循環証明体系におけるカット除去の反例について

総合研究大学院大学複合科学研究科情報学専攻 4 年 益岡 幸弘
Yukihiro Masuoka
The Graduate University for Advanced Studies (SOKENDAI)

December 3, 2021

概要

循環証明体系とはその証明図がサイクル付きの木の形になっている証明体系である．J. Brotherston は帰納的定義付き一階述語論理に対する循環証明体系 $\mathtt{CLKID}^\omega$ を提案した [4]．彼は $\mathtt{CLKID}^\omega$ におけるカット除去性が成り立たないと予想した [2]．その予想が正しいことを示した [11] の内容を紹介する．証明は反例を与えることによって行われる．つまり，$\mathtt{CLKID}^\omega$ でカットありでは証明可能であるが，カットなしでは証明できないシークエントを具体的に与える．反例となるシークエントには 2 種類の帰納的述語が出現する．これらの帰納的述語は直感的にはどちらも自然数の加法を定義していると解釈される．反例となるシークエントがカットありで証明可能なことは具体的な証明図を与えることで示す．反例となるシークエントがカットなしで証明できないことは，そのシークエントのカットなし証明図の存在を仮定し，$\mathtt{CLKID}^\omega$ の証明図の有限性に矛盾させることで示す．

Keyword: **循環証明体系** (*cyclic/circular proof system*)，**カット除去** (*cut-elimination*)，**帰納的定義** (*inductive definition*)，**帰納法** (*induction*)，**証明論** (*proof theory*)

1 はじめに

循環証明体系（*cyclic proof system* または *circular proof system*）は証明図の形がサイクル付きの木になっている証明体系である．少数の例外を除いて，循環証明体系は，帰納法を含んだ証明体系において，帰納法の導出則を別の導出則に置き換えた上で，証明図の形をサイクル付きの木に拡張した体系である．これは**フェルマーの無限降下法**などの形式化とも解釈できる．

循環証明体系において証明を考える場合，帰納法の導出則がないため，「帰納法を適用する論理式」を与える必要がない．このことは，証明探索という観点において，従来の帰納法を含む体系と比べて循環証明体系が優れている点である．実際，symbolic heap と呼ばれる分離論理の部分体系の decision procedure に循環証明体系は用いられている [5, 15]．

カット除去性（*cut-elimination property*）は証明体系が持っていると望ましい基本的性質である．たとえば，一階述語論理のシークエント計算におけるカット除去性はその無矛盾性，部分論理式性，および Craig の補間定理を導くことが知られている（詳細は [7] を参照せよ）．

循環証明体系がカット除去性を持つ時，任意の論理式が主論理式となりうる論理規則がないため，証明に出現する論理式の形をかなり制限することができる．このことは，その循環証明体系において，効率的な証明探索法があることを意味している．

カット除去性の重要性にも関わらず，**帰納的一階述語論理に対する循環証明体系** $\mathtt{CLKID}^\omega$ のカット除去性の有無は長年の間未解決のままであった．Brotherston は次のように予想をしていた [2]．

> "**Conjecture 5.2.4.** *Cut is not eliminable in the system* $CLKID^\omega$*. That is to say, there exists a sequent* $\Gamma \vdash \Delta$ *of* FOL_{ID} *which is provable in* $CLKID^\omega$*, but is not provable without the use of the rule (Cut).*"

ここで，$\mathrm{FOL_{ID}}$ というのは帰納的定義付き一階述語論理の言語のことである．

筆者を含むグループはこの Brotherston の予想が正しいことを示した [11]．つまり，帰納的定義付き一階述語論理に対する循環証明体系 $\mathtt{CLKID}^\omega$ においてカット除去はできないことを示した．その証明の概略を紹介するのがこの文書の目的である．

証明は反例を与えることによって行われる．つまり，$\mathtt{CLKID}^\omega$ でカットありでは証明可能であるが，カットなしでは証明できないシークエントを具体的に与える．反例となるシークエントには 2 種類の帰納的述語が出現する．これらの帰納的述語は直感的にはどちらも自然数の加法を定義していると解釈される．反例となるシークエントがカットありで $\mathtt{CLKID}^\omega$ に証明できることは証明図を具体的に与えることで示す．そのシークエントがカットなしでは証明できないことは背理法によって示す．具体的には反例のシークエントの $\mathtt{CLKID}^\omega$ におけるカットなしの証明図が存在することを仮定し，index, switching point などの概念を導入することで $\mathtt{CLKID}^\omega$ の証明図の有限性に矛盾させる．

関連研究： 様相 μ 計算（modal μ-calculus）[1]，線型時相 μ 計算（linear time μ-calculus）[6]，Gödel-Löb の証明可能性論理（Gödel-Löb provability logic）[14] に対しては，カットなしでかつ完全な循環証明体系があることが知られている．それに対して，分離論理（separation logic）[9]，帰納的定義付きバンチ論理（Bunched Logic with inductive propositions）[13]，一階算術（first-order arithmetic）[8] に対しては，カット無しでは証明できないがカットありならば証明できるシークエントの存在が知られている．

本文構成： まず，第 2 節で帰納的定義付き一階述語言語の定義を行う．第 3 節で帰納的定義付き一階述語言語の循環証明体系である $\mathtt{CLKID}^\omega$ を定義する．第 4 節で $\mathtt{CLKID}^\omega$ のカット除去性の反例とその証明の概略を紹介する．

2 帰納的定義付き一階述語言語

この節では**帰納的定義付き一階述語言語**（*first-order logic with inductive definitions*）を定義する．

帰納的定義付き一階述語言語の記号は一階述語言語の記号に**帰納的述語記号**（*inductive predicate symbol*）$P_1, \ldots, P_n$ を加えたものである．以下では，帰納的述語記号でない述語記号を**通常の述語記号**（*ordinal predicate symbol*）と呼び区別する．帰納的定義付き一階述語言語の項，（原子）論理式，自由変数，束縛変数などは一階述語論理と同様に定義する．ただし，言語に含まれる論理記号は $\neg, \vee, \wedge, \rightarrow, \forall, \exists$ とする．

