Title	非古典論理の証明論 - 直観主義部分構造論理の自然演繹体系と証明支援システムの構築
Author(s)	毛利, 元彦
Citation	
Issue Date	2002-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/922
Rights	
Description	Supervisor:小野 寛晰, 情報科学研究科, 博士

非古典論理の証明論 — 直観主義部分構造論理の自然演繹体系と証明支援システムの構築

北陸先端科学技術大学院大学
毛利元彦

目 次

序章

証明論とは形式化された数学的議論や数学の証明、すなわち証明図に対する数学的な研究である。与えられた論理式が正しいときにはその論理式に対する証明図を構成することができるが、その証明図は唯一に定まるとは限らない。一般には一つの論理式に対して無限パターンの証明図を構成することができる。そのように与えられた論理式に対して様々な証明図を取ることができるが、それらの証明図に論理式が証明可能であること以上の意味を見出すことはないのであろうか。

証明論における代表的な議論としてはシーケント計算の体系における cut の除去が挙げられるであろう。古典論理の形式的体系としては LK があるが、この LK では cut 規則があるために与えられた論理式に対して無限パターンの証明図を構成することができる。与えられた論理式が正しいことは、実際に証明図を構成することにより確かめらることができる。それに対し、その論理式が正しくないことを確認するためには無限にある証明図のどれもがその論理式を導出しないことをチェックしなければならないことになる。これは事実上不可能である。しかしながら、cut 除去定理により cut を一つも含まない有限個の証明図 (実際には contraction や exchange の繰り返しの適用を押さえる工夫も必要になる) を考えれば十分であり、正しい論理式と正しくない論理式の区別が可能となる。このことから決定可能性が導かれる。

シーケント計算における cut のない証明図に対応するのが、自然演繹体系における正規な証明図である。自然演繹体系において正規な証明図を得るための基本的なステップはリダクションと呼ばれる。このリダクションはラムダ計算におけるラムダ項のリダクションにうまく対応させることができる。そのため計算ステップと密接に対応しソフトウェア科学の立場からも関心が持たれている。

自然演繹体系のリダクションに関する基本的な問いとして次のようなものを挙げることができる。

1. リダクションを続けて行うことにより、必ず正規な証明図を得ることができるのだろうか (弱正規性、停止性)
2. どのようにリダクションンを行っても、必ず正規な証明図が得られるのだろうか (強正規性)
3. 与えられた証明図からリダクションを行って得られる正規な証明図はただ一通りに決まるのだろうか (一意性)
4. どのようにリダクションを行なうと最も効率よく正規な証明図が得られるのだろうか (最短リダクション戦略)

部分構造論理や、あるタイプのラムダ項に対するこれらの問いへの解答を本論文の 1 章及び 2 章で与えた。

上述したように cut 除去定理の成り立つようなシーケント計算においては、多くの場合決定可能になる。しかし理論的に決定可能性を得ることと計算機上に定理自動証明システムとして実装することの間には多きな隔たりがある。さらにこれまでの定理自動証明システムの多くは、文字通り定理であるか否かを自動的に行うにすぎず、論理学の研究を補助するためのツールになっていない。これらの点を考慮に入れた上で、3 章で様相論理 S4 に対する証明反駁アルゴリズムを与え、4 章でそのアルゴリズムの実装を行なった。3 章で展開する S4 の証明反駁アルゴリズムは次のような特徴を持つ。

- 通常の S4 の証明探索においては繰り返し現れるシーケントを排除するために、これまでの探索中に現れたシーケントと全て比較しなければならず、効率が悪い。この問題を避けるための工夫を行った。
- 証明可能なときには「証拠」として証明図を出力する。

- 与えられたシーケントの証明探索に失敗した場合、それまでに得られた探索木はそのシーケントが証明不可能である「証拠」となっている。この「証拠」を利用して、そのシーケントを反駁するモデルが実際に構成できることを示した。本論文で示した方法により得られる反例はクラスターのなす木構造をしており、さらに従来の方法に比べてより効率的なアルゴリズムからなる。

これらのアルゴリズムの提示とその正当性を示すためには、シーケントの書き換えといった証明図に対する操作が必要となるのである。

このように証明論は証明図に焦点を当てた研究分野であるが、そこで本論文では次のように大きく 4 つの研究を行なった。

1. 直観主義部分構造論理の自然演繹体系
2. ラムダ項の展開と BCK–ラムダ項における最短リダクション戦略
3. 証明反駁アルゴリズム
4. 証明支援システム xpe

付録では、4 章で説明する xpe のマニュアルをあげておく。この xpe では 3 章の証明反駁アルゴリズムの実装が行われている。
以下それぞれにの内容ついて概要を説明する。

1. 構造規則には *exchange*,*weakening*,*contraction* の 3 つがあり、Gentzen が導入した LK で用いられている。これら構造規則は論理結合子には直接関与せず、推論にどのように関与しているのかはっきりしない部分がある。そこで近年この構造規則の役割を明らかにするために部分構造論理という論理の枠組が提唱され、様々な角度から構造規則の研究がなされてきた。
本章ではシーケント計算の体系によって定義されている直観主義部分構造論理に対して自然演繹体系による形式化を行い、その形式的意味を明かにする。
こうした部分構造論理に対する自然演繹体系はいくつか知られているが、そのいずれもが構造規則を用いたシーケント計算に近い形で形式化された自然演繹体系であり、本当の意味での自然演繹体系とは呼び難い。そこで本章では構造規則が自明には現れない直観主義部分構造論理の自然演繹体系を導入し、さらにその強正規性の証明を与える。この強正規性の証明を全ての構造規則を含んだ体系に対して示すことによって、他の弱い体系に対する強正規性を得ることができるが、contraction のない体系に対してはより構成的な別証明を与えることができる。実際、これらの体系においてはリダクションの回数の上限を具体的に与えることができる。

2. 証明図の計算的性質を明らかにするため、本章ではラムダ項のリダクション戦略について考えていく。カリーハワード同型対応により型付きラムダ項と証明図は一対一に対応するため、ラムダ項は自然演繹体系と密接な関係にある。
ラムダ項を含むリダクション系ではリダクションが一意に定まらないため、常に次の根源的な問いがある。

- どのようにリダクションを行うと、最短となるか。
- どのようにリダクションを行うと、最長となるか。

前者の問いに対する答えを最短リダクションといい、後者の問いに対する答えを最長リダクション戦略という。
ラムダ項の最長のリダクション戦略に関しては、de Vrijer によってほとんどの場合 F_∞ 戦略が最長となることが知られている。では、最短となるのはどのような場合であろうか。
ラムダ項では一般的にはこの問題は否定的に解かれている。すなわち、ラムダ項では帰納的に最短リダクション戦略を求めることができないということが示されてている。

そこで本章では、二つのタイプの制限を加えたラムダ項について最短リダクション戦略を与える。ラムダ項の展開においては偽値呼び出しリダクション戦略が、また先の contraction のない体系に対応した BCK–ラムダ項においては最左リダクション戦略がそれぞれ最短となることが示される。

3. 本章では証明図とモデルとの関係について眺めていく。一般的に証明図が存在することと任意のモデルで妥当であることは同値である。この性質を完全性という。しかしながら証明図とモデルが具体的にどのように関係しているのかはあまり明確にされていない。完全性の証明では極大無矛盾な対を作るために証明可能か否かという判断が必要であるが、その判断アルゴリズムを与えているのではないため、具体的な証明図との関係は希薄なものとなる。また、無限個の論理式を含むシーケントについて判断するため、構成的な関係は見い出せない。

本論文で与えた決定手続きでは、証明可能なときには証明図が、証明不可能なときには反例が具体的に構成できるようになっている。このようなアルゴリズムを証明反駁アルゴリズムという。論理を研究する上では、証明可能なときの証明図、及び証明不可能なときの反例の分析が重要であるため、証明反駁アルゴリズムの提案は有用であると考えられる。

この証明反駁アルゴリズムが与えられれば次の性質が直ちに得られる。

- 決定可能性
- cut 除去性
- 完全性
- 有限モデル性

こうした論理学上重要な性質が一度に得られるという優れた手法であると思われる。

古典命題論理、直観主義命題論理、様相論理 K,KT では効率のよい証明反駁アルゴリズムが既に知られているが、本章で様相論理 S4 に対しても効率のよい証明反駁アルゴリズムが与えられる。

4. TEX で証明図を書くには通常は龍田真氏の proof.sty を用いて書かれる。ところが、proof.sty では証明図という木構造を括弧 $\{\cdots\}$ を用いて表現するため、証明図が複雑になってくると括弧が入れ子になってしまい、可読性が著しく落ちてしまう。

この問題を解決するために、xpe(X window system Proof Editor) を開発した。これは、TEX のコマンドを解釈・表示し、出来合いの証明図を見ながら書けるようなソフトウェアである。

このままでは単なる証明図作成ソフトに過ぎないが、さらに定理自動証明及び証明反駁アルゴリズムを実装することにより証明支援システムとして発展させた。一般的な定理自動証明システムでは証明可能か否かに重点が置かれ証明図や反例の出力はおざなりにされがちである。これでは論理学の研究を進めるためのツールとはなり得ない。そこで、証明図の出力を主眼にした定理自動証明を実装した。実際に多くの基本的な部分構造論理の定理自動証明が可能である。さらに上にあげたの体系の証明反駁アルゴリズムを実装し、証明不可能なときには反例をも出力する。付録としてこの xpe のマニュアルを巻末に添付するので、xpe の詳細はこちらを参照して欲しい。

第1章　直観主義部分構造論理の自然演繹体系

1.1　始めに

直観主義部分構造論理は直観主義論理の形式的体系 LJ から全てまたは一部の構造規則を取り除いたものとして形式化される。線型論理 [7] はこれらの部分構造論理の特別な場合である。exchange があり weakening と contruction がない部分構造論理が線型論理である。

部分構造論理 (線型論理も含む) の重要な性質として、論理積 ($\wedge$) が二つに分裂することが上げられる。一つは加法的論理積 ($\wedge$)、もう一つは乗法的論理積 ($*$) である。これまでになされてきた部分構造論理の自然演繹体系に関する研究では、そのほとんどが乗法的論理積についてのみ言及されていて ([3], [34])、加法的論理積については論じられていない。これは、加法的論理積も同時に取り扱うことがある種の難しさを持っているからである。同時に扱った論文としては [16] があるが、言及されているのは線型論理のみであり、部分構造論理を統一的に扱うものではない。また、型付きラムダ項から得られる証明図が一意には定まらないという、unique derivation が成り立たない体系である。そこで本論文では、この二つの論理積を扱うと同時に、統一的に部分構造論理を扱える体系の導入をその主な目的とする。

直観主義部分構造論理はもともとシーケントスタイルの体系により定義されてきた。例えば、図 1.1 の左上のように weakening 規則が明示的に用いられている。これを普通のスタイルの自然演繹の体系で書くとどうなるであろうか。おそらく、図 1.1 の右上のようになるであろう。こうなるともはや本当の意味での自然演繹の体系ではないのではなかろうか。というのは、明示的には構造規則が現れないことが自然演繹体系の一つの特徴であると考えられるからである。次に contraction の場合を考えてみよう。これをシーケントスタイルで書くと図 1.1 の左下になる。これを普通のスタイルで書くと右下の二つの規則のようになるであろうか。ここまでくるともはや自然演繹とはいえないのではなかろうか。

そこで本論文では、より自然な部分構造論理の自然演繹の体系を考える。以下に見るようにこの体系では直観主義部分構造論理を統一的な視点で捉えることが可能になる。その結果、構造規則の自然演繹体系での役割が明らかになってくる。

そして、これらの体系の強正規性を調べる。強正規性は Tait[29] のアイディアに基く可約集合 (reducible set) の概念を用いて証明できる。しかしながら、contruction の無い体系では強正規性のより簡単な証明を与えることができる。それは、自然演繹の体系に対応したラムダ項から自然数への関数を定義し、その値がリダクションを行うと小さくなるという証明である。つまり、その関数がリダクション回数の上限を示すことになる。

本論文では、まず論理結合子を含意のみに制限した場合の自然演繹体系を考える。これに対応したラムダ項を導入

$$\frac{\Gamma \vdash B}{A, \Gamma \vdash B}\ (Weakening) \qquad \frac{A \quad \begin{array}{c}\Gamma \\ \vdots \\ B\end{array}}{B}\ (Weakening)$$

$$\frac{A, A, \Gamma \vdash B}{A, \Gamma \vdash B}\ (Contraction) \qquad \frac{\begin{array}{c}\not A, A, \Gamma \\ \vdots \\ B\end{array}}{B} \quad \frac{A \qquad \begin{array}{c}\not A, \not A, \Gamma \\ \vdots \\ B\end{array}}{B}\ (Contraction)$$

図 1.1: シーケントスタイルと普通の自然演繹スタイルの weakening と contraction

し、強正規性について調べる。contraction の無い体系ではリダクションを行うといわゆるラムダ項の長さが真に短くなるため、強正規性が容易に得られる。(1.4.5 節参照)

次に、加法的論理積と乗法的論理積の二つの論理積を用いて体系を拡張する。そして対応するラムダ項も拡張する。このラムダ項に関して強正規性を調べることで、元の体系の強正規性もいえることになる。contraction の無い体系について、別証明を検討する。含意のみの場合とは異なり、いわゆる長さが短くなるとは限らないが、それでもリダクションの上限を示す関数を定義することで、強正規性が成り立つことを述べる。

1.2 FL

本節では 8 つの直観主義部分構造論理を定義する。その基本となるのは、FL(full Lambek logic) と呼ばれる体系である。FL では構造規則を一つも含まないが、FL に 3 つの構造規則を加えていくことで全部で 8 つの直観主義部分構造論理を定義する。

我々の考える体系は、論理結合子として含意 ($\supset$) と乗法的論理積 ($*$) と加法的論理積 ($\wedge$) の 3 つを含んだものである。まず論理式から定義する。

定義 1.2.1 (論理式) 無限または有限の命題変数が与えられているとする。

1. P : 命題変数 $\Rightarrow$ P : 論理式

2. A, B : 論理式 $\Rightarrow$ $(A \supset B), (A \wedge B), (A * B)$: 論理式

定義 1.2.2 (シーケント)

$A_1, \cdots, A_n, B_1, \cdots, B_m\,(n, m \geq 0)$: 論理式 $\Rightarrow$ $A_1, \cdots, A_n \to B_1, \cdots, B_m$: シーケント

次に推論規則を定義する。次の図 1.2 に示されているのが、FL の推論規則である。推論規則 (cut) よりも上にある規則、すなわち、(axiom)、$\supset$ に関する規則、二つの論理積 ($*$,$\wedge$) に関する規則、そして (cut) までが FL を定義する。

