\text{FsT} R_2$). Let $\text{FsT}(nt)$ be the principal formula of the rule.

(1) By the induction hypothesis (1), we have $s \not\vdash_{\Gamma_{i-1}} e$. Since $\Gamma_{i-1} \equiv \Gamma_i$, we have $s \not\vdash_{\Gamma_{i}} e$.

(2) Let $\text{FsT}(t') \in \Delta_i$. Define $t$ as $nt$ if $t' \equiv t$ and $t'$ otherwise. By the induction hypothesis (2), we have $t \not\vdash_{\Gamma_{i}} s$. Since $\Gamma_{i-1} \equiv \Gamma_i$, we have $t \not\vdash_{\Gamma_{i}} s$. Then, $t' \not\vdash_{\Gamma_{i}} s$.

(3) Assume $n^s s \equiv_{\Gamma_{i}} n^m s$. Since $\Gamma_{i-1} \equiv \Gamma_i$, we have $n^s s \equiv_{\Gamma_{i-1}} n^m s$. By the induction hypothesis (3), $n = m$.

**B.3. The proof of Lemma 26**

We show Lemma 26.

Let $\tau_i \equiv \text{TeF}(t_i)$.

(1) It suffices to show that $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$ holds if $t_k \not\vdash_{\Gamma_{p+k}} s$. We consider cases according to the rule with the conclusion $\Gamma_{p+k} \vdash \Delta_{p+k}$.

Case 1. If the rule is (WEAK), we have the statement by $\Gamma_{p+k+1} \subseteq \Gamma_{p+k}$.

Case 2. If the rule is (SUBST), we have the statement by Lemma 32 (ii).

Case 3. If the rule is ($= L'$), then we have the statement by Lemma 33 (ii).

Case 4. The case (CASE TeF) with the right assumption $\Gamma_{p+k+1} \vdash \Delta_{p+k+1}$. Let $\text{TeF}(t)$ be the principal formula of the rule. There exists $\Pi$ such that $\Gamma_{p+k} \equiv (\Pi, \text{TeF}(t))$ and $\Gamma_{p+k+1} \equiv (\Pi, t = x, \text{TeF}(nx))$ with a fresh variable $x$.

We prove this case by contrapositive. To show $t_k \not\vdash_{\Gamma_{p+k}} s$, assume $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$. Define $\hat{t}$ as $t$ if $t_k \equiv nx$ and $t_{k+1}$ otherwise. Since $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$ holds, we have $\hat{t} \not\vdash_{\Gamma_{p+k+1}} s$. By Lemma 35, $\hat{t} \not\vdash_{\Gamma_{p+k}} s$. By $t_k \equiv t$, we have $t_k \not\vdash_{\Gamma_{p+k}} s$.

Case 5. The case (CASE TeF) with the left assumption $\Gamma_{p+k+1} \vdash \Delta_{p+k+1}$. In this case, $\Gamma_{p+k} \vdash \Delta_{p+k}$ is a switching point.

Let $\text{TeF}(t)$ be the principal formula of the rule. There exists $\Pi$ such that $\Gamma_{p+k} \equiv (\Pi, \text{TeF}(t))$ and $\Gamma_{p+k+1} \equiv (\Pi, t = \varepsilon)$.

We prove this case by contrapositive. To show $t_k \not\vdash_{\Gamma_{p+k}} s$, assume $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$. Since $\Gamma_{p+k} \vdash \Delta_{p+k}$ is a switching point, we have $t \not\vdash_{\Gamma_{p+k}} s$. Since $\Gamma_{p+k} \vdash \Delta_{p+k}$ is a root-like sequent, we have $s \not\vdash_{\Gamma_{p+k}} e$. By Lemma 36, we see that $t_k \not\vdash_{\Gamma_{p+k}} s$.

Case 6. The case ($\text{FsT} R_2$).

In this case, since $\Gamma_{p+k}$ is the same as $\Gamma_{p+k+1}$, we have the statement.

(2) Let $d_k = n$.

Case 1. The case (WEAK).

If $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$, then $d_k = n = \bot$.

Assume $t_{k+1} \sim_{\Gamma_{p+k+1}} s$. By Definition 19, there exist $m, l \in \mathbb{N}$ such that $n^m t_{k+1} \equiv_{\Gamma_{p+k+1}} n^m s$. By $\Gamma_{p+k+1} \subseteq \Gamma_{p+k}$, we have $n^m t_{k+1} \equiv_{\Gamma_{p+k}} n^m s$. Since $t_k \equiv t_{k+1}$, we have $n^m t_k \equiv_{\Gamma_{p+k}} n^m s$. By $d_k = n$, we have $m_1 = 0 = n$. Thus, $d_{k+1} = n$.

Case 2. The case (SUBST) with a substitution $\theta$. Note that $t_k \equiv t_{k+1}[\theta]$.

If $t_{k+1} \not\vdash_{\Gamma_{p+k+1}} s$, then $d_k = n$. Assume that $t_{k+1} \sim_{\Gamma_{p+k+1}} s$. By Definition 19, there exist $m, m_1 \in \mathbb{N}$ such that $n^{m_1} t_{k+1} \equiv_{\Gamma_{p+k+1}} n^{m_1} s$. By Lemma 32 (i), $n^{m_1} t_{k+1}[\theta] \equiv_{\Gamma_{p+k}} n^{m_1} s$. Since $t_k \equiv t_{k+1}[\theta]$ holds, we have $n^{m_1} t_k \equiv_{\Gamma_{p+k}} n^{m_1} s$. By $d_k = n$, we have $m_1 = 0 = n$. Thus, $d_{k+1} = n$.