次に帰納的定義の生成ステップの形式化である**プロダクション**（*production*）という概念を定義する．

定義 1 (プロダクション)**.** 原子論理式の有限集合と帰納的述語を含む原子論理式のペアを**プロダクション**と呼ぶ．

プロダクションは通常，次のように導出規則のように書かれることが多い．

$$\frac{Q_1\mathbf{u}_1 \quad \cdots \quad Q_h\mathbf{u}_h \quad P_{j_1}\mathbf{t}_1 \quad \cdots \quad P_{j_m}\mathbf{t}_m}{P_i\mathbf{t}}$$

ただし，$Q_1\mathbf{u}_1, \ldots, Q_h\mathbf{u}_h$ は通常の述語記号を含む原子論理式，$P_{j_1}\mathbf{t}_1, \ldots, P_{j_m}\mathbf{t}_m, P_i\mathbf{t}$ は帰納的述語記号を含む原子論理式である．

プロダクションの有限集合を**帰納的定義集合**（*inductive definition set*）と呼び，**帰納的定義付き一階述語言語**は帰納的定義付き一階述語言語の記号の集合と帰納的定義集合のペアとして定義される．

例 2. プロダクションの例をあげる．

$$\frac{}{\mathrm{N}(0)} \qquad \frac{\mathrm{N}(x)}{\mathrm{N}(sx)} \qquad \frac{}{\mathrm{E}(0)} \qquad \frac{\mathrm{E}(x)}{\mathrm{O}(sx)} \qquad \frac{\mathrm{O}(x)}{\mathrm{E}(sx)}$$

N, E, O はそれぞれ自然数全体，偶数全体，奇数全体を定義していると直感的には理解される．

帰納的定義付き一階述語論理の意味論において，帰納的述語記号に対応する集合は帰納的定義集合から生成される単調写像の最小不動点として定義される．帰納的定義付き一階述語論理の意味論の詳細については [3] などを参照せよ．

3 循環証明体系 CLKID$^\omega$

この節では帰納的定義付き一階述語言語に対する循環証明体系である CLKID$^\omega$ を定義する．CLKID$^\omega$ はシークエント計算の一種である．

定義 3 (シークエント)．**シークエント**（*sequent*）とは 2 つの論理式の有限集合のペアであり，$\Gamma \vdash \Delta$ と書く（ただし，Γ, Δ は論理式の有限集合）．

図 1 に帰納的述語のための規則を除いた CLKID$^\omega$ の導出規則をまとめた．コンマ “,” は集合の結合を意味する．また，Contraction は必要な時に暗に使われるものとする．それぞれの規則の主論理式は下線を引いた論理式と定義する．

帰納的述語のための規則を定義する．まず，右導出規則についてであるが，帰納的定義集合の各元ごとに対応する推論規則が定義される．

$$\frac{Q_1\mathbf{u}_1(\mathbf{x}) \quad \cdots \quad Q_h\mathbf{u}_h(\mathbf{x}) \quad P_{j_1}\mathbf{t}_1(\mathbf{x}) \quad \cdots \quad P_{j_m}\mathbf{t}_m(\mathbf{x})}{P_i\mathbf{t}(\mathbf{x})}$$

を帰納的定義集合の元とする．この元に対応する右導出規則は

$$\frac{\Gamma \vdash Q_1\mathbf{u}_1(\mathbf{u}), \Delta \ \cdots \ \Gamma \vdash Q_h\mathbf{u}_h(\mathbf{u}), \Delta \quad \Gamma \vdash P_{j_1}\mathbf{t}_1(\mathbf{u}), \Delta \ \cdots \ \Gamma \vdash P_{j_m}\mathbf{t}_m(\mathbf{u}), \Delta}{\Gamma \vdash P_i\mathbf{t}(\mathbf{u}), \Delta}\ (P_i\ \mathrm{R}_r)$$

である．この規則の主論理式は結論部の $P_i\mathbf{t}(\mathbf{u})$ と定義する．

次に左導出規則 case-split rule を定義する．そのために case distinction と呼ばれる概念を導入する．$P_i\mathbf{u}$ について，帰納的定義集合の元

$$\frac{Q_1\mathbf{u}_1(\mathbf{x}) \quad \ldots \quad Q_h\mathbf{u}_h(\mathbf{x}) \quad P_{j_1}\mathbf{t}_1(\mathbf{x}) \quad \ldots \quad P_{j_m}\mathbf{t}_m(\mathbf{x})}{P_i\mathbf{t}(\mathbf{x})}$$

に対応する case distinction とは

$$\Gamma, \mathbf{u} = \mathbf{t}(\mathbf{y}), Q_1\mathbf{u}_1(\mathbf{y}), \ldots, Q_h\mathbf{u}_h(\mathbf{y}), P_{j_1}\mathbf{t}_1(\mathbf{y}), \ldots, P_{j_m}\mathbf{t}_m(\mathbf{y}) \vdash \Delta$$

である（ただし，$\mathbf{y}$ は $\mathbf{x}$ と同じ長さのの fresh variables の列）．帰納的述語 P_i に対する左導出規則 case-split rule は

$$\frac{}{\Gamma \vdash \Delta}\,(\textsc{Axiom})(\Gamma \cap \Delta \neq \emptyset) \qquad \frac{\Gamma' \vdash \Delta'}{\Gamma \vdash \Delta}\,(\textsc{Weak})(\Gamma' \subseteq \Gamma,\ \Delta' \subseteq \Delta)$$

$$\frac{\Gamma \vdash \underline{\varphi},\ \Delta \qquad \Gamma,\ \underline{\varphi} \vdash \Delta}{\Gamma \vdash \Delta}\,(\textsc{Cut}) \qquad \frac{\Gamma \vdash \Delta}{\Gamma\,[\theta] \vdash \Delta\,[\theta]}\,(\textsc{Subst})$$