次に、FL に構造規則を加えて得られる体系を構造規則の頭文字を FL に付け加えて表記する。例えば FL_{ec} は FL に exchange と contraction 規則を加えて得られる体系である。線型論理は FL に exchange を加えて得られる体系であるから FL_e となる。

$$A \vdash A\,(Axiom)$$

$$\frac{\Gamma \vdash A \quad \Delta, B, \Sigma \vdash C}{\Delta, A \supset B, \Gamma, \Sigma \vdash C}\,(\supset L) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}\,(\supset R)$$

$$\frac{\Gamma, A, \Delta \vdash C}{\Gamma, A \wedge B, \Delta \vdash C}\,(\wedge L_l) \qquad \frac{\Gamma, B, \Delta \vdash C}{\Gamma, A \wedge B, \Delta \vdash C}\,(\wedge L_r)$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\,(\wedge R)$$

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, A * B, \Delta \vdash C}\,(*L) \qquad \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Delta, \Gamma \vdash A * B}\,(*R)$$

$$\frac{\Gamma \vdash A \quad \Delta, A, \Sigma \vdash B}{\Delta, \Gamma, \Sigma \vdash B}\,(Cut)$$

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}\,(Exchange)$$

$$\frac{\Gamma, \Delta \vdash B}{\Gamma, A, \Delta \vdash B}\,(Weakening)$$

$$\frac{\Gamma, A, A, \Delta \vdash B}{\Gamma, A, \Delta \vdash B}\,(Contraction)$$

図 1.2: FL の推論規則と構造規則

この二つの論理積は直観的にはどのような意味の違いがあるのであろうか。このことを考えるために、次の表を眺めてみよう。

	FL,FL_e	FL_w,FL_{ew}	FL_c,FL_{ec}	FL_{wc},FL_{ecw}
$A \wedge B \vdash A$	◯	◯	◯	◯
$A \wedge B \vdash B$	◯	◯	◯	◯
$A * B \vdash A$	×	◯	×	◯
$A * B \vdash B$	×	◯	×	◯
$A \wedge B \vdash A * B$	×	×	◯	◯
$A * B \vdash A \wedge B$	×	◯	×	◯

表 1.1: 二つの論理積に関する証明可能性

FL と FL_e では証明できるのは $A \wedge B \vdash A$ と $A \wedge B \vdash B$ のみである。これは $A \wedge B$ が成り立っているときには、A か B の好きなほうを手に入れることができるが、両方同時に手に入れることはできないことを意味する。$A \wedge B$ が成り立っているのきには、我々には A のみか B のみの 2 通りの選択肢しかない。

そして、$A * B$ が成り立っているときには、たとえ片方が不必要であっても、A と B の両方を使う必要がある。$A * B$ が成り立っているときには、我々には A と B の両方を使うという 1 通りの選択肢があることになる。

FL_w と FL_{ew} では、さらに $A * B \vdash A$, $A * B \vdash B$, $A * B \vdash A \wedge B$ が証明可能になる。つまり $\wedge$ の意味は変わらないが、$*$ の意味が次のように変わることになる。それは、$A * B$ が成り立つときには、A と B のうち 1 つまたは 2 つを好きなように手に入れることができる。つまり $A * B$ が成り立つときには、我々には A のみ、B のみ、A と B の両方の 3 通りの選択肢があることになる。

では、残りの体系ではそれぞれの論理積の直観的意味はどのようになるであろうか。その答えを考えてみるのもよいであろう。

1.3　含意に制限した場合の自然演繹体系 $\mathrm{NFL}_{\supset}$

1.3.1　体系 $\mathrm{NFL}_{\supset}$ の形式化

まず、含意のみに制限した場合の自然演繹体系を考えていく。この場合自然演繹体系は間接的にラムダ項を通してすでに議論されている。例えば FL_e に対応するラムダ項は、BCI ラムダ項と呼ばれ議論されているし、FL_{ec} に対応したラムダ項はラムダ–I 項として議論されてきている。

この節ではいままでに議論されてきた事柄をまとめる意味で、含意のみの体系について議論していく。強正規性は [8] や [12] ですでに示されている。もともとの NJ の自然演繹体系については Prawitz の [28] を参照せよ。

定義 1.3.1 (assumption)
A:論理式, n:自然数 $\Rightarrow$ A^n:仮定

なぜ、自然数を付加したものを仮定とするのであろうか。普通の自然演繹の体系では、仮定は論理式の集合である。しかしながら、部分構造論理では同じ論理式を違うものとして扱う必要があるため、仮定を単なる集合として扱うことはできない。そのため同じ論理式とある時には同じものとみなしたり、またあるときには違うものとみなせるように、論理式に自然数を付加したものを仮定とするのである。

こうすれば、A^n と A^n は同じ論理式とみなせるし、n と m が異なるときには A^n と A^m を違う論理式とみなすことができる。この自然数のラベル付きの論理式を用いて、各部分構造論理に対応する自然演繹体系を定義していく。

次のようにして、仮定に順序を導入しておく。

定義 1.3.2 (仮定に関する半順序) 以下のように、仮定に半順序を導入する。

1. $A^n = B^m \Leftrightarrow A \equiv B$ かつ $n = m$
2. $A^n < B^m \Leftrightarrow n < m$
3. $A^n \leq B^m \Leftrightarrow n < m$ あるいは $A^n = B^m$

(注意) $A \not\equiv B$ のときは、A^n と B^n の間に順序は付かない。

そして、この順序を用いて仮定の集合上にも順序を入れる。

定義 1.3.3 (仮定の集合への順序の拡張) Γ,Δ を仮定の集合とする。

1. $\Gamma < \Delta \Leftrightarrow \forall A \in \Gamma, \forall B \in \Delta (A < B)$
2. $\Gamma \leq \Delta \Leftrightarrow \forall A \in \Gamma, \forall B \in \Delta (A \leq B)$

このとき、$\Gamma \leq \Delta$ かつ $\Delta \leq \Gamma$ ならば $\Gamma = \Delta$ であるが、逆は成り立たない。それは、$\Gamma = \Delta = \{A^m, B^n\} (m \neq n)$ とすれば明らかである。

定義 1.3.4 (体系 $\mathrm{NFL}_{\supset}$) A, B, C を論理式を動くメタ変数とし、Γ, Δ を仮定の有限集合を動くメタ変数とする。このとき、*NFL* の証明図と仮定の有限集合と論理式の間の関係 $\vdash$ を同時に定義する。

(assumption) A^n が仮定 $\Rightarrow$ A^n は証明図であり $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W 条件 : $A^n \in \Gamma$、E 条件 : $\Gamma \leq \{A^n\}$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}\ (\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

(⊃ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

これらの証明図はそれぞれ以下のように図示される。

$$A^n\text{:(Axiom)}$$

$$\begin{array}{c} \text{W:}A^n \in \Gamma \\ \text{E} : \Gamma \leq \{A^n\} \\ \Gamma - \{A^n\} \\ \vdots \\ \dfrac{B}{A \supset B}\ (\supset I)^n \end{array} \qquad \begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E} : \Gamma \leq \Delta \\ \Gamma \qquad \Delta \\ \vdots \qquad \vdots \\ \dfrac{A \supset B \quad A}{B}\ (\supset E) \end{array}$$

図 1.3: NFL の推論規則

NFL(以下添字の $_\supset$ は省略する) の定義では、E 条件,C 条件,W 条件という 3 つの適用条件がある。それぞれ、

E 条件 $\cdots$ Exchange 規則

C 条件 $\cdots$ Contraction 規則

W 条件 $\cdots$ Weakening 規則

に対応した条件である。この NFL を用いて残りの 7 つの体系を定義する。NFL に Weakening 規則を加えたものを NFL$_w$ とすると、これは NFL から W 条件を取り除いたものである。ほかの体系も同様である。念のため定義を一通り述べておく。

定義 1.3.5 (体系 NFL$_w$) (assumption) A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

(⊃ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で E 条件 : $\Gamma \leq \{A^n\}$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

(⊃ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.6 (体系 NFL$_c$) (assumption) A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

(⊃ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W 条件 : $A^n \in \Gamma$、E 条件 : $\Gamma \leq \{A^n\}$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

(⊃ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。E 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.7 (体系 NFL$_{cw}$) (assumption) A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

(⊃ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で E 条件 : $\Gamma \leq \{A^n\}$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。E 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.8 (体系 $\mathbf{NFL}_e$) **(assumption)** A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W 条件 : $A^n \in \Gamma$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.9 (体系 $\mathbf{NFL}_{ew}$) **(assumption)** A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図であるとする。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.10 (体系 $\mathbf{NFL}_{ec}$) **(assumption)** A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W 条件 : $A^n \in \Gamma$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

定義 1.3.11 (体系 $\mathbf{NFL}_{ecw}$) **(assumption)** A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図であるとする。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}$ $(\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}$ $(\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

これらの体系のうち、体系 NFL_{ecw} は Gentzen の NJ の部分体系である。これらの体系が対応する FL を拡張したシーケント計算と同値なことを証明していく。

補題 1.3.12 (仮定の自然数の付け替え) 体系 NFL において、$\mathcal{D}$ が $\Gamma, A_1^{n_1}, \cdots, A_k^{n_k} \vdash B$ を証明する証明図であるとき、$\Gamma, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \vdash B$ $(m \geq 0)$ を証明する $\mathcal{D}$ と同じ高さの証明図 $\mathcal{E}$ が存在する。

(証明) 証明図の高さに関する帰納法。

1. $\mathcal{D} \equiv A^n$ のとき。$\mathcal{E} \equiv A^{n+m}$ を取ればよい。

2. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{B \supset A}\ (\supset I)^n$ で、$\Gamma, A_1^{n_1}, \cdots, A_k^{n_k} \vdash B \supset A$ を証明するとき。証明図は、

$$\begin{array}{c} \Gamma, A_1^{n_1}, \cdots, A_k^{n_k} \\ \vdots \\ \dfrac{B}{A \supset B}\ (\supset I)^n \end{array}$$

という形をしていて、帰納法の仮定より、

$$\begin{array}{c} \Gamma, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \\ \vdots \\ B \end{array}$$

なる証明図が存在する。よって $\mathcal{E}$ として、

$$\begin{array}{c} \Gamma, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \\ \vdots \\ \dfrac{B}{A \supset B}\ (\supset I)^n \end{array} \quad \text{あるいは} \quad \begin{array}{c} \Gamma, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \\ \vdots \\ \dfrac{B}{A \supset B}\ (\supset I)^{n+m} \end{array}$$

を取ればよい。

3. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}\ (\supset E)$ で、$\Gamma, A_1^{n_1}, \cdots, A_k^{n_k} \vdash B$ を証明するとき。証明図は、

$$\dfrac{\begin{array}{cc} \Gamma, A_1^{n_1}, \cdots, A_l^{n_l} & A_{l+1}^{n_{l+1}}, \cdots, A_k^{n_k} \\ \vdots & \vdots \\ A \supset B & A \end{array}}{B}\ (\supset E) \quad \text{あるいは} \quad \dfrac{\begin{array}{cc} \Gamma_1 & \Gamma_2, A_1^{n_1}, \cdots, A_k^{n_k} \\ \vdots & \vdots \\ A \supset B & A \end{array}}{B}\ (\supset E)$$

という形をしていて、帰納法の仮定より、

$$\begin{array}{c} \Gamma, A_1^{n_1}, \cdots, A_l^{n_l} \\ \vdots \\ A \supset B \end{array} \ \text{と}\ \begin{array}{c} A_{l+1}^{n_{l+1}}, \cdots, A_k^{n_k} \\ \vdots \\ A \end{array} \quad \text{あるいは} \quad \begin{array}{c} \Gamma_1 \\ \vdots \\ A \supset B \end{array} \ \text{と}\ \begin{array}{c} \Gamma_2, A_1^{n_1}, \cdots, A_k^{n_k} \\ \vdots \\ A \end{array}$$

なる証明図が存在する。よって $\mathcal{E}$ として、

$$\dfrac{\begin{array}{cc} \Gamma, A_1^{n_1+m}, \cdots, A_l^{n_l+m} & A_{l+1}^{n_{l+1}+m}, \cdots, A_k^{n_k+m} \\ \vdots & \vdots \\ A \supset B & A \end{array}}{B}\ (\supset E) \quad \text{あるいは} \quad \dfrac{\begin{array}{cc} \Gamma_1 & \Gamma_2, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \\ \vdots & \vdots \\ A \supset B & A \end{array}}{B}\ (\supset E)$$

を取ればよい。

この補題を用いて、シーケントで定義された部分構造論理と、自然演繹での体系が同値であることを示す。その証明は、まず最も大きな体系、すなわち FL_{ecw} と NFL_{ecw} との同等性を示す。そして、他の体系はこの定理の系として得られる。

定理 1.3.13 ($\mathbf{FL}_{ecw}$ と $\mathbf{NFL}_{ecw}$ の同等性) FL_{ecw} と NFL_{ecw} は同値である。すなわち FL_{ecw} で $A_1, ..., A_k \vdash A$ を証明する証明図 $\mathcal{D}$ が存在するとき、及びそのときに限って NFL_{ecw} で $\{A_1^{n_1}\} \cup \cdots \cup \{A_k^{n_k}\} \vdash A$ を証明する証明図 $\mathcal{E}$ が存在する。但し $A_1^{n_1} < \cdots < A_k^{n_k}$。

(証明) 証明図の構成に関する帰納法。

まず、$\mathrm{FL}_{ecw} \Rightarrow \mathrm{NFL}_{ecw}$ を示す。

1. $\mathcal{D} \equiv A$ のとき。$\mathcal{E} \equiv A^n$ を取ればよい。

2. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{\Gamma \vdash A}\,(Cut)$ のとき。 定義から $\Gamma = \Delta, \Pi, \Sigma$ であり、$\mathcal{D}_1$ は $\Pi \vdash B$ を証明する証明図とし、また $\mathcal{D}_2$ は $\Delta, B, \Sigma \vdash A$ を証明する証明図とする。帰納法の仮定から、$\Pi' \vdash B$ を証明する証明図 $\mathcal{E}_1$ と $\Delta' \cup \{B^n\} \cup \Sigma' \vdash A$ を証明する証明図 $\mathcal{E}_2$ が存在する。

   前補題から $\mathcal{E}_1$ で Π' の添字を n だけ移動した $\Pi'^{+n} \vdash B$ を証明する証明図 $\mathcal{E}_1'$ が存在する。

   $$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_2}{\Sigma \supset A}\,(\supset I)}{B \supset \Sigma \supset A}\,(\supset I)^n \qquad \mathcal{E}_1'}{\Sigma \supset A}\,(\supset E) \qquad \Sigma'^{+m}}{A}\,(\supset E)$$