(3) Let $d_k = n$.

Case 1. The case ($= L'$) with the principal formula $u_1 = u_2$.

In this case, there exists a term $t$ such that $t_k \equiv t[v_1 := u_1, v_2 := u_2]$ and $t_{k+1} \equiv t[v_1 := u_2, v_2 := u_1]$ for variables $v_1, v_2$. 
By \( d_k = n \), there exist \( m_0, m_1 \in \mathbb{N} \) such that \( n^{m_0}t[v_1 := u_1, v_2 := u_2] \equiv \Gamma_{p+k} n^{m_1}s \) and \( m_1 - m_0 = n \). From Lemma 33 (i), \( n^{m_0}t[v_1 := u_2, v_2 := u_1] \equiv \Gamma_{p+k+1} n^{m_1}s \). Thus, \( d_{k+1} = m_1 - m_0 = n \).

Case 2. The case \( (\text{FsT R}_2) \).

Since \( \tau_{p+k+1} \equiv \tau_{p+k} \) holds and \( \Gamma_{p+k} \) is the same as \( \Gamma_{p+k+1} \), we have \( d_{k+1} = d_k \).

(4) Let \( d_k = n \). Let \( \text{TeF}(t) \) be the principal formula of the rule \( (\text{CASE TeF}) \) with the conclusion \( \Gamma_{p+k} \vdash \Delta_{p+k} \).

(4a) The case where \( \Gamma_{p+k+1} \vdash \Delta_{p+k+1} \) is the left assumption of the rule. In this case, \( \Gamma_{p+k} \vdash \Delta_{p+k} \) is a switching point. There exists \( \Pi \) such that \( \Gamma_{p+k} \equiv (\Pi, \text{TeF}(t)) \) and \( \Gamma_{p+k+1} \equiv (\Pi, t = e) \).

By \( d_k = n \), there exist \( m_0, m_1 \in \mathbb{N} \) such that \( n^{m_0}t_k \equiv \Gamma_{p+k} n^{m_1}s \) and \( m_1 - m_0 = n \).

Since the set of formulas with \( = \) in \( \Gamma_{p+k+1} \) includes the set of formulas with \( = \) in \( \Gamma_{p+k} \), we have \( n^{m_0}t_k \equiv \Gamma_{p+k+1} n^{m_1}s \). By \( \tau_k \equiv \tau_{k+1} \), we have \( n^{m_0}t_{k+1} \equiv \Gamma_{p+k+1} n^{m_1}s \). Thus, \( d_{k+1} = m_1 - m_0 = n \).

(4b) The case where \( \Gamma_{p+k+1} \vdash \Delta_{p+k+1} \) is the right assumption of the rule and \( \tau_k \) is not a progress point of the trace.

Since \( \tau_k \) is not a progress point of the trace, we have \( \tau_{k+1} \equiv \tau_k \). By \( d_k = n \), there exist \( m_0, m_1 \in \mathbb{N} \) such that \( n^{m_0}t_k \equiv \Gamma_{p+k} n^{m_1}s \) and \( m_1 - m_0 = n \).

Since the set of formulas with \( = \) in \( \Gamma_{p+k} \) includes the set of formulas with \( = \) in \( \Gamma_{p+k+1} \), we have \( n^{m_0}t_k \equiv \Gamma_{p+k+1} n^{m_1}s \). By \( \tau_{k+1} \equiv \tau_k \), we have \( n^{m_0}t_{k+1} \equiv \Gamma_{p+k+1} n^{m_1}s \). Thus, \( d_{k+1} = m_1 - m_0 = n \).

(4c) The case where \( \Gamma_{p+k+1} \vdash \Delta_{p+k+1} \) is the right assumption of the rule and \( \tau_k \) is a progress point of the trace.

There exists \( \Pi \) such that \( \Gamma_{p+k} \equiv (\Pi, \text{TeF}(t)) \) and \( \Gamma_{p+k+1} \equiv (\Pi, t = x, \text{TeF}(nx)) \) for a fresh variable \( x \). Since \( \tau_k \) is a progress point of the trace, we have \( \tau_k \equiv \text{TeF}(t) \) and \( \tau_{k+1} \equiv \text{TeF}(nx) \). Therefore, \( t_k \equiv t \) and \( t_{k+1} \equiv nx \). By \( d_k = n \), there exist \( m_0, m_1 \in \mathbb{N} \) such that \( n^{m_0}t \equiv \Gamma_{p+k} n^{m_1}s \) and \( m_1 - m_0 = n \). Since the set of formulas with \( = \) in \( \Gamma_{p+k+1} \) includes the set of formulas with \( = \) in \( \Gamma_{p+k} \), we have \( n^{m_0}t \equiv \Gamma_{p+k+1} n^{m_1}s \). By \( s \equiv n^{p+k+1}x \), we have \( n^{m_0}x \equiv \Gamma_{p+k+1} n^{m_1}s \). Hence, \( n^{m_0}nx \equiv \Gamma_{p+k+1} n^{m_1}ns \). Therefore, \( n^{m_0}t_{k+1} \equiv \Gamma_{p+k+1} n^{m_1+1}s \).

Thus, \( d_{k+1} = m_1 + 1 - m_0 = n + 1 \).