構造規則

$$\frac{\Gamma \vdash \varphi,\ \Delta}{\Gamma,\ \underline{\neg\varphi} \vdash \Delta}\,(\neg\ \mathrm{L}) \qquad \frac{\Gamma,\ \varphi \vdash \Delta}{\Gamma \vdash \underline{\neg\varphi},\ \Delta}\,(\neg\ \mathrm{R})$$

$$\frac{\Gamma,\ \varphi \vdash \Delta \qquad \Gamma,\ \psi \vdash \Delta}{\Gamma,\ \underline{\varphi \vee \psi} \vdash \Delta}\,(\vee\ \mathrm{L}) \qquad \frac{\Gamma \vdash \varphi,\ \psi,\ \Delta}{\Gamma \vdash \underline{\varphi \vee \psi},\ \Delta}\,(\vee\ \mathrm{R})$$

$$\frac{\Gamma,\ \varphi,\ \psi \vdash \Delta}{\Gamma,\ \underline{\varphi \wedge \psi} \vdash \Delta}\,(\wedge\ \mathrm{L}) \qquad \frac{\Gamma \vdash \varphi,\ \Delta \qquad \Gamma \vdash \psi,\ \Delta}{\Gamma \vdash \underline{\varphi \wedge \psi},\ \Delta}\,(\wedge\ \mathrm{R})$$

$$\frac{\Gamma \vdash \varphi,\ \Delta \qquad \Gamma,\ \psi \vdash \Delta}{\Gamma,\ \underline{\varphi \to \psi} \vdash \Delta}\,(\to\ \mathrm{L}) \qquad \frac{\Gamma,\ \varphi \vdash \psi,\ \Delta}{\Gamma \vdash \underline{\varphi \to \psi},\ \Delta}\,(\to\ \mathrm{R})$$

$$\frac{\Gamma,\ \varphi\,[x := t] \vdash \Delta}{\Gamma,\ \underline{\forall x \varphi} \vdash \Delta}\,(\forall\ \mathrm{L}) \qquad \frac{\Gamma \vdash \varphi,\ \Delta}{\Gamma \vdash \underline{\forall x \varphi},\ \Delta}\,(\forall\ \mathrm{R})(x \notin \mathrm{FV}(\Gamma \cup \Delta))$$

$$\frac{\Gamma,\ \varphi \vdash \Delta}{\Gamma,\ \underline{\exists x \varphi} \vdash \Delta}\,(\exists\ \mathrm{L})(x \notin \mathrm{FV}(\Gamma \cup \Delta)) \qquad \frac{\Gamma \vdash \varphi\,[x := t],\ \Delta}{\Gamma \vdash \underline{\exists x \varphi},\ \Delta}\,(\exists\ \mathrm{R})$$

論理的規則

$$\frac{\Gamma\,[x := u,\ y := t] \vdash \Delta\,[x := u,\ y := t]}{\Gamma\,[x := t,\ y := u],\ \underline{t = u} \vdash \Delta\,[x := t,\ y := u]}\,(=\ \mathrm{L}) \qquad \frac{}{\Gamma \vdash \underline{t = t},\ \Delta}\,(=\ \mathrm{R})$$

等号規則

図 1　CLKID$^{\omega}$ の導出規則（帰納的述語のための規則は除く）

$$\frac{}{\mathrm{N}(0)} \qquad \frac{\mathrm{N}(x)}{\mathrm{N}(sx)}$$

$$\frac{}{\Gamma \vdash \Delta,\ \mathrm{N}(0)}\ (\mathrm{N\ R}_1) \qquad \frac{\Gamma \vdash \Delta,\ \mathrm{N}(t)}{\Gamma \vdash \Delta,\ \mathrm{N}(st)}\ (\mathrm{N\ R}_2)$$

$$x \text{ is fresh } \frac{\Gamma,\ t = 0 \vdash \Delta \qquad \Gamma,\ t = sx,\ \mathrm{N}(x) \vdash \Delta}{\Gamma,\ \mathrm{N}(t) \vdash \Delta}\ (\textsc{Case}\ \mathrm{N})$$

$$\frac{}{\mathrm{E}(0)} \qquad \frac{\mathrm{E}(x)}{\mathrm{O}(sx)} \qquad \frac{\mathrm{O}(x)}{\mathrm{E}(sx)}$$

$$\frac{}{\Gamma \vdash \Delta,\ \mathrm{E}(0)}\ (\mathrm{E\ R}_1) \qquad \frac{\Gamma \vdash \Delta,\ \mathrm{E}(t)}{\Gamma \vdash \Delta,\ \mathrm{O}(st)}\ (\mathrm{O\ R}) \qquad \frac{\Gamma \vdash \Delta,\ \mathrm{O}(t)}{\Gamma \vdash \Delta,\ \mathrm{E}(st)}\ (\mathrm{E\ R}_2)$$

$$x \text{ is fresh } \frac{\Gamma,\ t = 0 \vdash \Delta \qquad \Gamma,\ t = sx,\ \mathrm{O}(x) \vdash \Delta}{\Gamma,\ \mathrm{E}(t) \vdash \Delta}\ (\textsc{Case}\ \mathrm{E})$$

$$x \text{ is fresh } \frac{\Gamma,\ t = sx,\ \mathrm{E}(x) \vdash \Delta}{\Gamma,\ \mathrm{O}(t) \vdash \Delta}\ (\textsc{Case}\ \mathrm{O})$$

図 2　N, E, O に対する導出規則

$$\frac{P_i\mathbf{u} \text{ について } P_i \text{ を結論として持つ帰納的定義集合の元に対応する case distinction すべて}}{\Gamma,\ P_i\mathbf{u} \vdash \Delta}\ (\textsc{Case}\ P_i)$$