   とおけばこの $\mathcal{E}$ は $\Delta' \cup \Pi'^{+n} \cup \Sigma'^{+m} \vdash A$ を証明する証明図である。

3. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma, A, \Delta \vdash B}\,(We)$ のとき。定義から $\mathcal{D}_1$ は $\Gamma, \Delta \vdash B$ を証明する証明図である。帰納法の仮定から $\Gamma' \cup \Delta' \vdash B$ を証明する証明図 $\mathcal{E}_1$ が存在する。このとき、$\Gamma' < \{A^n\}$ をとり

   $$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_1}{\Delta \supset B}\,(\supset I)}{A \supset \Delta \supset B}\,(\supset I)^n \qquad A^n}{\Delta \supset B}\,(\supset E) \qquad \Delta'^{+n}}{B}\,(\supset E)$$

   とおけば $\mathcal{E}$ は $\Gamma' \cup \{A^n\} \cup \Delta'^{+n} \vdash B$ を証明する証明図である。

4. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma, A, \Delta \vdash B}\,(Co)$ のとき。定義から $\mathcal{D}_1$ は $\Gamma, A, A, \Delta \vdash B$ を証明する証明図である。帰納法の仮定から、$\Gamma' \cup \{A^n, A^n + 1\} \cup \Delta'^{+n+1} \vdash B$ を証明する証明図 $\mathcal{E}_1$ が存在する。但し $\Gamma' < \{A^n\}$。

   $$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_1}{\Delta \supset B}\,(\supset I)}{A \supset \Delta \supset B}\,(\supset I)^{n+1} \qquad A^n}{\Delta \supset B}\,(\supset E) \qquad \Delta}{B}\,(\supset E)$$

   とおけば $\mathcal{E}$ は $\Gamma' \cup \{A^n\} \cup \Delta' \vdash B$ を証明する証明図である。

5. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma \vdash A \supset B}\,(\supset I)^n$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma, A \vdash B$ を証明する証明図である。帰納法の仮定から $\Gamma' \cup \{A^n\} \vdash B$ を証明する証明図 $\mathcal{E}_1$ が存在する。このとき

   $$\mathcal{E} \equiv \dfrac{\mathcal{E}_1}{A \supset B}\,(\supset I)^n$$

   とおけば、$\mathcal{E}$ は $\Gamma' \vdash A \supset B$ を証明する証明図である。

6. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{\Delta, A \supset B, \Gamma, \Sigma \vdash C}\,(\supset E)$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma \vdash A$ を証明する証明図で、$\mathcal{D}_2$ は $\Delta, B, \Sigma \vdash C$ を証明する証明図である。帰納法の仮定から $\Gamma' \vdash A$ を証明する証明図 $\mathcal{E}_1$ と、$\Delta' \cup \{B^n\} \cup \Sigma' \vdash C$ を証明する証明図 $\mathcal{E}_1$ が存在する。また一方、前補題から $\Gamma'^{+n+1} \vdash A$ を証明する証明図 $\mathcal{E}_1'$ が存在する。このとき

   $$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_2}{\Sigma \supset C}\,(\supset I)}{B \supset \Sigma \supset C}\,(\supset I)^n \qquad \dfrac{(A \supset B)^n \quad \mathcal{E}_1'}{B}\,(\supset E)}{\Sigma \supset C}\,(\supset E) \qquad \Sigma'^{+m}}{C}\,(\supset E)$$

   とおけば、$\mathcal{E}$ は $\Delta' \cup \{(A \supset B)^n\} \cup \Gamma' \cup \Sigma \vdash C$ を証明する証明図である。

次に逆方向、すなわち $\mathrm{NFL}_{ecw} \Rightarrow \mathrm{FL}_{ecw}$ を示す。

1. $\mathcal{E} \equiv \dfrac{\mathcal{E}_1}{A \supset B}\,(\supset I)^n$ のとき。定義から、 $\Gamma' \cup \{A^n\} \cup \Delta' \vdash B$ を証明する証明図 $\mathcal{E}_1$ が存在するか、あるいは $A^n \notin \Gamma' \cup \Delta'$ であり $\Gamma' \cup \Delta' \vdash B$ を証明する証明図 $\mathcal{E}_1$ が存在する。帰納法の仮定から [(a) $\Gamma, A, \Delta \vdash B$ を証明する $\mathcal{D}_1$ が存在] あるいは [(b) $\Gamma, \Delta \vdash B$ を証明する $\mathcal{D}_1$ が存在] する。

(a) のとき $\mathcal{D} \equiv \dfrac{\dfrac{\mathcal{E}_1}{\overline{\overline{\Gamma, \Delta, A \vdash B}}}\,(Ex)}{\Gamma, \Delta \vdash A \supset B}\,(\supset R)$ とおき、

(b) のとき $\mathcal{D} \equiv \dfrac{\dfrac{\mathcal{E}_1}{\Gamma, \Delta, A \vdash B}\,(We)}{\Gamma, \Delta \vdash A \supset B}\,(\supset R)$ とおけば、$\mathcal{D}$ は $\Gamma, \Delta \vdash A \supset B$ を証明する証明図となる。

2. $\mathcal{E} \equiv \dfrac{\mathcal{E}_1 \quad \mathcal{E}_2}{B}\,(\supset E)$ のとき。定義から、$\Gamma \cup \Delta \vdash A \supset B$ を証明する証明図 $\mathcal{E}_1$ と、$\Delta \cup \Pi \vdash A$ を証明する証明図 $\mathcal{E}_2$ が存在する。帰納法の仮定より、$\Gamma', \Delta' \vdash A \supset B$ を証明する証明図 $\mathcal{D}_1$ と $\Delta', \Pi' \vdash A$ を証明する証明図 $\mathcal{D}_2$ が存在する。

$$\mathcal{D} \equiv \dfrac{\dfrac{\mathcal{D}_2 \quad \dfrac{\mathcal{D}_1 \quad \dfrac{A \vdash A \quad B \vdash B}{A \supset B, A \vdash B}\,(\supset L)}{\Gamma', \Delta', A \vdash B}\,(Cut)}{\Gamma', \Delta', \Delta', \Pi' \vdash B}\,(Cut)}{\overline{\overline{(\Gamma', \Delta', \Pi' \vdash B)}}}\,(Co, Ex) \quad ,$$

とおけば $\mathcal{D}$ は $\Gamma', \Delta', \Pi' \vdash B$ を証明する証明図となる。

そして FL の拡張体系と、対応すると NFL の拡張体系についてもそれらの同等性が証明できる。

定理 1.3.14 (FL の拡張体系と、対応する NFL の拡張体系の同等性) また、*FL* あるいは *FL* にいくつかの構造規則を加えた体系と *NFL* から対応する条件を取り除いて得られる体系が同値であることが証明できる。すなわち、*FL* あるいは *FL* にいくつかの構造規則を加えた体系で $A_1, ..., A_k \vdash A$ を証明する証明図 $\mathcal{D}$ が存在するとき、及びそのときに限って *NFL* から対応する条件を取り除いて得られる体系で $\{A_1^{n_1}\} \cup \cdots \cup \{A_k^{n_k}\} \vdash A$ を証明する証明図 $\mathcal{E}$ が存在する。但し $A_1^{n_1} < \cdots < A_k^{n_k}$。

証明は省略する。

以上のように、FL の拡張体系と対応する NFL の拡張体系は強さが同じであることが証明される。次に証明図のリダクションを考える。

1.3.2 NFL のレデックスとリダクション

正規化を考えるために、本節では証明図のレデックスとリダクションを考える。レデックスというのはいわば証明図の無駄な部分であり、これを減らそうとする操作がリダクションである。

定義 1.3.15 (証明図の代入) $\mathcal{E}$ を $\Gamma \vdash A$ を証明する証明図、A^n を仮定とする。$\mathcal{D}$ に出現する A^n を $\mathcal{E}$ で置き換える代入を $[\mathcal{E}/A^n]\mathcal{D}$ で表し $\mathcal{D}$ の高さに関して帰納的に定義する。

1. $\mathcal{D} \equiv A^n$ のとき。$[\mathcal{E}/A^n]A^n \equiv \mathcal{E}$。

2. $\mathcal{D} \equiv A^m$ $(n \neq m)$ のとき。$[\mathcal{E}/A^n]A^m \equiv A^m$。

3. $\mathcal{D} \equiv B^m$ $(B \not\equiv A)$ のとき。$[\mathcal{E}/A^n]B^m \equiv B^m$。

4. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{A \supset C}\,(\supset I)^n$ のとき。 $[\mathcal{E}/A^n]\mathcal{D} \equiv \mathcal{D}$。

5. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{D \supset C}\,(\supset I)^m$ $(D \not\equiv A)$ で、$D^m \notin \Gamma$ あるいは $A^m \notin \Delta'$ のとき。ただし、Δ' は $\mathcal{D}_1$ の仮定の集合とする。
$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]\mathcal{D}_1}{D \supset C}\,(\supset I)^m$。

6. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{D \supset C}\ (\supset I)^s$ $(D \not\equiv A)$ で、$A^m \in \Gamma$ でありかつ $A^m \in \Delta'$ のとき。ただし、$\mathcal{D}_1$ は $\Delta' \vdash C$ を証明するの証明図であるとする。$D^k \notin \Delta' \cup \Gamma$ となる D^k を取り、$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]([D^k/D^m]\mathcal{D}_1)}{A \supset B}\ (\supset I)^k$

7. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}\ (\supset E)$ のとき。$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]\mathcal{D}_1 \quad [\mathcal{E}/A^n]\mathcal{D}_2}{B}\ (\supset E)$

以上のように代入は、証明図 $\mathcal{D}$ に出現する仮定 A^n の上に、$\Gamma \vdash A$ を証明する証明図 $\mathcal{E}$ を乗せる操作である。

このように代入を定義したとき、FL_c と FL_{cw} 以外の体系では特定の条件下で代入に対して閉じている。

補題 1.3.16 (NFL と NFL$_w$ における代入) *体系 NFL と NFL$_w$ において以下のことが成立する。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$ かつ $\Gamma < \{A\}$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。いま、$\Gamma < \Delta$(すなわち、$\Gamma \cap \Delta = \emptyset$ かつ $\Gamma \leq \Delta$) が成り立つとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。*

補題 1.3.17 (NFL$_e$ と NFL$_{ew}$ における代入) *体系 NFL$_e$ と NFL$_{ew}$ において以下のことが成立する。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。いま、$\Gamma \cap \Delta = \emptyset$ が成り立つとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。*

補題 1.3.18 (NFL$_{ec}$ と NFL$_{ecw}$ における代入) *体系 NFL$_{ec}$ と NFL$_{ecw}$ において以下のことが成立する。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。このとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。*

(証明) 証明図の高さに関する帰納法。

では NFL_c と NFL_{cw} では代入はどうなるのであろうか。例えば次のような $\mathcal{D}$ と $\mathcal{E}$ に対して $[\mathcal{E}/A^2]\mathcal{D}$ 場合に代入がうまくいかない。

$$\mathcal{D} \equiv \dfrac{\dfrac{(A \supset A \supset B)^1 \quad A^2}{A \supset B}\ (\supset E) \qquad A^2}{B}\ (\supset E) \quad \mathcal{E} \equiv \dfrac{(C \supset A)^3 \quad C^4}{A}\ (\supset E)$$

として $[\mathcal{E}/A^2]\mathcal{D}$ の場合である。この 2 つの体系の場合はリダクションと合わせて命題 1.3.24 で再考する。

定義 1.3.19 (レデックス) $\mathcal{D}_1, \mathcal{D}_2$ を証明図として、証明図に次のような部分証明図が存在するとき、この部分証明図をレデックスと呼ぶ。

$$\dfrac{\dfrac{\mathcal{D}_1}{A \supset B}\ (\supset I)^n \qquad \mathcal{D}_2}{B}\ (\supset E)$$

定義 1.3.20 (リダクション) 証明図が上の形のレデックスを含むとき、その証明図を右辺の証明図で置き換えることをリダクションと呼ぶ。そのことを $\rhd$ を間に置いて次のように表現する。

$$\dfrac{\dfrac{\mathcal{D}_1}{A \supset B}\ (\supset I)^n \qquad \mathcal{D}_2}{B}\ (\supset E) \ \rhd [\mathcal{D}_2/A^n]\mathcal{D}_1$$

さらに $\mathcal{D} \rhd \mathcal{D}'$ のとき次の左辺を右辺で置き換えることもリダクションである。

$$\dfrac{\mathcal{D}}{A}\ (\supset I)^n \quad \rhd \quad \dfrac{\mathcal{D}'}{A}\ (\supset I)^n$$

$$\dfrac{\mathcal{D} \quad \mathcal{E}}{A}\ (\supset E) \quad \rhd \quad \dfrac{\mathcal{D}' \quad \mathcal{E}}{A}\ (\supset E) \qquad\qquad \dfrac{\mathcal{E} \quad \mathcal{D}}{A}\ (\supset E) \quad \rhd \quad \dfrac{\mathcal{E} \quad \mathcal{D}'}{A}\ (\supset E)$$

例 1.3.21 (リダクション)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(A\supset A\supset B)^1 \quad [A^2]}{A\supset B}(\supset E) \quad [A^2]}{B}(\supset E)}{A\supset B}(\supset I)^2 \quad \dfrac{[(C\supset A)^3] \quad C^4}{A}(\supset E)}{B}(\supset E)}{(C\supset A)\supset B}(\supset I)^3 \quad (C\supset A)^5}{B}(\supset E)$$

$$\rhd \quad \dfrac{\dfrac{\dfrac{\dfrac{(A\supset A\supset B)^1 \quad \dfrac{[(C\supset A)^3] \quad C^4}{A}(\supset E)}{A\supset B}(\supset E) \quad \dfrac{[(C\supset A)^3] \quad C^4}{A}(\supset E)}{B}(\supset E)}{(C\supset A)\supset B}(\supset I)^3 \quad (C\supset A)^5}{B}(\supset E)$$

$$\rhd \quad \dfrac{\dfrac{(A\supset A\supset B)^1 \quad \dfrac{[(C\supset A)^5] \quad C^4}{A}(\supset E)}{A\supset B}(\supset E) \quad \dfrac{[(C\supset A)^5] \quad C^4}{A}(\supset E)}{B}(\supset E)$$