と定義する．この規則の主論理式は結論部の $P_i\mathbf{u}$ と定義する．($\textsc{Case}\ P_i$) の主論理式 $P_i\mathbf{u}$ の *Case-descendant* とは仮定部で新しく導入される帰納的述語を伴う原子論理式のことを言う．

例 4. 例 2 で定義された N, E, O に対するプロダクションと導出規則を図 2 にまとめた．対応を見やすくするため，対応するプロダクションと右導出規則は縦に並べて書いてある．

シークエントをラベルとして持つグラフで，各ノードと辺の関係が CLKID$^\omega$ の導出規則になっているものを CLKID$^\omega$ **の導出図**と呼ぶ．CLKID$^\omega$ の導出図で特に台となるグラフが木であるものを CLKID$^\omega$ **の導出木**と呼ぶ．以下では CLKID$^\omega$ の導出図は結論から仮定部の方向に向き付けがなされているとする．CLKID$^\omega$ の導出図において，どの導出規則の結論にもなっていないノードを**芽**（*bud*）と呼ぶ．CLKID$^\omega$ の導出図における芽は必ず葉になっていることに注意する．CLKID$^\omega$ の導出図において，ある芽と同じシークエントをラベルとして持つ芽以外のノードを**コンパニオン**（*companion*）と呼ぶ．

定義 5 (CLKID$^\omega$ **の擬証明**)．CLKID$^\omega$ の有限導出木 $\mathscr{P}$ と $\mathscr{P}$ の各芽に対して唯一つのコンパニオンを対応させる写像 $\mathcal{C}$ の組 $(\mathscr{P}, \mathcal{C})$ を CLKID$^\omega$ **の擬証明**（*pre-proof*）または単に**擬証明**と呼ぶ．

CLKID$^\omega$ の擬証明 $(\mathscr{P}, \mathcal{C})$ において，$\mathcal{C}$ によって対応付けられた芽とコンパニオンを同一視することによって得られる導出図を $(\mathscr{P}, \mathcal{C})$ **のグラフ** と呼び，$\mathcal{G}(\mathscr{P}, \mathcal{C})$ と書く．

CLKID$^\omega$ の擬証明は芽とコンパニオンの同一視によって，循環構造があると見なせる．循環構造を認めてしまうため 図 3 のように矛盾を導く擬証明が比較的簡単に書けてしまう．そのため，**大域トレース条件**と呼ばれる条件を満たす擬証明のみを CLKID$^\omega$ の証明とする必要がある．

大域トレース条件を定義するため，**トレース**を定義する．

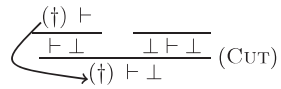

図 3　矛盾を導く CLKID$^\omega$ の擬証明

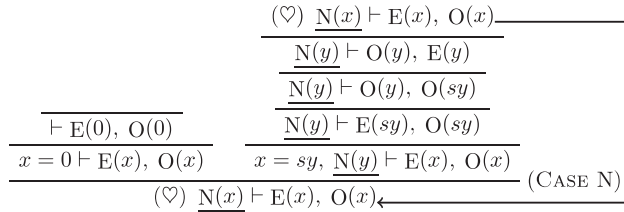

図 4　CLKID$^\omega$ の証明の例

定義 6 (トレース). T を導出図，$(\Gamma_i \vdash \Delta_i)_{0\le i<\alpha}$ を T の道[1]とする．$(\Gamma_i \vdash \Delta_i)_{0\le i<\alpha}$ のトレースとは次の条件を満たす論理式の列 $(\tau_i)_{0\le i<\alpha}$ である．

(1)τ_i は Γ_i の元であって，帰納的述語を伴う原子論理式.
(2)もし $\Gamma_i \vdash \Delta_i$ が θ による (Subst) の結論であれば, $\tau_i \equiv \tau_{i+1}[\theta]$.
(3)もし $\Gamma_i \vdash \Delta_i$ が $t = u$ を主論理式とする $(=\mathrm{L})$ の結論であれば，$\tau_i \equiv F[x := t, y := u]$ かつ $\tau_{i+1} \equiv F[x := u, y := t]$.
(4)もし $\Gamma_i \vdash \Delta_i$ が (Case P_i) の結論であるとする.
- τ_{i+1} は τ_i と同じである，または
- τ_i が主論理式であって τ_{i+1} は τ_i の case-descendant

後者の場合， i をトレースの**進歩地点**（*progress point*） と言う.
(5)もし $\Gamma_i \vdash \Delta_i$ がその他の導出規則の結論である場合は $\tau_{i+1} \equiv \tau_i$.

無限の進歩地点を持つトレースを**無限に進歩するトレース**（*infinitely progressing trace*）と呼ぶ.

定義 7 (大域トレース条件 Global trace condition). ある導出図が次の条件を満たすとする.

条件: その導出図の中のすべての無限道 $(v_i)_{i\ge 0}$ について，その尾（tail of the path） $(v_i)_{i\ge k}$ （$k \ge 0$）が無限に進歩するトレースを持つ.

このとき，その導出図は**大域トレース条件**（*global trace condition*）を満たすと言う.

定義 8 (CLKID$^\omega$ の証明). CLKID$^\omega$ の擬証明で，そのグラフが大域トレース条件を満たすものを CLKID$^\omega$ **の証明**と呼ぶ.

図 3 の擬証明は case-split rule が出現しないことから，進歩する点がどのような道に対しても存在しない．それゆえ，この擬証明は大域トレース条件を満たさない.

例 9 (CLKID$^\omega$ の証明の例). 図 4 は $\mathrm{N}(x) \vdash \mathrm{E}(x), \mathrm{O}(x)$ に対する CLKID$^\omega$ の証明の例である（紙面の都合で導出規則のラベルの一部は省略した）．(♡) は芽とコンパニオンの対応を表し，下線は無限道に対応する無限に進歩するトレースを表している.