NFL$_c$ と NFL$_{cw}$ を除く各体系がリダクションに関して閉じていることを次の補題で示す。

補題 1.3.22 (NFL,NFL$_e$,NFL$_{ec}$ のリダクション) $\mathcal{D}$ を *NFL* あるいは *NFL$_e$* あるいは *NFL$_{ec}$* の $\Gamma \vdash A$ を証明する証明図とする。いま $\mathcal{D} \rhd \mathcal{E}$ のとき、$\mathcal{E}$ もまた $\Gamma \vdash A$ を証明するその体系での証明図である。

補題 1.3.23 (NFL$_w$,NFL$_{ew}$,NFL$_{ecw}$ のリダクション) $\mathcal{D}$ を *NFL$_w$* あるいは *NFL$_{ew}$* あるいは *NFL$_{ecw}$* の $\Gamma \vdash A$ を証明する証明図とする。いま $\mathcal{D} \rhd \mathcal{E}$ のとき、ある $\Gamma' \subset \Gamma$ が存在して $\mathcal{E}$ もまた $\Gamma' \vdash A$ を証明するその体系での証明図である。

では、NFL$_c$ と NFL$_{cw}$ の場合はどうであろうか。残念ながら、この場合はリダクションに関して閉じていない。

命題 1.3.24 (NFL$_c$ と NFL$_{cw}$ のリダクション) *NFL$_c$* と *NFL$_{cw}$* はリダクションに関して閉じていない。

(反例)

$$\dfrac{\dfrac{\dfrac{\dfrac{(A\supset A\supset B)^1 \quad A^2}{A\supset B}(\supset E) \quad A^2}{B}(\supset E)}{A\supset B}(\supset I)^2 \quad \dfrac{(C\supset A)^3 \quad C^4}{A}(\supset E)}{B}(\supset E)$$

をリダクションすると、

$$\dfrac{\dfrac{(A\supset A\supset B)^1 \quad \dfrac{(C\supset A)^3 \quad C^4}{A}(\supset E)}{A\supset B}(\supset E) \quad \dfrac{(C\supset A)^3 \quad C^4}{A}(\supset E)}{B}(\supset E)$$

$C^4 > (C \supset A)^3$ なので $(\supset E)$ の適用条件を満たさない。

NFL$_c$ や NFL$_{wc}$ をリダクションに関して閉じるように体系を定義できないのであろうか。[2] によれば、FL$_c$ と FL$_{cw}$ ではカットが取れないことが分かっている。カット除去とリダクションは対応していると考えられるので、NFL$_c$ と NFL$_{cw}$ ではリダクションに関して閉じるように定義するのは困難であると思われる。

参考までに、FL_c や FL_{cw} では $A \supset A \supset B, C \supset A, C \vdash B$ は以下のように cut なしで証明できる。

$$\dfrac{C \vdash C \qquad \dfrac{\dfrac{A \vdash A \qquad \dfrac{A \vdash A \quad B \vdash B}{A \supset B, A \vdash B}\,(\supset L)}{A \supset A \supset B, A, A \vdash B}\,(\supset L)}{A \supset A \supset B, A \vdash B}\,(Co)}{A \supset A \supset B, C \supset A, C \vdash B}\,(\supset L)$$

同じ論理式が NFL_c あるいは NFL_{cw} で正規形で証明できるかは分かっていない。

次にこのリダクションをどのような順序で行っても、その手続きが必ず停止すること、すなわち強正規性を持つことを対応するラムダ項を考えて証明する。

1.3.3 直観主義部分構造に対応する型付きラムダ項

証明図のままではリダクションが扱いにくいため、強正規性を証明するために型付きラムダ項を導入する。型付ラムダ項については様々な性質が研究されているので、いままでに証明されている結果等を利用することができる。

まず、変数の上に順序を考える。そして、それを用いて対応する型付ラムダ項を定義していく。

定義 1.3.25 (型付変数) 無限または有限の変数があるとする。
x : 変数, A : 型 $\Rightarrow x^A$: 型付き変数

定義 1.3.26 (型付の変数の半順序) x, y を変数とし、A, B を型とする。変数の上に半順序 $(<, \leq)$ が定義されているとする。このとき、

1. $x^A = y^B \Leftrightarrow x = y$ かつ $A \equiv B$
2. $x^A < y^B \Leftrightarrow x < y$
3. $x^A \leq y^B \Leftrightarrow x < y$ あるいは $x^A = y^B$

これを元に型付の変数の集合上に半順序を導入する。X, Y を変数の集合として、

1. $X < Y \Leftrightarrow \forall x \in X, \forall y \in Y (x < y)$
2. $X \leq Y \Leftrightarrow \forall x \in X, \forall y \in Y (x \leq y)$

体系 NFL に対してカリーハワード同型対応を満たすようにラムダ項を以下のように定義する。

定義 1.3.27 (NFL に対応するラムダ項) $\mathrm{FV}(M)$ で M の中の自由変数の集合を表すとする。

(仮定) x^A : 型付き変数 $\Rightarrow x^A$: *NFL* のラムダ項

$(\supset I)$ x^A : 型付き変数, M^B : *NFL* のラムダ項, *(W 条件)* : $x^A \in \mathrm{FV}(M^B)$, *(E 条件)* : $\mathrm{FV}(M^B) \leq \{x^A\}$
$\Rightarrow (\lambda x^A.M^B)^{A \supset B}$: *NFL* のラムダ項

$(\supset E)$ $M^{A \supset B}, N^A$: *NFL* のラムダ項, *(C 条件)* : $\mathrm{FV}(M^{A \supset B}) \cap \mathrm{FV}(N^A) = \emptyset$, *(E 条件)* : $\mathrm{FV}(M^{A \supset B}) \leq \mathrm{FV}(N^A)$
$\Rightarrow (M^{A \supset B} N^A)^B$: *NFL* のラムダ項

他の体系に対応するラムダ項について: 自然演繹の場合と同様に、NFL に構造規則を加えた他の体系に対応するラムダ項を定義する。ここでは全ては述べないが、例えば NFL_{ec} に対応するラムダ項は、NFL に対応するラムダ項から、E 条件と C 条件を省いたものとして定義される。

定義 1.3.28 (リダクション) 以下のようにラムダ項の左辺を右辺で置き換えることをリダクションと呼ぶ。

1. $((\lambda x.M)^{A\supset B}N^A)^B \rhd [N^A/x^A]M^B$

   $M \rhd N$ のとき、

2. $(\lambda x.M)^{B\supset A} \rhd (\lambda x.N)^{B\supset A}$

3. $(ML)^B \rhd (NL)^B$

補題 1.3.29 (リダクション) NFL, NFL_w, NFL_e,NFL_{ew},NFL_{ec}, NFL_{ecw} の各体系に対応したラムダ項はリダクションに関して閉じている

自然演繹の場合と同様に NFL_c と NFL_{cw} ではリダクションに関して閉じていない。
次に強正規性について考えるが、各 NFL のラムダ項は通常の型付きラムダ項の部分集合であるので、強正規性が成り立つことが直ちに言える。

定理 1.3.30 (強正規性) 各ラムダ項は強正規性を持つ。

(証明) [8], [12] および 1.4.4 節を参照。

また、contraction を持たない体系に対応したラムダ項では、強正規性を簡単に証明することができる。それを見ていくことにする。

定義 1.3.31 (ラムダ項の長さ) 次のようにラムダ項 M の長さ $\mathrm{len}(M)$ を定義する。$\mathrm{len}(M)$ はラムダ項 M に出現する記号 λ の数である。

1. $\mathrm{len}(x) = 0$

2. $\mathrm{len}(\lambda x.M) = \mathrm{len}(M) + 1$

3. $\mathrm{len}(MN) = \mathrm{len}(M) + \mathrm{len}(N)$

補題 1.3.32 (ラムダ項の長さ) NFL,NFL_w,NFL_e,NFL_{ew} に対応したラムダ項ではリダクションを行うと長さが短かくなる。すなわち、

$$M \rhd N \Rightarrow \mathrm{len}(M) > \mathrm{len}(N)$$

(証明) リダクションの構成に関する帰納法による。
この補題から NFL,NFL_w,NFL_e,NFL_{ew} に対応したラムダ項では強正規性を簡単に証明することができることが分かる。

また、チャーチロッサー性についても直ちに言える。

定理 1.3.33 (チャーチロッサー性) 各 NFL の体系に対応したラムダ項はチャーチロッサー性を持つ。すなわち、

$$M \rhd_\beta M_1 \text{かつ } M \rhd_\beta M_2 \Rightarrow \exists N(M_1 \rhd_\beta N \text{ かつ } M_2 \rhd_\beta N)$$

但し $\rhd_\beta$ を $\rhd$ の推移律と反射律の閉包とする。

(証明) [30] 参照。

以上で含意の記号に限定した場合の議論は終了する。次の節では二つの論理積により拡張した場合にどうなるかを議論していく。

1.4　論理積を含んだ NFL

本節では、NFL を二つの論理積により拡張した体系を考える。一つは乗法的論理積 $*$、もう一つは加法的論理積 $\wedge$ である。このうち乗法的論理積 $*$ による拡張はこれまでにも議論されてきた。論理積として乗法的論理積を採用する場合には、次のように考えればよいことが分かっている。

それは、仮定を集合ではなく多重集合と考え、$(\supset I)$ 規則で、消去する仮定にある制限を設けることである。Weakening がない場合には、1 つ以上の仮定を消去する。Contraction がない場合には、一つ以下の仮定を消去すればよい。

ところが、さらに加法的論理積を加えた場合にはこの方法は破綻する。それは次の場合、

$$\dfrac{\dfrac{\begin{array}{cc}\Gamma & \Gamma\\ \vdots & \vdots\\ A & B\end{array}}{A \wedge B}\,(\wedge I)}{C \supset A \wedge B}\,(\supset I)$$

C を Γ から消去する必要があるが、$\Gamma \vdash A$ からいくつかと $\Gamma \vdash B$ からいくつかというようになり、$(\wedge I)$ が適用されると全体でいくつというようには定義することができなるなる。つまり、パスのようなものを考慮しなければならず複雑になりすぎてしまう。

そこで、含意のみの NFL でも述べたように論理式に自然数を付加したものを仮定とし、適用条件を設けることでこの問題点を解決する。

1.4.1　体系 NFL と FL との同値性

これから 8 つの直観主義の体系を定義する。これらは前節の各 NFL を 2 つ論理積の概念を用いて拡張したものである。そして、これらの体系が FL の各体系と同値であることを証明する。

定義 1.4.1 (全順序集合) 仮定の集合 Γ が次の条件を満たすとき、Γ は**全順序集合**であるという。

$$\forall A^n, B^m \in \Gamma(A^n < B^m \text{あるいは} A^n = B^m \text{あるいは} A^n > B^m)$$

定義 1.4.2 (体系 NFL) $A,\ B,\ C$ を論理式のメタ変数とし、$\Gamma,\ \Delta$ を仮定の有限集合のメタ変数とする。このとき、*NFL* の証明図とその証明図が証明するものとを同時に定義する。

(assumption) A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W 条件 : $A^n \in \Gamma$、E 条件 : $\Gamma \leq \{A^n\}$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}\ (\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}\ (\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

($\wedge$ I) $\mathcal{D}_1$ が $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash B$ を証明する証明図であるとする。W 条件 : $\Gamma = \Delta$、E 条件 : $\Gamma \cup \Delta$ は全順序集合を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A \wedge B}\ (\wedge I)$ は $\Gamma \cup \Delta \vdash A \wedge B$ を証明する証明図である。

($\wedge$ E$_l$) $\mathcal{D}_1$ が $\Gamma \vdash A \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A \wedge B$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}\ (\wedge E_l)$ は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

($\wedge$ E$_r$) $\mathcal{D}_1$ が $\Gamma \vdash B \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A \wedge B$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}$ $(\wedge E_r)$は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

($*$ I) $\mathcal{D}_1$ が $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash B$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A * B}$ $(*I)$は $\Gamma \cup \Delta \vdash A * B$ を証明する証明図である。

($*$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A * B$ を証明する証明図であるとする。C 条件 : $\Gamma \cap \Delta = \emptyset$、$E$ 条件 : $\Gamma \leq \Delta$ を満たす。
$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}$ $(*E)$は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

$$A^n\text{:(Axiom)}$$

$$\begin{array}{c} \text{W:}A^n \in \Gamma \\ \text{E:}\Gamma \leq \{A^n\} \\ \Gamma - \{A^n\} \\ \vdots \\ \dfrac{B}{A \supset B}\ (\supset I)^n \end{array} \qquad \begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E:}\Gamma \leq \Delta \\ \Gamma \qquad \Delta \\ \vdots \qquad \vdots \\ \dfrac{A \supset B \quad A}{B}\ (\supset E) \end{array}$$

$$\begin{array}{c} \text{W:}\Gamma = \Delta \\ \text{E:}\Gamma \cup \Delta\text{は全順序集合} \\ \Gamma \quad \Delta \\ \vdots \quad \vdots \\ \dfrac{A \quad B}{A \wedge B}\ (\wedge I) \end{array}$$

$$\begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E:}\Gamma \leq \Delta \\ \Gamma \qquad \Delta \\ \vdots \qquad \vdots \\ \dfrac{A \supset C \quad A \wedge B}{C}\ (\wedge E_l) \end{array} \qquad \begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E:}\Gamma \leq \Delta \\ \Gamma \qquad \Delta \\ \vdots \qquad \vdots \\ \dfrac{B \supset C \quad A \wedge B}{C}\ (\wedge E_r) \end{array}$$

$$\begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E:}\Gamma \leq \Delta \\ \Gamma \quad \Delta \\ \vdots \quad \vdots \\ \dfrac{A \quad B}{A * B}\ (*I) \end{array} \qquad \begin{array}{c} \text{C:}\Gamma \cap \Delta = \emptyset \\ \text{E:}\Gamma \leq \Delta \\ \Gamma \qquad \Delta \\ \vdots \qquad \vdots \\ \dfrac{A \supset B \supset C \quad A * B}{C}\ (*E) \end{array}$$

図 1.4: NFL の推論規則

E 条件,C 条件,W 条件のうちいくつか、あるいは全てを取り除くことにより、含意に制限した場合と同様に NFL を拡張した体系を定義する。ここでは、例として体系 NFL$_{ec}$ を示す。

定義 1.4.3 (体系 NFL$_{ec}$) 体系 NFL_{ec} は、NFL から E 条件と C 条件を取り除いた体系として以下のように定義される。

(assumption) A^n が仮定 $\Rightarrow$ A^n は証明図で $\{A^n\} \vdash A$ を証明する。

($\supset$ I) $\mathcal{D}$ が $\Gamma \vdash B$ を証明する証明図で W条件 : $A^n \in \Gamma$ を満たす。

$\Rightarrow \dfrac{\mathcal{D}}{A \supset B}\ (\supset I)^n$ は $\Gamma - \{A^n\} \vdash A \supset B$ を証明する証明図である。

($\supset$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A$ を証明する証明図であるとする。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}\ (\supset E)$ は $\Gamma \cup \Delta \vdash B$ を証明する証明図である。

($\wedge$ I) $\mathcal{D}_1$ が $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash B$ を証明する証明図であるとする。W条件 : $\Gamma = \Delta$ を満たす。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A \wedge B}\ (\wedge I)$ は $\Gamma \cup \Delta \vdash A \wedge B$ を証明する証明図である。

($\wedge$ E$_l$) $\mathcal{D}_1$ が $\Gamma \vdash A \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A \wedge B$ を証明する証明図であるとする。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}\ (\wedge E_l)$は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

($\wedge$ E$_r$) $\mathcal{D}_1$ が $\Gamma \vdash B \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A \wedge B$ を証明する証明図であるとする。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}\ (\wedge E_r)$は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

($*$ I) $\mathcal{D}_1$ が $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash B$ を証明する証明図であるとする。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A * B}\ (*I)$は $\Gamma \cup \Delta \vdash A * B$ を証明する証明図である。

($*$ E) $\mathcal{D}_1$ が $\Gamma \vdash A \supset B \supset C$ を証明する証明図で $\mathcal{D}_2$ が $\Delta \vdash A * B$ を証明する証明図であるとする。

$\Rightarrow \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}\ (*E)$は $\Gamma \cup \Delta \vdash C$ を証明する証明図である。

例 1.4.4

$\dfrac{\dfrac{A^0 \quad A^0}{A \wedge A}\ (\wedge I)}{A \supset A \wedge A}\ (\supset I)^0$ は FL の全ての体系における $\vdash A \supset A \wedge A$ の証明図である。

$\dfrac{\dfrac{A^1 \quad A^0}{A \wedge A}\ (\wedge I)}{A \supset A \wedge A}\ (\supset I)^0$ は FL_{ew}, FL_{ewc} における $\{A^1\} \vdash A \supset A \wedge A$ を証明する証明図である。

$(\wedge E)$ はなぜこのような形をしているのだろうか。直観主義論理の自然演繹の体系 NJ では $(\wedge E)$ は$\dfrac{A \wedge B}{A}\ (\wedge E)_l$, $\dfrac{A \wedge B}{B}\ (\wedge E)_r$ という形をしている。([8] 参照) $(\wedge E)$ をこのように定義した場合にもこれから示す FL との同値性や強正規性を示すことができる。しかしながら、前述の定義 1.4.2 のように定義しておくと、対応するラムダ項を考えたときに、除去規則が全てある種のアプリケーションの形で統一的に書けるために、ここでは定義 1.4.2 の形を採用した。他方、この形の $(\wedge E)$ を用いると subformula property が失われる。

補題 1.4.5 (仮定の自然数の付け替え) 体系 NFL において、$\mathcal{D}$ が $\Gamma, A_1^{n_1}, \cdots, A_k^{n_k} \vdash B$ を証明する証明図であるとき、$\Gamma, A_1^{n_1+m}, \cdots, A_k^{n_k+m} \vdash B$ $(m \geq 0)$ を証明する $\mathcal{D}$ と同じ高さの証明図 $\mathcal{E}$ が存在する。

(証明) 証明図の高さに関する帰納法。含意に制限した場合 (補題 1.3.12) と同様なので、証明は省略する。

定理 1.4.6 (FL$_{ecw}$ と NFL$_{ecw}$ の同等性) FL_{ecw} と NFL_{ecw} は同値である。すなわち FL で $A_1, ..., A_k \vdash A$ を証明する証明図 $\mathcal{D}$ が存在するとき、及びそのときに限って NFL で $\{A_1^{n_1}, \cdots, A_k^{n_k}\} \vdash A$ を証明する証明図 $\mathcal{E}$ が存在する。但し $A_1^{n_1} < \cdots < A_k^{n_k}$。

(証明) 証明図の高さに関する帰納法。

まず、$\mathrm{FL}_{ecw} \Rightarrow \mathrm{NFL}_{ecw}$ を示す。以下、シーケント計算の仮定 Γ, Δ に対し、対応する自然演繹の仮定を Γ', Δ' と $'$ を付けて表現する。すなわち、$\Gamma \equiv A_1, ..., A_k$ ならば、$\Gamma' = \{A_1^{n_1}, \cdots, A_k^{n_k}\}$ $(A_1^{n_1} < \cdots < A_k^{n_k})$。

1. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma, A \wedge B, \Delta \vdash C}\ (\wedge L)_l$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma, A, \Delta \vdash C$ を証明する証明図である。帰納法の仮定から、$\Gamma', A^n, \Delta' \vdash C$ を証明する証明図 $\mathcal{E}_1$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_1}{\Delta \supset C}\ (\supset I)}{A \supset \Delta \supset C}\ (\supset I)^n \quad (A \wedge B)^n}{\Delta \supset C}\ (\wedge E) \quad \Delta'}{C}\ (\supset E)$$

とおけば、$\mathcal{E}$ は $\Gamma' \cup \{(A \wedge B)^n\} \cup \Delta' \vdash C$ を証明する証明図である。

2. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma, A \wedge B, \Delta \vdash C}\ (\wedge L)_r$ のとき。上と同様。

3. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{\Gamma \vdash A \wedge B}\ (\wedge R)_l$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ は $\Gamma \vdash B$ を証明する証明図である。帰納法の仮定から、$\Gamma' \vdash A$ を証明する証明図 $\mathcal{E}_1$ と $\Gamma'' \vdash B$ を証明する証明図 $\mathcal{E}_2$ が存在する。前補題を繰り返し適用することにより、共通する仮定 Γ''' が存在して、$\Gamma''' \vdash A$ を証明する証明図 $\mathcal{E}_1$ と $\Gamma''' \vdash B$ を証明する証明図 $\mathcal{E}_2$ が存在する。

   このとき、

$$\mathcal{E} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A \wedge B}\ (\wedge I)$$

   とおけば、$\mathcal{E}$ は $\Gamma' \vdash A \wedge B$ を証明する証明図である。

4. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{\Gamma, A * B, \Delta \vdash C}\ (*L)$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma, A, B, \Delta \vdash C$ を証明する証明図である。帰納法の仮定から、$\Gamma', A^n, B^m, \Delta' \vdash C$ を証明する証明図 $\mathcal{E}_1$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{E}_1}{\Delta \supset C}\ (\supset I)}{B \supset \Delta \supset C}\ (\supset I)^m}{A \supset B \supset \Delta \supset C}\ (\supset I)^n \quad (A * B)^n}{\Delta \supset C}\ (*E) \quad \Delta'}{C}\ (\supset E)$$

とおけば、$\mathcal{E}$ は $\Gamma' \cup \{(A * B)^n\} \cup \Delta' \vdash C$ を証明する証明図である。

5. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{\Gamma, \Delta \vdash A * B}\ (\wedge R)_l$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma \vdash A$ を証明する証明図で $\mathcal{D}_2$ は $\Delta \vdash B$ を証明する証明図である。帰納法の仮定から、$\Gamma' \vdash A$ を証明する証明図 $\mathcal{E}_1$ と $\Delta' \vdash B$ を証明する証明図 $\mathcal{E}_2$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A * B}\ (*I)$$

   とおけば、$\mathcal{E}$ は $\Gamma' \cup \Delta' \vdash A * B$ を証明する証明図である。

6. その他の場合。含意のみに制限した場合と同様。

次に逆方向、すなわち $\mathrm{NFL}_{ecw} \Rightarrow \mathrm{FL}_{ecw}$ を示す。すなわち、NFL_{ecw} で $\Gamma' \vdash A$ を証明する証明図 $\mathcal{E}$ が存在するならば、FL_{ecw} で $\Gamma \vdash A$ を証明する証明図 $\mathcal{D}$ が存在する。

1. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A \wedge B}\ (\wedge I)$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma' \cup \Delta' \vdash A$ を証明する証明図で $\mathcal{D}_2$ は $\Delta' \cup \Pi' \vdash B$ を証明する証明図である。帰納法の仮定から、$\Gamma, \Delta \vdash A$ を証明する証明図 $\mathcal{E}_1$ と $\Delta, \Pi \vdash B$ を証明する証明図 $\mathcal{E}_2$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\mathcal{E}_1}{\Gamma, \Delta, \Pi \vdash A}\ (We, Ex) \quad \dfrac{\mathcal{E}_2}{\Gamma, \Delta, \Pi \vdash B}\ (We, Ex)}{\Gamma, \Delta, \Pi \vdash A \wedge B}\ (\wedge R)$$

とおけば、$\mathcal{E}$ は $\Gamma, \Delta, \Pi \vdash A \wedge B$ を証明する証明図である。

2. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A}\ (\wedge E)_l$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma', \Delta' \vdash B \supset A$ を証明する証明図である。$\mathcal{D}_2$ は $\Delta', \Pi' \vdash B \wedge C$ 帰納法の仮定から、$\Gamma, \Delta \vdash B \supset A$ を証明する証明図 $\mathcal{E}_1$ と $\Delta, \Pi \vdash B \wedge C$ を証明する証明図 $\mathcal{E}_2$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\mathcal{E}_2 \quad \dfrac{\mathcal{E}_1 \quad \dfrac{\dfrac{B \vdash B \quad A \vdash A}{B \supset A, B \vdash A}\,(\supset L)}{B \supset A, B \wedge C \vdash A}\,(\wedge L)_l}{\Gamma, \Delta, B \wedge C \vdash A}\,(Cut)}{\Gamma, \Delta, \Delta, \Pi \vdash A}\,(Cut)}{\Gamma, \Delta, \Pi \vdash A}\,(Ex), (Co)$$

とおけば、$\mathcal{E}$ は $\Gamma, \Delta, \Pi \vdash A$ を証明する証明図である。

3. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A}\ (\wedge E)_r$ のとき。上と同様。

4. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A * B}\ (*I)$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma' \cup \Delta' \vdash A$ を証明する証明図で $\mathcal{D}_2$ は $\Delta' \cup \Pi' \vdash B$ を証明する証明図である。帰納法の仮定から、$\Gamma, \Delta \vdash A$ を証明する証明図 $\mathcal{E}_1$ と $\Delta, \Pi \vdash B$ を証明する証明図 $\mathcal{E}_2$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\mathcal{E}_1 \quad \mathcal{E}_2}{\Gamma, \Delta, \Delta, \Pi \vdash A \wedge B}\,(*R)}{\Gamma, \Delta, \Pi \vdash A \wedge B}\,(Ex), (Co)$$

とおけば、$\mathcal{E}$ は $\Gamma, \Delta, \Pi \vdash A * B$ を証明する証明図である。

5. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{A}\ (*E)$ のとき。定義から、$\mathcal{D}_1$ は $\Gamma', \Delta' \vdash B \supset C \supset A$ を証明する証明図であり、$\mathcal{D}_2$ は $\Delta', \Pi' \vdash B * C$ を証明する証明図である。帰納法の仮定から、$\Gamma, \Delta \vdash C \supset B \supset A$ を証明する証明図 $\mathcal{E}_1$ と $\Delta, \Pi \vdash B * C$ を証明する証明図 $\mathcal{E}_2$ が存在する。このとき、

$$\mathcal{E} \equiv \dfrac{\dfrac{\mathcal{E}_2 \quad \dfrac{\mathcal{E}_1 \quad \dfrac{\dfrac{B \vdash B \quad \dfrac{C \vdash C \quad A \vdash A}{C \supset A, C \vdash A}\,(\supset L)}{B \supset C \supset A, B, C \vdash A}\,(\wedge L)_l}{B \supset C \supset A, B * C \vdash A}\,(*L)}{\Gamma, \Delta, B * C \vdash A}\,(Cut)}{\Gamma, \Delta, \Delta, \Pi \vdash A}\,(Cut)}{\Gamma, \Delta, \Pi \vdash A}\,(Ex), (Co)$$

とおけば、$\mathcal{E}$ は $\Gamma, \Delta, \Pi \vdash A$ を証明する証明図である。

そして、含意に制限した場合と同様に次の定理も証明できる。

定理 1.4.7 (FL の拡張体系と対応する NFL の拡張体系の同等性) *FL* あるいは *FL* にいくつかの構造規則を加えた体系と *NFL* から対応する条件を取り除いて得られる体系が同値である。すなわち、*FL* あるいは *FL* にいくつかの構造規則を加えた体系で $A_1, ..., A_k \vdash A$ を証明する証明図 $\mathcal{D}$ が存在するとき、及びそのときに限って *NFL* 対応する条件を取り除いて得られる体系で $\{A_1^{n_1}, \cdots, A_k^{n_k}\} \vdash A$ を証明する証明図 $\mathcal{E}$ が存在する。但し $A_1^{n_1} < \cdots < A_k^{n_k}$。

FL_{cw} は次の興味深い性質を持つ。

命題 1.4.8 FL_{cw} では *exchange* 規則が導出される。

(証明)$\Gamma, A, B, \Delta \vdash C \Rightarrow \Gamma, B, A, \Delta \vdash C$ を示せば十分である。

$$
\dfrac{
\dfrac{B \vdash B \quad A \vdash A}{B, A \vdash B \ast A}\,(\ast L)
\qquad
\dfrac{
\dfrac{
\dfrac{\dfrac{\dfrac{A \vdash A}{B, A \vdash A}\,(We)}{B \ast A \vdash A}\,(\ast L)
\quad
\dfrac{\dfrac{B \vdash B}{B, A \vdash B}\,(We)}{B \ast A \vdash B}\,(\ast L)}
{B \ast A, B \ast A \vdash A \ast B}\,(\ast R)
}{B \ast A \vdash A \ast B}\,(Co)
}{}
}{B, A \vdash A \ast B}\,(Cut)
\qquad
\dfrac{\begin{array}{c}\vdots\\ \Gamma, A, B, \Delta \vdash C\end{array}}{\Gamma, A \ast B \Delta \vdash C}\,(\ast L)
$$
$$
\dfrac{B, A \vdash A \ast B \qquad \Gamma, A \ast B \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}\,(Cut)
$$