1)　この文書で道と言った場合，頂点と辺の反復を許す.

$$\frac{}{\Gamma \vdash \mathrm{Add}_1(0,\ b,\ b),\ \Delta}\ (\mathrm{Add}_1\ \mathrm{R}_1) \qquad \frac{\Gamma \vdash \Delta,\ \mathrm{Add}_1(a,\ b,\ c)}{\Gamma \vdash \Delta,\ \mathrm{Add}_1(sa,\ b,\ sc)}\ (\mathrm{Add}_1\ \mathrm{R}_2)$$

$$\frac{}{\Gamma \vdash \mathrm{Add}_2(0,\ b,\ b),\ \Delta}\ (\mathrm{Add}_2\ \mathrm{R}_1) \qquad \frac{\Gamma \vdash \Delta,\ \mathrm{Add}_2(a,\ sb,\ c)}{\Gamma \vdash \Delta,\ \mathrm{Add}_2(sa,\ b,\ c)}\ (\mathrm{Add}_2\ \mathrm{R}_2)\ ,$$

$$\frac{\Gamma,\ a=0,\ b=y,\ c=y \vdash \Delta \qquad \Gamma,\ a=sx,\ b=y,\ c=sz,\ \mathrm{Add}_1(x,\ y,\ z) \vdash \Delta}{\Gamma,\ \mathrm{Add}_1(a,\ b,\ c) \vdash \Delta}\ (\textsc{Case}\ \mathrm{Add}_1)$$

（ただし，$x, y, z \notin \mathrm{FV}(\Gamma \cup \Delta \cup \{\mathrm{Add}_1(a,\ b,\ c)\})$ で，x, y, z は相異なる）

$$\frac{\Gamma,\ a=0,\ b=y,\ c=y \vdash \Delta \qquad \Gamma,\ a=sx,\ b=y,\ c=z,\ \mathrm{Add}_2(x,\ sy,\ z) \vdash \Delta}{\Gamma,\ \mathrm{Add}_2(a,\ b,\ c) \vdash \Delta}\ (\textsc{Case}\ \mathrm{Add}_2)$$

（ただし，$x, y, z \notin \mathrm{FV}(\Gamma \cup \Delta \cup \{\mathrm{Add}_2(a,\ b,\ c)\})$ で，x, y, z は相異なる）．

図 5　Add_1, Add_2 の導出規則

4　CLKID$^\omega$ のカット除去性の反例

この節の目的は次の定理の証明の概略を説明することである．

定理 10. 0 を定数記号，s を *1* 引数関数記号とする．Add_1, Add_2 を次のように定義される 3 項帰納的述語とする．

$$\frac{}{\mathrm{Add}_1(0,\ y,\ y)}\ ,\quad \frac{\mathrm{Add}_1(x,\ y,\ z)}{\mathrm{Add}_1(\mathrm{s}x,\ y,\ \mathrm{s}z)}\ ,\quad \frac{}{\mathrm{Add}_2(0,\ y,\ y)}\ ,\quad \frac{\mathrm{Add}_2(x,\ \mathrm{s}y,\ z)}{\mathrm{Add}_2(\mathrm{s}x,\ y,\ z)}$$

このとき，次の 2 つの主張が成り立つ．

(1)$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ は CLKID$^\omega$ において証明可能．

(2)$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ は CLKID$^\omega$ においてカットなしでは証明できない．

定理 10 は $\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ というシークエントが CLKID$^\omega$ のカット除去性の反例であることを主張している．

議論のために Add_1, Add_2 に対する導出規則を 図 5 にまとめておく．

Add_1 と Add_2 は形式的な定義の違う述語である．しかし，どちらも自然数の加法を定義していると直感的には理解される．そのため，$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ というシークエントが CLKID$^\omega$ で証明可能であることが期待される．実際，図 6 のような CLKID$^\omega$ の証明がある．図 6 において，(†)，(⋆) は芽とコンパニオンの対応を表し，下線は無限道に対応する無限に進歩するトレースを表している．証明図中に (Cut) が使われていることに注意してほしい．$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ の CLKID$^\omega$ での証明があることから，定理 10(1) は成立する．

以下では 定理 10(2) の証明の概略を述べる．詳細は [11] を参照されたい．

$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ のカットなし証明 $(\mathcal{D}_{\mathrm{cf}},\ \mathcal{C}_{\mathrm{cf}})$ には無限の頂点があることを示して矛盾を導く．CLKID$^\omega$ は cycle-normalization という「芽に対応するコンパニオンを芽の祖先に制限しても（カットなし）証明可能性は変わらない」という性質を持っている [2]．この性質から，このカットなし証明 $(\mathcal{D}_{\mathrm{cf}},\ \mathcal{C}_{\mathrm{cf}})$ における芽に対応するコンパニオンは芽の祖先であるとして良い．

カットなし証明 $(\mathcal{D}_{\mathrm{cf}},\ \mathcal{C}_{\mathrm{cf}})$ に出現する項は $\mathrm{s}^n x$, $\mathrm{s}^n 0$ のいずれかの形をしていることに注意する．以下の議論では項の形を $\mathrm{s}^n x$, $\mathrm{s}^n 0$ に制限して議論する．