または、次のようにしても証明できる。

$$
\dfrac{
\dfrac{\dfrac{A \vdash A}{B, A \vdash A}\,(We) \quad \dfrac{B \vdash B}{B, A \vdash B}\,(We)}{B, A \vdash A \wedge B}\,(\wedge R)
\qquad
\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\\ \Gamma, A, B, \Delta \vdash C\end{array}}{\Gamma, A \wedge B, B, \Delta \vdash C}\,(\wedge L_l)}{\Gamma, A \wedge B, A \wedge B, \Delta \vdash C}\,(\wedge L_r)}{\Gamma, A \wedge B, \Delta \vdash C}\,(Co)
}{\Gamma, B, A, \Delta \vdash C}
\qquad \square
$$

そして、NFL$_{cw}$ でも対応する命題が成り立つ。

命題 1.4.9 *NFL$_{cw}$ では exchange 規則が導出される。*

(証明) $\Gamma \cup \{A^n\} \cup \{B^{n+1}\} \cup \Delta \vdash C \Rightarrow \Gamma \cup \{B^n\} \cup \{A^{n+1}\} \cup \Delta \vdash C$ を証明すれば十分である。

$\quad\;\;\Gamma \cup \{A^n\} \cup \{B^{n+1}\} \cup \Delta \vdash C$ が NFL$_{cw}$ で証明可能

$\Rightarrow \quad \Gamma, A, B, \Delta \vdash C$ が FL$_{cw}$ で証明可能 (定理 1.4.7 より)

$\Rightarrow \quad \Gamma, B, A, \Delta \vdash C$ が FL$_{cw}$ で証明可能 (前命題より)

$\Rightarrow \quad \Gamma \cup \{B^n\} \cup \{A^{n+1}\} \cup \Delta \vdash C$ が NFL$_{cw}$ で証明可能 (定理 1.4.7 より)

直接的な証明は以下のよう。

$$
\dfrac{
\dfrac{
\begin{array}{c}\Gamma, \cancel{A}^n, \cancel{B}^{n+1}, \cancel{\Delta}\\ \vdots\\ C\\ \vdots\;(\supset I)\\ A \supset B \supset (\Delta) \supset C\end{array}
\qquad
\dfrac{
\dfrac{
\dfrac{
\dfrac{\dfrac{\cancel{A}^0}{A \supset A}\,(\supset I)^0}{B \supset A \supset A}\,(\supset I)^0 \quad (B \ast A)^0
}{A}\,(\ast E)
\quad
\dfrac{
\dfrac{\dfrac{\cancel{B}^0}{A \supset B}\,(\supset I)^0}{B \supset A \supset B}\,(\supset I)^0 \quad (B \ast A)^0
}{B}\,(\ast E)
}{A \ast B}\,(\ast I)
}{B \ast A \supset A \ast B}\,(\supset I)^0
\quad
\dfrac{B^n \quad A^{n+1}}{B \ast A}\,(\ast I)
}{A \ast B}\,(\supset E)
}{(\Delta) \supset C}
\qquad
\begin{array}{c}\Delta\\ \vdots\end{array}
}{C}\,(\supset E)
$$

または、次のようにしても証明できる。

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\begin{array}{c}\Gamma, \cancel{A}^n, \cancel{B}^{n+1}, \cancel{\Delta}\\ \vdots\\ C\\ \vdots\;(\supset I)\\ A \supset B \supset (\Delta) \supset C\end{array}
\quad (B \wedge A)^n
}{B \supset (\Delta) \supset C}\,(\wedge E)_r
\quad (B \wedge A)^n
}{(\Delta) \supset C}\,(\wedge E)_l
}{(B \wedge A) \supset (\Delta) \supset C}\,(\supset I)^n
\qquad
\dfrac{B^n \quad A^{n+1}}{B \wedge A}
}{(\Delta) \supset C}\,(\supset E)
\qquad
\begin{array}{c}\Delta\\ \vdots\end{array}
}{C}\,(\supset E)
\qquad \square
$$

命題 1.4.8 及び 1.4.9 の証明を眺めると、exchange を導出するためには、含意と二つの論理積のうちのどちらかがあればよいことが分かる。

1.4.2 レデックスとリダクション

含意だけの場合と同様に、レデックスとリダクションを定義する。

定義 1.4.10 (証明図の代入) $\mathcal{E}$ を $\Gamma \vdash A$ を証明する証明図、A^n を仮定とする。$\mathcal{D}$ に出現する A^n を $\mathcal{E}$ で置き換える代入を $[\mathcal{E}/A^n]\mathcal{D}$ で表し $\mathcal{D}$ の高さに関して帰納的に定義する。

1. $\mathcal{D} \equiv A^n$ のとき。$[\mathcal{E}/A^n]A^n \equiv \mathcal{E}$。

2. $\mathcal{D} \equiv A^m$ $(n \neq m)$ のとき。$[\mathcal{E}/A^n]A^m \equiv A^m$。

3. $\mathcal{D} \equiv B^m$ $(B \not\equiv A)$ のとき。$[\mathcal{E}/A^n]B^m \equiv B^m$。

4. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{A \supset C}\ (\supset I)^n$ のとき。 $[\mathcal{E}/A^n]\mathcal{D} \equiv \mathcal{D}$。

5. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{D \supset C}\ (\supset I)^m$ $(D \not\equiv A)$ で、$D^m \notin \Gamma$ あるいは $A^m \notin \Delta'$ のとき。ただし、Δ' は $\mathcal{D}_1$ の仮定の集合とする。$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]\mathcal{D}_1}{D \supset C}\ (\supset I)^m$。

6. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1}{D \supset C}\ (\supset I)^s$ $(D \not\equiv A)$ で、$A^m \in \Gamma$ でありかつ $A^m \in \Delta'$ のとき。ただし、$\mathcal{D}_1$ は $\Delta' \vdash C$ を証明するの証明図であるとする。$D^k \notin \Delta' \cup \Gamma$ となる D^k を取り、$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]([D^k/D^m]\mathcal{D}_1)}{A \supset B}\ (\supset I)^k$

7. $\mathcal{D} \equiv \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B}\ (Rule)$ のとき。$[\mathcal{E}/A^n]\mathcal{D} \equiv \dfrac{[\mathcal{E}/A^n]\mathcal{D}_1 \quad [\mathcal{E}/A^n]\mathcal{D}_2}{B}\ (Rule)$
(但し $(Rule) = (\supset E), (\wedge I), (\wedge E_l), (\wedge E_r), (*I), (*E)$)

補題 1.4.11 (NFL と NFL$_w$ における代入) 体系 NFL と NFL_w において以下のことが成り立つ。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$ かつ $\Gamma < \{A\}$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。いま、$\Gamma < \Delta$(すなわち、$\Gamma \cap \Delta = \emptyset$ かつ $\Gamma \leq \Delta$) が成り立つとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。

補題 1.4.12 (NFL$_e$ と NFL$_{ew}$ における代入) 体系 NFL_e と NFL_{ew} において以下のことが成り立つ。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。いま、$\Gamma \cap \Delta = \emptyset$ が成り立つとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。

補題 1.4.13 (NFL$_{ec}$ と NFL$_{ecw}$ における代入) 体系 NFL_{ec} と NFL_{ecw} において以下のことが成り立つ。$\mathcal{D}$ を $\Gamma \cup \{A^n\} \vdash B$ 証明する証明図 (但し $A^n \notin \Gamma$) とし、$\mathcal{E}$ を $\Delta \vdash A$ を証明する証明図とする。このとき、$[\mathcal{E}/A^n]\mathcal{D}$ は証明図であり、$\Gamma \cup \Delta \vdash B$ を証明する。

これらの補題は証明図の高さに関する帰納法を用いて証明できる。

定義 1.4.14 (レデックス) $\mathcal{D}_1, \mathcal{D}_2$ を証明図として、証明図に次のような部分証明図が存在するとき、この部分証明図をレデックスと呼ぶ。

$$\dfrac{\dfrac{\mathcal{D}_1}{A \supset B}\ (\supset I)^n \qquad \mathcal{D}_2}{B}\ (\supset E) \quad \dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A \wedge B}\ (\wedge I)}{C}\ (\wedge E_l) \quad \dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A \wedge B}\ (\wedge I)}{C}\ (\wedge E_r) \quad \dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A * B}\ (*I)}{C}\ (*E)$$

定義 1.4.15 (リダクション) 証明図がレデックスを含むとき、その部分証明図を右辺の証明図で置き換えることをリダクションと呼ぶ。

$$\dfrac{\dfrac{\mathcal{D}_1}{A \supset B}\,(\supset I)^n \quad \mathcal{D}_2}{B}\,(\supset E) \ \rhd [\mathcal{D}_2/A^n]\mathcal{D}_1 \qquad \dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A * B}\,(*I)}{C}\,(*E) \ \rhd \ \dfrac{\dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{B \supset C}\,(\supset E) \quad \mathcal{D}_3}{C}\,(\supset E)$$

$$\dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A \wedge B}\,(\wedge I)}{C}\,(\wedge E_l) \ \rhd \ \dfrac{\mathcal{D}_1 \quad \mathcal{D}_2}{C}\,(\supset E) \qquad \dfrac{\mathcal{D}_1 \quad \dfrac{\mathcal{D}_2 \quad \mathcal{D}_3}{A \wedge B}\,(\wedge I)}{C}\,(\wedge E_r) \ \rhd \ \dfrac{\mathcal{D}_1 \quad \mathcal{D}_3}{C}\,(\supset E)$$

$\mathcal{D} \rhd \mathcal{D}'$ のとき次の左辺を右辺で置き換えることもリダクションであるとする。

$$\dfrac{\mathcal{D}}{A}\,(\supset I)^n \quad \rhd \quad \dfrac{\mathcal{D}'}{A}\,(\supset I)^n$$

$$\dfrac{\mathcal{D} \quad \mathcal{E}}{A}\,(Rule) \quad \rhd \quad \dfrac{\mathcal{D}' \quad \mathcal{E}}{A}\,(Rule) \qquad\qquad \dfrac{\mathcal{E} \quad \mathcal{D}}{A}\,(Rule) \quad \rhd \quad \dfrac{\mathcal{E} \quad \mathcal{D}'}{A}\,(Rule)$$

$$(\text{但し } (Rule) = (\supset E), (\wedge I), (\wedge E)_l, (\wedge E)_r, (*I), (*E))$$

NFL_c と NFL_{cw} を除く各体系がリダクションに関して閉じていることを次の補題で示す。

補題 1.4.16 (NFL,NFL$_e$,NFL$_{ec}$ のリダクション) $\mathcal{D}$ を NFL あるいは NFL_e あるいは NFL_{ec} の $\Gamma \vdash A$ を証明する証明図とする。いま $\mathcal{D} \rhd \mathcal{E}$ とのき、$\mathcal{E}$ もまた $\Gamma \vdash A$ を証明するその体系での証明図である。

補題 1.4.17 (NFL$_w$,NFL$_{ew}$,NFL$_{ecw}$ のリダクション) $\mathcal{D}$ を NFL_w あるいは NFL_{ew} あるいは NFL_{ecw} の $\Gamma \vdash A$ を証明する証明図とする。いま $\mathcal{D} \rhd \mathcal{E}$ とのき、ある $\Gamma' \subset \Gamma$ が存在して $\mathcal{E}$ は $\Gamma' \vdash A$ を証明するその体系での証明図である。

代入の場合は、Weakening 規則のあるなしによって 2 通りに補題が別れるのに対して、リダクションの場合は Weakening 規則のあるなしには依らず、Contraction と Exchange のあるなしで 3 通りに別れるところが興味深い。

命題 1.4.18 (NFL$_c$ と NFL$_{cw}$ のリダクション) NFL_c と NFL_{cw} はリダクションに関して閉じていない。

1.4.3 直観主義部分構造論理に対応する型付きラムダ項

含意に制限した場合と同様に、NFL に対応するラムダ項を定義する。

定義 1.4.19 (NFL に対応するラムダ項) 体系 NFL に対してカリーハワード同型対応を満たすラムダ項を以下のように定義する。FV(M) で M の中の自由変数の集合を表す。

(仮定) x^A : 型付き変数 $\Rightarrow x^A$: NFL のラムダ項

$(\supset I)$ x^A : 型付き変数, M^B : NFL のラムダ項, W 条件 : $x^A \in \mathrm{FV}(M^B)$, E 条件 : $\mathrm{FV}(M^B) \leq \{x^A\}$
$\Rightarrow (\lambda x^A.M^B)^{A \supset B}$: NFL のラムダ項

$(\supset E)$ $M^{A \supset B}, N^A$: NFL のラムダ項, $(C$ 条件$)$: $\mathrm{FV}(M^{A \supset B}) \cap \mathrm{FV}(N^A) = \emptyset$, $(E$ 条件$)$: $\mathrm{FV}(M^{A \supset B}) \leq \mathrm{FV}(N^A)$
$\Rightarrow (M^{A \supset B} N^A)^B$: NFL のラムダ項

$(\wedge I)$ M^A, N^B : NFL のラムダ項, $(W$ 条件$)$:$\mathrm{FV}(M^A) = \mathrm{FV}(N^B)$, $(E$ 条件$)$:$\mathrm{FV}(M^A) \cup \mathrm{FV}(N^B)$ は全順序集合
$\Rightarrow \langle M^A, N^B \rangle^{A \wedge B}$: NFL のラムダ項

$(\wedge E_l)$ $M^{A\supset C}, N^{A\wedge B}$: *NFL* のラムダ項, *(C 条件)*:$\mathrm{FV}(M^{A\supset C}) \cap \mathrm{FV}(N^{A\wedge B}) = \emptyset$, *(E 条件)*:$\mathrm{FV}(M^{A\supset C}) \leq \mathrm{FV}(N^{A\wedge B})$
$\Rightarrow (M^{A\supset C} \circ_l N^{A\wedge B})^C$: *NFL* のラムダ項

$(\wedge E_r)$ $M^{B\supset C}, N^{A\wedge B}$: *NFL* のラムダ項, *(C 条件)*:$\mathrm{FV}(M^{B\supset C}), \cap \mathrm{FV}(N^{A\wedge B}) = \emptyset$*(E 条件)*:$\mathrm{FV}(M^{B\supset C}) \leq \mathrm{FV}(N^{A\wedge B})$
$\Rightarrow (M^{B\supset C} \circ_r N^{A\wedge B})^C$: *NFL* のラムダ項

$(*I)$ M^A, N^B : *NFL* のラムダ項, *(C 条件)*:$\mathrm{FV}(M^A) \cap \mathrm{FV}(N^B) = \emptyset$, *(E 条件)*:$\mathrm{FV}(M^A) \leq \mathrm{FV}(N^B)$
$\Rightarrow [M^A, N^B]^{A*B}$: *NFL* のラムダ項