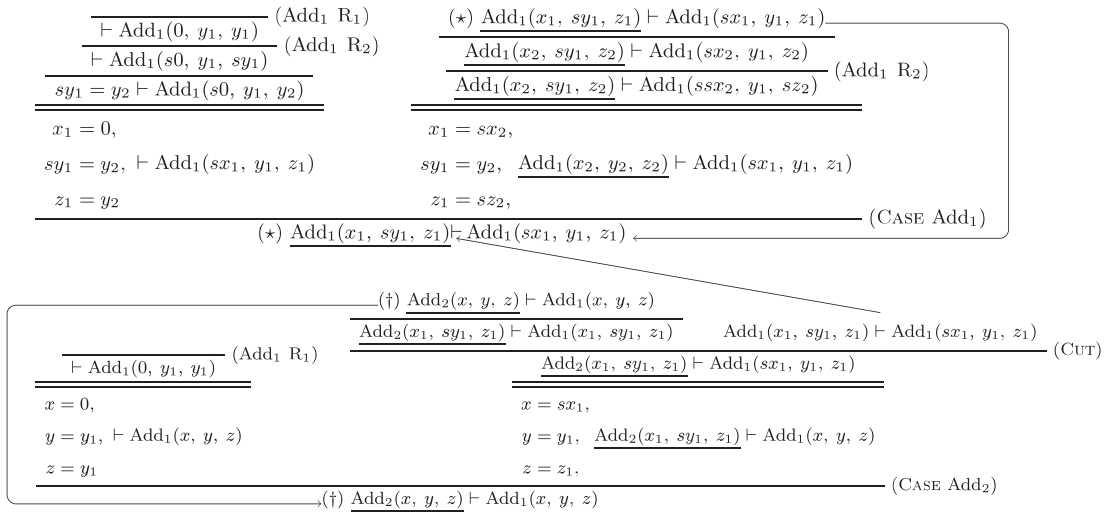

図 6　$\mathrm{Add}_2(x,\, y,\, z) \vdash \mathrm{Add}_1(x,\, y,\, z)$ に対する $\mathtt{CLKID}^\omega$ の証明

定義 11 ($\cong_\Gamma$)．「$t_1 = t_2 \in \Gamma$ ならば $t_1 \cong_\Gamma t_2$」を満たす項全体の集合上の最小の合同関係 the smallest congruence relation を $\cong_\Gamma$ と書く．

$t_1 \cong_\Gamma t_2$ は任意の Γ のモデルで $t_1 = t_2$ が成立することを直感的には意味している．

定義 12 ($\sim_\Gamma$)．項全体の集合上の関係 $t_1 \sim_\Gamma t_2$ を次のように定める．

$$t_1 \sim_\Gamma t_2 :\Longleftrightarrow \mathrm{s}^n t_1 \cong_\Gamma \mathrm{s}^m t_2 \text{ for some } n,\, m \in \mathbb{N}$$

$t_1 \sim_\Gamma t_2$ もまた合同関係であることに注意する．これらの関係を用いて index という概念を定義する．

定義 13 (index)．$\Gamma \vdash \Delta$ をシークエントとする．$\mathrm{Add}_2(a,\, b,\, c) \in \Gamma$ について，$\Gamma \vdash \Delta$ における $\mathrm{Add}_2(a,\, b,\, c)$ の *index* を次のように定義する．

(1)任意の $\mathrm{Add}_1(a',\, b',\, c') \in \Delta$ について $b \not\sim_\Gamma b'$ のとき，$\Gamma \vdash \Delta$ における $\mathrm{Add}_2(a,\, b,\, c)$ の index を $\bot$ と定義する．

(2)$\mathrm{Add}_1(a',\, b',\, c') \in \Delta$ について $s^n b \cong_\Gamma s^m b'$ （$n,\, m \in \mathbb{N}$） かつ $m - n$ が一意に決まるとき，$\Gamma \vdash \Delta$ における $\mathrm{Add}_2(a,\, b,\, c)$ の index を $m - n$ と定義する（ここで，$m - n$ が一意とは「$s^{n'} b \cong_\Gamma s^{m'} b''$ かつ $\mathrm{Add}_1(a'',\, b'',\, c'') \in \Delta$ ならば $m - n = m' - n'$」を意味する）．

$\mathrm{Add}_1(a',\, b',\, c') \in \Delta$ について $s^n b \cong_\Gamma s^m b'$ （$n,\, m \in \mathbb{N}$） だが $m - n$ が一意でないときは undefined であることに注意せよ．また，次の三点にも注意する．

- index が $\bot$ でないとき，進歩点では，トレースの index は増える（図 7）．
- index が $\bot$ でないとき，進歩点以外では，トレースの index は増えないか，$\bot$ に変わるかのいずれかである．
- index が $\bot$ のときはその次のトレースの index は変わらない．

$$\dfrac{\vdots \quad x = sx_1,\ y = y_1,\ z = z_1,\ \mathrm{Add}_2(x_1,\ sy_1,\ z_1)\mathbf{[1]} \vdash \mathrm{Add}_1(x,\ y,\ z)}{\mathrm{Add}_2(x,\ y,\ z)\mathbf{[0]} \vdash \mathrm{Add}_1(x,\ y,\ z)}\ (\textsc{Case}\ \mathrm{Add}_2)$$

図 7　進歩点では，トレースの index は増える

定義 14 (switching point)．($\textsc{Case}\ \mathrm{Add}_2$) の結論になっておりかつその主論理式の index が $\bot$ になっている頂点を *switching point* と呼ぶ．

定義 15 (index sequent)．シークエント $\Gamma \vdash \Delta$ が次の条件をすべて満たすとき *index sequent* と呼ぶ．

(1)$t \in \mathrm{B}_1(\Gamma \vdash \Delta)$, $u \in \mathrm{C}(\Gamma \vdash \Delta)$ について，$s^n t \not\cong_\Gamma s^m u$ for any $n, m \in \mathbb{N}$.
(2)$s^n b \cong_\Gamma s^m b'$ $(b, b' \in \mathrm{B}_1(\Gamma \vdash \Delta))$ ならば $n = m$.

$$\mathrm{B}_1(\Gamma \vdash \Delta) = \{b | \mathrm{Add}_1(a,\ b,\ c) \in \Delta\},$$
$$\mathrm{C}(\Gamma \vdash \Delta) = \{c | \mathrm{Add}_2(a,\ b,\ c) \in \Gamma \text{ or } \mathrm{Add}_1(a,\ b,\ c) \in \Delta\}.$$

$\mathrm{Add}_2(x,\ y,\ z) \vdash \mathrm{Add}_1(x,\ y,\ z)$ が index sequent であることに注意する．