$(*E)$ $M^{A\supset(B\supset C)}, N^{A*B}$: *NFL* のラムダ項, *(C 条件)*:$\mathrm{FV}(M^{A\supset(B\supset C)}) \cap \mathrm{FV}(N^{A*B}) = \emptyset$,
(E 条件):$\mathrm{FV}(M^{A\supset(B\supset C)}) \leq \mathrm{FV}(N^{A*B}) = \emptyset$
$\Rightarrow (M^{A\supset(B\supset C)} \circ N^{A*B})^C$: *NFL* のラムダ項

(注意) $(\wedge I), (*I)$ に対応する構造 $[A, B]\,\langle A, B\rangle$ をペアリングと呼び、$(\supset\ E), (\wedge E_l), (\wedge E_r), (*E)$ に対応する構造 $(AB)\,(A \circ_l B)\,(A \circ_r B)\,(A \circ B)$ をアプリケーションと呼ぶ。

定義 1.4.20 (他の体系に対応するラムダ項) 自然演繹の場合と同様に、*NFL* に構造規則を加えた体系に対応するラムダ項を定義する。ここでは全ては述べないが、例えば NFL_{ec} に対応するラムダ項は、*NFL* に対応するラムダ項から、E–条件と C–条件を省いたものとして定義される。

定義 1.4.21 (リダクション) 以下のようにラムダ項の左辺を右辺で置き換えることをリダクションと呼ぶ。

1. $((\lambda x.M)^{A\supset B} N^A)^B \rhd [N^A/x^A]M^B$
2. $(M^{A\supset C} \circ_l \langle N^A, L^B\rangle^{A\wedge B})^C \rhd (M^{A\supset C} N^A)^C$
3. $(M^{B\supset C} \circ_r \langle N^A, L^B\rangle^{A\wedge B})^C \rhd (M^{B\supset C} L^B)^C$
4. $(M^{A\supset(B\supset C)} \circ [N^A, L^B]^{A*B})^C \rhd ((M^{A\supset(B\supset C)} N^A)^{B\supset C} L^B)^C$
   以下、$M \rhd N$ のとき。
5. $(\lambda x.M)^{B\supset A} \rhd (\lambda x.N)^{B\supset A}$
6. $(M, L)^B \rhd (N, L)^B,\ (L, M)^B \rhd (L, N)^B$ ((,) は任意のペアリングまたはアプリケーションとする)

補題 1.4.22 (リダクション) *NFL*, NFL_w, NFL_e,NFL_{ew},NFL_{ec}, NFL_{ecw} の各体系に対応したラムダ項はリダクションに関して閉じている

また、自然演繹の場合と同様に NFL_c と NFL_{cw} に対応するラムダ項はリダクションに関して閉じていない。
そして強正規性について考えていく。

1.4.4 NFL_{ecw} に対応したラムダ項の強正規性

先に定義したリダクリョンの手続きは停止するだろうか。この性質を強正規性と呼ぶが、含意に制限した場合の手法 (定理 1.3.30 参照) はそのままでは使えない。$\wedge$ と $*$ に関する場合が問題である。
例えば、今 M^{A*B} を考えるとする。強正規性の証明では、$N^{A\supset B\supset A}$ をとり、$(N \circ M)^A$ というラムダ項を考えて、型の構造に関する帰納法を用いようとする。なんらかの尺度で測ったときに、この N の型 $A \supset B \supset A$ が $A * B$ よりも構造が小さくならないと帰納法の仮定を用いることは出来ない。
そこで、型の複雑さを次の CP という関数で捉えて、この複雑さに関する帰納法を用い、参考文献 [8] と同様な考え方により強正規性を示す。

定義 1.4.23 (型の複雑さ (CP)) 型の複雑さ *(CP:型 → 自然数)* を以下のように型の構成に関して帰納的に定義する。

1. $A:$ アトミック $\Rightarrow \mathrm{CP}(A) = 1$
2. $\mathrm{CP}(A \supset B) = \mathrm{CP}(A) + \mathrm{CP}(B)$
3. $\mathrm{CP}(A \wedge B) = (\mathrm{CP}(A) + \mathrm{CP}(B)) \times 2$
4. $\mathrm{CP}(A * B) = (\mathrm{CP}(A) + \mathrm{CP}(B)) \times 2$

定義 1.4.24 (可約集合) ラムダ項の部分集合 RED_A を以下のように型の構成に関して帰納的に定義する。

1. $M^A \in \mathrm{RED}_A$ $(A:$ アトミック$) \Leftrightarrow M$ が強正規性 *(SN)* を持つ
2. $M^{A \supset B} \in \mathrm{RED}_{A \supset B} \Leftrightarrow \forall N^A \in \mathrm{RED}_A ((MN)^B \in \mathrm{RED}_B)$
3. $M^{A \wedge B} \in \mathrm{RED}_{A \wedge B} \Leftrightarrow \forall N^{A \supset A} \in \mathrm{RED}_{A \supset A}, \forall L^{B \supset B} \in \mathrm{RED}_{B \supset B}$
   $((N \circ_l M)^A \in \mathrm{RED}_A$ かつ $(N \circ_r M)^B \in \mathrm{RED}_B)$
4. $M^{A * B} \in \mathrm{RED}_{A * B} \Leftrightarrow \forall N^{A \supset (B \supset A)} \in \mathrm{RED}_{A \supset (B \supset A)}, \forall L^{A \supset (B \supset B)} \in \mathrm{RED}_{A \supset (B \supset B)}$
   $((N \circ M)^A \in \mathrm{RED}_A$ かつ $(L \circ M)^B \in \mathrm{RED}_B)$

定義 1.4.25 (ニュートラル) M^A が型付き変数かあるいはアプリケーションの形をしているとき、ニュートラルと呼ぶ。すなわち、何らかのラムダ項を右にアプリケーションとして付け加えても、それ自身がレデックスにならないようなラムダ項である。

定義 1.4.26 (最長のパスの長さ) M^A が強正規性も持つとき M^A の任意のリダクションは有限の手続きで停止する。それらのリダクションパスのうち最長のものの長さを $\nu(M^A)$ で表わす。

補題 1.4.27 (可約集合) 任意のラムダ項 M^A は *(CR1)* から *(CR4)* までの *CR* 性を全て満たす。

(CR1) $M^A \in \mathrm{RED}_A \Rightarrow M^A : SN$

(CR2) $M^A \in \mathrm{RED}_A, M^A \rhd M'^A \Rightarrow M'^A \in \mathrm{RED}_A$

(CR3) $\forall M:$ ニュートラル, $\forall M'^A (M^A \rhd M'^A \Rightarrow M'^A \in \mathrm{RED}_A) \Rightarrow M^A \in \mathrm{RED}_A$

(CR4) M^A:ニュートラルでかつ正規形 $\Rightarrow M^A \in \mathrm{RED}_A$
((CR4) は (CR3) の特殊形である)

(証明) 型 A の複雑さに関する帰納法。M^A が (CR1) から (CR4) までを満たすとき $M^A \in$ (CR) と書くことにする。

1. $A:$ がアトミックのとき。

   (CR1) RED の定義から明らか。

   (CR2) $M^A \in \mathrm{RED}_A$ をとり、 $M^A \rhd M'^A$ とする。RED の定義から M^A は強正規性を持つ。よって、明らかに $M'^A:$ も強正規性を持つ。故に、$M'^A \in \mathrm{RED}_A$ である。

   (CR3) $M^A \rhd M'^A$ かつ $M'^A \in \mathrm{RED}_A$ なる M'^A をとる。RED の性質より、M'^A は強正規性を持つ。すると、M^A をリダクションした M'^A 全てが強正規性をもつから、M^A も強正規性を持つ。故に、$M^A \in \mathrm{RED}_A$ である。

2. $A \equiv B \supset C$ のとき。

(CR1) $M^{B\supset C} \in \mathrm{RED}_{B\supset C}$ とし変数 x^B をとる。

$\Rightarrow \quad x^B, Mx^C \in (\mathrm{CR}) \quad$ (I.H. より)
$\Rightarrow \quad x^B \in \mathrm{RED}_B \quad$ ((CR4) より)
$\Rightarrow \quad Mx^C \in \mathrm{RED}_C \quad$ (定義より)
$\Rightarrow \quad Mx^C : \mathrm{SN} \quad$ ((CR1) より)

もし $M^{B\supset C}$ から始まる無限のリダクション列があるならば、Mx^B も無限になり矛盾する。故に、$M^{B\supset C}$ は強正規性を持つ。

(CR2) $M^{B\supset C} \in \mathrm{RED}_{B\supset C}$ をとり、$M^{B\supset C} \triangleright M'^{B\supset C}$ なる $M'^{B\supset C}$ と $N^B \in \mathrm{RED}_B$ を取る。

$\Rightarrow \quad N^B, MN^C \in (\mathrm{CR}) \quad$ (I.H. より)
$\Rightarrow \quad MN^C \in \mathrm{RED}_C \quad$ (定義より)
$\Rightarrow \quad MN^C \triangleright M'N^C \quad$ (定義より)
$\Rightarrow \quad M'N^C \in \mathrm{RED}_C \quad$ ((CR2) より)
$\Rightarrow \quad M'^{B\supset C} \in \mathrm{RED}_{B\supset C} \quad$ (N は任意より)

(CR3) $M^{B\supset C}$ をニュートラルとし $M \triangleright M'$ かつ $M'^{B\supset C} \in \mathrm{RED}_{B\supset C}$ とする。今 $N^B \in \mathrm{RED}_B$ を取ると帰納法の仮定より $N, MN \in (\mathrm{CR})$。

MN が正規形ならば (CR4) より $MN \in \mathrm{RED}_C$ である。そうでない場合は、MN のリダクションを考えると、

$$MN \triangleright L \equiv \begin{cases} M'N & \\ MN' & (N \triangleright N') \end{cases}$$

のどちらかが成り立つ。なぜならば、M はニュートラルだからである。そして、$\nu(N)$ に関する帰納法で $L \in \mathrm{RED}_C$ を示す。

(a) $\nu(N) = 0$ のとき。$MN \triangleright M'N$ である。定義より、$M'N \in \mathrm{RED}_C$。

(b) $\nu(N) \neq 0$ のとき。$MN \triangleright M'N$ のときは、定義から $M'N \in \mathrm{RED}_C$。$MN \triangleright MN'$ のときは、(CR2) から $N' \in \mathrm{RED}_B$。I.H. より、$N' \in (\mathrm{CR})$。(CR1) より、N' :SN。$\nu(N) > \nu(N')$ であるから、I.H. より $MN' \in \mathrm{RED}_C$。

故に、$L \in \mathrm{RED}_C$。(CR3) より、$MN \in \mathrm{RED}_C$。

3. $A \equiv B \wedge C$ のとき。

(CR1) $M^{B\supset C} \in \mathrm{RED}_{B\wedge C}$ とし、変数 $x^{B\supset B}$ をとる。

$\Rightarrow \quad x^{B\supset B}, xM^B \in (\mathrm{CR}) \quad$ (I.H. より)
$\Rightarrow \quad x^{B\supset B} \in \mathrm{RED}_{B\supset B} \quad$ ((CR3) より)
$\Rightarrow \quad xM^B \in \mathrm{RED}_B \quad$ (定義より)
$\Rightarrow \quad xM^B : \mathrm{SN} \quad$ ((CR1) より)

もし $M^{B\wedge C}$ から始まる無限のリダクション列があるならば、xM^B も無限となり矛盾する。故に、$M^{B\supset C}$ は強正規性を持つ。

(CR2) $M^{B\wedge C} \in \mathrm{RED}_{B\wedge C}$ とし、$M^{B\wedge C} \triangleright M'^{B\wedge C}$ なる $M'^{B\wedge C}$ と $N^{B\supset B} \in \mathrm{RED}_{B\supset B}$ を取る。

$\Rightarrow \quad N^{B\supset B}, (N \circ_l M)^B \in (\mathrm{CR}) \quad$ (I.H. より)
$\Rightarrow \quad (N \circ_l M)^B \in \mathrm{RED}_B \quad$ (定義より)
$\Rightarrow \quad (N \circ_l M)^B \triangleright (N \circ M')^B \quad$ (定義より)
$\Rightarrow \quad (N \circ_l M')^B \in \mathrm{RED}_B \quad$ ((CR2) より)

同様に、任意の $L^{C\supset C} \in \mathrm{RED}_{C\supset C}$ に対して $(L \circ_r M)^C \in \mathrm{RED}_C$。$N, L$ は任意より、$M'^{B\wedge C} \in \mathrm{RED}_{B\wedge C}$。

(CR3) $M^{B\wedge C}$ をニュートラルとし、任意の $M \rhd M'$ なる M' に対して ${M'}^{B\wedge C} \in \mathrm{RED}_{B\wedge C}$ とする。$N^{B\supset B} \in \mathrm{RED}_{B\supset B}$ を取ると、I.H. より $(N \circ_l M) \in (\mathrm{CR})$。

$(N \circ_l M)$ が正規性ならば (CR4) より、$(N \circ_l M) \in \mathrm{RED}_B$。そうでない場合は $(N \circ_l M)$ から始まるリダクションを考えると、

$$(N \circ_l M) \rhd L \equiv \begin{cases} (N \circ_l M) \\ (N' \circ_l M) \quad (N \rhd N') \end{cases}$$

のいずれかが成り立つ。なぜなら M なニュートラルであるからである。そして、$\nu(N)$ に関する帰納法で $L \in \mathrm{RED}_B$ を示す。

(a) $\nu(N) = 0$ のとき。$(N \circ_l M) \rhd (N \circ_l M')$。定義から $(N \circ_l M) \in \mathrm{RED}_B$。

(b) $\nu(N) \neq 0$ のとき。$(N \circ_l M) \rhd (N \circ_l M')$ ならば、定義より $(N \circ_l M') \in \mathrm{RED}_B$。$(N \circ_l M) \rhd (N' \circ_l M)$ ならば、(CR2) より、$N' \in \mathrm{RED}_B$。I.H. より、$N' \in (\mathrm{CR})$。(CR1) より、$N' : \mathrm{SN}$。よって、$\nu(N) > \nu(N')$ であるから、I.H. より $(N' \circ_l M) \in \mathrm{RED}_B$。

よって $L \in \mathrm{RED}_B$。故に (CR3) より $(N \circ_l M) \in \mathrm{RED}_B$。同様に、任意の $P^{C\supset C} \in \mathrm{RED}_{C\supset C}$ に対して、$(P \circ_r M) \in \mathrm{RED}_C$。

N, P は任意であるから、定義より $M \in \mathrm{RED}_{B\wedge C}$。

4. $A \equiv B * C$ のとき、$A \equiv B \wedge C$ の場合と同様。 □

以上で $M \in RED \Rightarrow M \in SN$ が言えることになる。次に任意のラムダ項 M に対し $M \in RED$ となることを示す。

補題 1.4.28 (仮定) x^A 変数 $\Rightarrow x^A \in \mathrm{RED}_A$.