補題 16. *index sequent* の前件に出現する $\mathrm{Add}_2(a,\ b,\ c)$ の *index* は常に定義される．

証明.　$\Gamma \vdash \Delta$ を index sequent とする．$\mathrm{Add}_2(a,\ b,\ c) \in \Gamma$ とする．

任意の $\mathrm{Add}_1(a',\ b',\ c') \in \Delta$ について $b \not\sim_\Gamma b'$ のとき $\mathrm{Add}_2(a,\ b,\ c)$ の index は $\bot$ である．

$\mathrm{Add}_1(a'_0,\ b'_0,\ c'_0) \in \Delta$ について $b \sim_\Gamma b'_0$ と仮定する．定義 12 から $\mathrm{s}^{n_0} b \cong_\Gamma \mathrm{s}^{m_0} b'_0$ を満たす自然数 n_0, m_0 が存在する．$m_0 - n_0$ の一意性を示すために，$\mathrm{Add}_1(a'_1,\ b'_1,\ c'_1) \in \Delta$ について $\mathrm{s}^{n_1} b \cong_\Gamma \mathrm{s}^{m_1} b'_1$ を仮定する．$\mathrm{s}^{n_0+n_1} b \cong_\Gamma \mathrm{s}^{m_0+n_1} b'_0$ かつ $\mathrm{s}^{n_1+n_0} b \cong_\Gamma \mathrm{s}^{m_1+n_0} b'_1$ であるから，$\mathrm{s}^{m_0+n_1} b'_0 \cong_\Gamma \mathrm{s}^{m_1+n_0} b'_1$．定義 15(2) から $m_0 + n_1 = m_1 + n_0$．よって，$m_0 - n_0 = m_1 - n_1$．ゆえに，$\mathrm{Add}_2(a,\ b,\ c)$ の index は $m_0 - n_0$ と定義される．□

定義 17 (index path)．パス $(\Gamma_i \vdash \Delta_i)_{0 \le i < \alpha}$ が次の条件をすべて満たすとき *index path* と呼ぶ．

(1)$\Gamma_0 \vdash \Delta_0$ は index sequent.
(2)もし $\Gamma_i \vdash \Delta_i$ $(i > 0)$ が ($\textsc{Case}\ \mathrm{Add}_2$) の左の仮定であるならば $\Gamma_{i-1} \vdash \Delta_{i-1}$ は switching point.

index sequent から始まる一番右側のパスは index path であることに注意する．

補題 18. *index path* に出現するすべてのシークエントは *index sequent*.

証明概略.　index path の長さについての帰納法による．□

switching point 以外の ($\textsc{Case}\ \mathrm{Add}_2$) は index を増加させてしまうことと，証明図内の無限 path には ($\textsc{Case}\ \mathrm{Add}_2$) が無限に存在することを合わせると次の補題が成り立つ．

補題 19. $(\Gamma_i \vdash \Delta_i)_{i \ge 0}$ を *infinite index path* とする．次の条件を満たす $l \in \mathbb{N}$ が存在する．

(1)$\Gamma_l \vdash \Delta_l$ は *switching point*.
(2)$\Gamma_{l+1} \vdash \Delta_{l+1}$ は *(*$\textsc{Case}\ \mathrm{Add}_2$*)* の右の仮定.

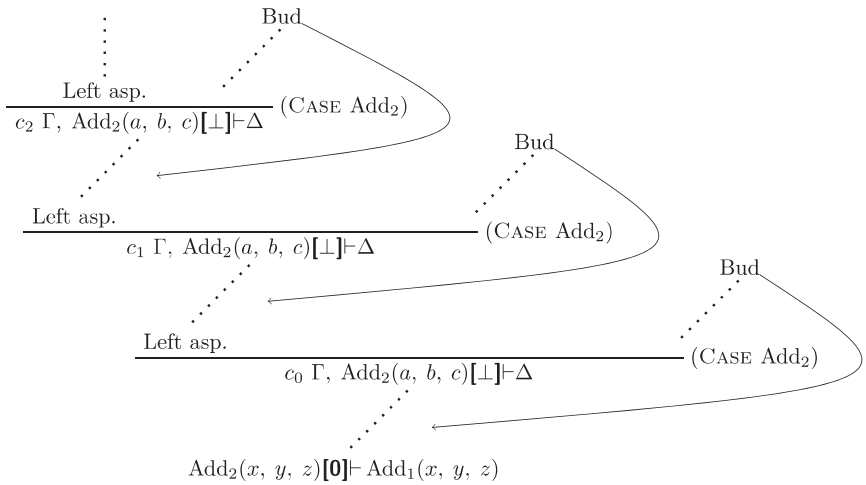

図 8　$(c_i)_{i\in\mathbb{N}}$ の構成

以上の準備の下，$\mathcal{D}_{\mathrm{cf}}$ 上に無限個のノードがあることを示し，矛盾を導く．

定理 10(2) の証明概略． $\mathcal{D}_{\mathrm{cf}}$ の根から伸びる一番右側の道は index path であることから，一番右側の道には bud があることがまずわかる．また，この道の上には switching point が存在することもわかる．そのような switching point のうち，一番下のノードを c_0 とする．

さて，c_0 の左側の仮定は index sequent であるから，c_0 の左側の仮定から始まる一番右側の道には bud がありしかも index path である．よって，そこには switching point が存在する．そのような switching point の中で一番下のノードを c_1 とする．

c_1 の左側の仮定は index sequent である．すると，c_1 の左側の仮定から始まる一番右側の道は bud がありしかも index path である．よって，そこには switching point が存在する．そのような switching point の中で一番下のノードを c_2 とする．ここまでの状況を 図 8 に図示した．

このようなプロセスを繰り返すと，無限の頂点が得られる．しかし，$\mathtt{CLKID}^\omega$ の証明に出現する頂点は有限なのでこれは矛盾である． □

以上のことから，$\mathrm{Add}_2(x,\ y,\ z)\vdash\mathrm{Add}_1(x,\ y,\ z)$ がカットなしでは $\mathtt{CLKID}^\omega$ を証明できないことがわかる．ゆえに，カットは $\mathtt{CLKID}^\omega$ において除去可能でない．より詳細な証明は [11] を参照されたい．

5　おわりに

$\mathtt{CLKID}^\omega$ で証明可能であるが，カットなしでは証明できないシークエントを与えた．それゆえ，$\mathtt{CLKID}^\omega$ においてカット除去は可能でない．

循環証明体系において，「あるシークエントがカット無しで証明できないこと」の現在知られている証明に共通しているテクニックは，循環証明体系の証明に出現するシークエントの有限性を使っていることである [9, 13, 8, 11, 12]．ところで，$\mathtt{CLKID}^\omega$ において証明図の形を無限の高さの木

に拡張した体系である $\mathtt{LKID}^\omega$ はカット除去可能であることが知られている [3]．これは一階算術の通常の証明体系において，帰納法の代わりに ω 規則という無限規則を導入した場合にカット除去が成り立つこと（たとえば [10, 16] を見よ）と「有限性を捨てると，カット除去できるようになる」という点で類似していることは興味深い．そういったことから，「有限性とカット除去」の関係に注意を払っている．

つい最近，帰納的述語を単項に制限しても $\mathtt{CLKID}^\omega$ のカット除去性は成り立たないことがわかった [12]．それでは，$\mathtt{CLKID}^\omega$ にどのような制限をすればカット除去性が成り立つのだろうか？先に触れた「有限性とカット除去」という観点から，「有限ステップで真偽判定のできる」つまり決定可能であるプレスバーガー算術に対する循環証明体系はカット除去可能ではないかと予想している．

逆に $\mathtt{CLKID}^\omega$ の証明図の形のクラスを拡張したらカット除去性が成立するようになることが知られている．先にも触れたが，証明図の形を無限の高さの木に拡張した体系である $\mathtt{LKID}^\omega$ はカット除去可能である．実は，特に証明図の形を Recursive tree に制限しても証明能力は同等かつカット除去可能であることもわかっている [3]．無限木のクラスという観点からすると，$\mathtt{CLKID}^\omega$ は $\mathtt{LKID}^\omega$ の証明図の形を Regular tree に制限した体系ととらえることもできる．では，$\mathtt{LKID}^\omega$ の証明図の形の制限としてどのような無限木のクラスに制限した場合であれば，カット除去は可能になるのであろうか？ Regular tree と Recursive tree の間のクラスでそのような無限木のクラスは存在するのだろうか？このことを明らかにすることで，「有限性とカット除去の関係」をよりはっきりした言葉で表現できるようになることを期待している．

参考文献

[1] B. Afshari and G. E. Leigh. Cut-free completeness for modal mu-calculus. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pp. 1–12, June 2017.

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

[3] J. Brotherston and A. Simpson. Sequent calculi for induction and infinite descent. *Journal of Logic and Computation*, Vol. 21, No. 6, pp. 1177–1216, 2011.

[4] James Brotherston. Cyclic Proofs for First-Order Logic with Inductive Definitions. In Bernhard Beckert, editor, *Automated Reasoning with Analytic Tableaux and Related Methods*, pp. 78–92. Springer Berlin Heidelberg, 2005.

[5] James Brotherston, Nikos Gorogiannis, and Rasmus L. Petersen. A Generic Cyclic Theorem Prover. In *Programming Languages and Systems*, Vol. 7705, pp. 350–367. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[6] Kai Brünnler and Martin Lange. Cut-free sequent systems for temporal logic. *The Journal of Logic and Algebraic Programming*, Vol. 76, No. 2, pp. 216 – 225, 2008.

[7] Samuel R Buss. An introduction to proof theory. *Handbook of proof theory*, Vol. 137, pp. 1–78, 1998.

[8] Anupam Das. On the logical complexity of cyclic arithmetic. *Logical Methods in Computer Science*, Vol. 16, No. 1, 2020.

[9] Daisuke Kimura, Koji Nakazawa, Tachio Terauchi, and Hiroshi Unno. Failure of cut-elimination

in cyclic proofs of separation logic. *Computer Software*, Vol. 37, No. 1, pp. 39–52, 2020.

[10] Georg Kreisel. A survey of proof theory. *The Journal of Symbolic Logic*, Vol. 33, No. 3, pp. 321–388, 1968.

[11] Yukihiro Masuoka, James Brotherston, and Makoto Tatsuta. Counterexample to cut-elimination in cyclic proof system for first-order logic with inductive definitions. *arXiv:2106.11798*, June 2021.

[12] Yukihiro Masuoka and Daisuke Kimura. A study for recovering the cut-elimination property in cyclic proof systems by restricting the arity of inductive predicates. *Proceedings of PPL2022*, 2022.

[13] Kenji Saotome, Koji Nakazawa, and Daisuke Kimura. Failure of cut-elimination in the cyclic proof system of bunched logic with inductive propositions. In *6th International Conference on Formal Structures for Computation and Deduction, FSCD 2021*, Vol. 195 of *LIPIcs*, pp. 11:1–11:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[14] Daniyar Salkarbekovich Shamkanov. Circular proofs for the gödel-löb provability logic. *Mathematical Notes*, Vol. 96, No. 3-4, pp. 575–585, 2014.

[15] Makoto Tatsuta, Koji Nakazawa, and Daisuke Kimura. Completeness of Cyclic Proofs for Symbolic Heaps with Inductive Definitions. In Anthony Widjaja Lin, editor, *Programming Languages and Systems*, pp. 367–387. Springer International Publishing, 2019.

[16] Anne Sjerp Troelstra and Helmut Schwichtenberg. *Basic proof theory, Second Edition*, Vol. 43 of *Cambridge tracts in theoretical computer science.* Cambridge University Press, 2000.