(証明) (CR3) より明らか。

補題 1.4.29 ($\supset I$) $M \in \mathrm{RED}_B$ とする。$\forall N^A \in \mathrm{RED}_A([N^A/x^A]M^B \in \mathrm{RED}_B) \Rightarrow (\lambda x.M)^{A\supset B} \in \mathrm{RED}_{A\supset B}$.

(証明) $\forall N^A \in \mathrm{RED}_A(\,((\lambda x.M)N)^B \in \mathrm{RED}_B)$ を示す。

$M, N \in \mathrm{RED}$ であるから $M, N : \mathrm{SN}$ である。$\nu(M) + \nu(N)$ に関する帰納法で $\forall L(((\lambda x.M)N \rhd L \Rightarrow L \in \mathrm{RED}_B)$ をまず示す。

1. $\nu(M) + \nu(N) = 0$ のとき。$(\lambda x.M)N \rhd [N/x]M$ であるから、仮定より $[N/x]M \in \mathrm{RED}_B$。

2. $\nu(M) + \nu(N) \neq 0$ のとき。

$$((\lambda x.M)N) \rhd \begin{cases} [N/x]M \\ (\lambda x.M')N \quad (M \rhd M') \\ (\lambda x.M)N' \quad (N \rhd N') \end{cases}$$

(a) $(\lambda x.M)N \rhd [N/x]M$ のとき。仮定より $[N/x]M \in \mathrm{RED}_B$。

(b) $(\lambda x.M)N \rhd (\lambda x.M')N \quad (M \rhd M')$ のとき。(CR2) より $M' \in \mathrm{RED}_B$。M' :SN であり $\nu(M) > \nu(M')$。I.H. より $(\lambda x.M')N \in \mathrm{RED}_B$。

(c) $(\lambda x.M)N \rhd (\lambda x.M)N' \quad (N \rhd N')$ のとき。$(\lambda x.M)N \rhd (\lambda x.M')N$ の場合と同様。

よって $\forall L(((\lambda x.M)N \rhd L \Rightarrow L \in \mathrm{RED}_B)$。$(\lambda x.M)N$ はニュートラルであるから $(\lambda x.M)N \in \mathrm{RED}_B$。$N$ は任意より、定義から $(\lambda x.M) \in \mathrm{RED}_{A\supset B}$。 □

補題 1.4.30 ($\wedge I$) $M^A \in \mathrm{RED}_A$ かつ $N^B \in \mathrm{RED}_B \Rightarrow \langle M, N\rangle^{A\wedge B} \in \mathrm{RED}_{A\wedge B}$

(証明)$L^{A\supset A} \in \mathrm{RED}_{A\supset A}$ をとり、$\nu(L)+\nu(M)+\nu(N)$ に関する帰納法で、$\forall P(L \circ_l \langle M,N\rangle \rhd P \Rightarrow P \in \mathrm{RED}_A)$ をまず示す。

1. $\nu(L)+\nu(M)+\nu(N)=0$ のとき。$L \circ_l \langle M,N\rangle \rhd LM$ であるから、定義より $LM \in \mathrm{RED}_A$。

2. $\nu(L)+\nu(M)+\nu(N)\neq 0$ のとき。

$$L \circ_l \langle M,N\rangle \rhd \begin{cases} (LM) & \\ L' \circ_l \langle M,N\rangle & (L \rhd L') \\ L \circ_l \langle M',N\rangle & (M \rhd M') \\ L \circ_l \langle M,N'\rangle & (N \rhd N') \end{cases}$$

   (a) $L \circ_l \langle M,N\rangle \rhd (LM)$ とのき。定義から $LM \in \mathrm{RED}_A$。

   (b) その他の場合。(CR2) より、$L', M', N' \in \mathrm{RED}$。(CR1) よりそれらはいずれも強正規性を持つ。I.H. より、いずれの場合も $P \in \mathrm{RED}_A$。

よって、$\forall P(L \circ_l \langle M,N\rangle \rhd P \;\Rightarrow\; P \in \mathrm{RED}_A)$ が示された。$L \circ_l \langle M,N\rangle$ はニュートラルであるから、(CR4) より $L \circ_l \langle M,N\rangle \in \mathrm{RED}_A$。

同様に $Q^{B\supset B} \in \mathrm{RED}_B$ をとると、$\forall R(Q \circ_r \langle M,N\rangle \rhd R \Rightarrow R \in \mathrm{RED}_B)$ が言える。

故に、定義より $\langle M,N\rangle \in \mathrm{RED}_{A\wedge B}$ である。 □

補題 1.4.31 **($*I$)** $M^A \in \mathrm{RED}_A$ かつ $N^B \in \mathrm{RED}_B \Rightarrow [M,N]^{A*B} \in \mathrm{RED}_{A*B}$

(証明) 前補題と同様。

補題 1.4.32 **($\wedge E_l$)** $M^{A\supset C} \in \mathrm{RED}_{A\supset C}$かつ $N^{A\wedge B} \in \mathrm{RED}_{A\wedge B} \Rightarrow (M \circ_l N)^C \in \mathrm{RED}_C$

(証明)$(M \circ_l N)$ が正規形ならば、(CR4) より、$(M \circ_l N)^C \in \mathrm{RED}_C$ である。そうでない場合には、$\nu(M)+\nu(N)$ に関する帰納法で $\forall L((M \circ_l N) \rhd L \Rightarrow L \in \mathrm{RED}_C)$ を示す。

1. $\nu(M)+\nu(N)=0$ のとき。$N \equiv \langle P,Q\rangle$ であり $(M \circ_l N) \rhd (MP)$。$(\lambda x.x)^{A\supset A}$ をとると、$(\lambda x.x) \in \mathrm{RED}_{A\supset A}$ であり、定義より $(\lambda x.x) \circ_l \langle P,Q\rangle \in \mathrm{RED}_A$。(CR2) より、$(\lambda x.x) \circ_l \langle P,Q\rangle \rhd (\lambda x.x)P \rhd P \in \mathrm{RED}_A$。定義より $(MP) \in \mathrm{RED}_C$。

2. $\nu(M)+\nu(N)\neq 0$ のとき。

$$(M \circ_l N) \rhd \begin{cases} (MP) & (N \equiv \langle P,Q\rangle) \\ (M' \circ_l N) & (M \rhd M') \\ (M \circ_l N') & (N \rhd N') \end{cases}$$

   (a) $M \circ_l N \rhd (MP)$ のとき。$\nu(M)+\nu(N)=0$ の場合と同様。

   (b) $(M \circ_l N) \rhd (M' \circ_l N)$ $(M \rhd M')$ あるいは $(M \circ_l N) \rhd (M \circ_l N')$ $(N \rhd N')$ のとき。I.H. より、いずれの場合も $L \in \mathrm{RED}_C$。

   よって $\forall L(M \circ_l N \rhd L \Rightarrow L \in \mathrm{RED}_C)$。$(M \circ_l N)$ はニュートラルであるから、(CR3) より $(M \circ_l N) \in RED_C$。
   □

補題 1.4.33 **($\wedge E_r$)** $M^{B\supset C} \in \mathrm{RED}_{B\supset C}$かつ $N^{A\wedge B} \in \mathrm{RED}_{A\wedge B} \Rightarrow (M \circ_r N)^C \in \mathrm{RED}_C$

(証明) 前補題と同様。

補題 1.4.34 $(*E)$ $M^{A\supset(B\supset C)} \in \mathrm{RED}_{A\supset(B\supset C)}$かつ $N^{A*B} \in \mathrm{RED}_{A*B} \Rightarrow (M \circ N)^C \in \mathrm{RED}_C$

(証明) $N \equiv [P, Q]$ のときは、$(\lambda xy.x)^{A\supset(B\supset A)}$ と $(\lambda xy.y)^{A\supset(B\supset B)}$ をとることで $P \in \mathrm{RED}_A,\ Q \in \mathrm{RED}_B$ を示すことができる。

そうでないときには、前補題と同様である。

補題 1.4.35 (主補題) M^A : ラムダ項、$N_1^{B_1}, ..., N_n^{B_n} \in \mathrm{RED} \Rightarrow [N_1^{B_1}/x_1^{B_1}, ..., N_n^{B_n}/x_n^{B_n}]M^A \in \mathrm{RED}_A$

(証明) M^A の構成に関する帰納法。$\sigma = [N_1^{B_1}/x_1^{B_1}, ..., N_n^{B_n}/x_n^{B_n}]$ とする。

1. $M^A \equiv x_i^{B_i}$ $(1 \le i \le n)$ のとき。$\sigma x_i^{B_i} \equiv N_i^{B_i} \in \mathrm{RED}_{B_i}$。
2. $M^A \equiv x^A$ and $x_A \ne x_1^{B_1}, ..., x_n^{B_n}$ のとき。$\sigma x^A \equiv x^A \in \mathrm{RED}_A$. (補題より)
3. $M^A \equiv (N^B, L^C)^A$ ($(,)$ はペアリングまはたアプリケーション) のとき。I.H. より、$\sigma N, \sigma L \in \mathrm{RED}$。故に補題あるいは定義から $\sigma M \equiv (\sigma N, \sigma L) \in \mathrm{RED}_A$。
4. $M^A \equiv (\lambda x.N)^{B\supset C}$ のとき。$L^B \in \mathrm{RED}_B$ をとり、$\sigma' = (\sigma - [N_i^B/x^B]) \cup [L^B/x^B]$ とおく。I.H. より、$\sigma' M \in \mathrm{RED}_A$。$y^A \notin FV(M) \cup FV(N_1^{B_1}) \cup \cdots \cup FV(N_n^{B_n})$ をとると、$\sigma' M \equiv [L^A/y^A](\sigma - [N_i^A/x^A])[y^A/x^A]M$ である。L^A は任意より、補題から $(\lambda y.(\sigma - [N_i^A/x^A])[y^A/x^A]M) \equiv \sigma(\lambda x.M) \in \mathrm{RED}_{B\supset C}$。 □

定理 1.4.36 (強正規性) NFL_{ecw} に対応するラムダ項は強正規性を持つ。

(証明) 前補題で $N_1 \equiv x_1, ..., N_n \equiv x_n$ とおくと任意のラムダ項 M^A に対して $M^A \in \mathrm{RED}_A$。よって (CR1) より M^A は強正規性を持つ。 □

系 1.4.37 (強正規性) NFL_{ecw} よりも弱い体系に対応するラムダ項も強正規性を持つ。

1.4.5 Contraction の無いラムダ項の強正規性

1.3.3 節で示したように、含意に制限した場合の Contraction の無いラムダ項では、強正規性の証明は容易であった。なぜなら、補題 1.3.32 で示したように、リダクションを行うとその長さが短くなるからである。同じように、論理積によって拡張した場合も証明できるであろうか。残念ながらそのままのアイデアを用いることはできない。それは、例えば次の場合である。

$$(\lambda x.\langle x, x\rangle)M \rhd \langle M, M\rangle$$

この場合には長さが長くなってしまう。しかしながら、それでもなお注意深く「複雑さ」を示す関数を定義することにより、リダクションによりその値が小さくすることができる。それが、次に定義する MP である。この MP を用いることによりラムダ項のリダクションの上限が与えられる。

それに対し、一般のラムダ項ではこのような上限を与えることができない。MP の定義では型に関する情報を考慮していないため、MP のようなリダクションの上限を与える関数が定義されるとなると、一般の型なしラムダ項までもが強正規性を持つことになってしまう。ところが、一般の型なしラムダ項は強正規性を持たないため、これは矛盾となる。

定義 1.4.38 (複雑さ) ラムダ項の複雑さ *(MP:ラムダ項 → 自然数)* を構成に関して帰納的に定義する。

1. $\mathrm{MP}(x) = 2$
2. $\mathrm{MP}([M, N]) = \mathrm{MP}(M) \times \mathrm{MP}(N)$

3. $\mathrm{MP}(\langle M, N\rangle) = \mathrm{MP}(M) + \mathrm{MP}(N)$

4. $\mathrm{MP}((\lambda x.M)) = \mathrm{MP}(M)$

5. $\mathrm{MP}((M \circ N)) = (\mathrm{MP}(M) + 1) \times (\mathrm{MP}(N) + 1)$

6. $\mathrm{MP}((M, N)) = \mathrm{MP}(M) \times \mathrm{MP}(N)$ （それ以外のとき）

ここで興味深いことがある。それは MP は、加法的論理積は加法で、乗法的論理積は乗法で定義されているという点である。

強正規性の証明には、$M \rhd N \Rightarrow \mathrm{MP}(M) > \mathrm{MP}(N)$ を示せば十分であり、このためには $\mathrm{MP}((\lambda x.M)N) > \mathrm{MP}([N/x]M)$ を証明することが必要である。そのためにまず次の補題を示す。

補題 1.4.39 $(\lambda x.M)N$ が *Contraction* の無いラムダ項ならば、

$$\mathrm{MP}((\lambda x.M)N) - (\mathrm{MP}(N) - 1) > \mathrm{MP}([N/x]M)$$

(証明) M の構成に関する帰納法。

$$L(eft) = \mathrm{MP}((\lambda x.M)N) - (\mathrm{MP}(N) - 1)$$
$$R(ight) = \mathrm{MP}([N/x]M)$$
$$n = \mathrm{MP}(N)$$
$$p = \mathrm{MP}(P)$$
$$q = \mathrm{MP}(Q)$$

とする。

1. $M \equiv x$ のとき。

$$\begin{aligned} L &= \mathrm{MP}((\lambda x, x)N) - (\mathrm{MP}(N) - 1) \\ &= \mathrm{MP}(\lambda x, x) \times \mathrm{MP}(N) - (\mathrm{MP}(N) - 1) \\ &= 2 \times n - (n - 1) \\ &= n + 1 \\ R &= \mathrm{MP}([N/x]x) = \mathrm{MP}(N) = n \\ L - R &= 1 > 0 \end{aligned}$$

故に $L > R$。

2. $M \equiv [P, Q]$ のとき。

$$\begin{aligned} L &= \mathrm{MP}((\lambda x, [P, Q])N) - (\mathrm{MP}(N) - 1) \\ &= \mathrm{MP}(\lambda x, [P, Q])) \times (\mathrm{MP}(N) - (\mathrm{MP}(N) - 1) \\ &= (p \times q) \times n - (n - 1) \\ &= (pq - 1)n + 1 \\ R &= \mathrm{MP}([N/x][P, Q]) \\ &= \begin{cases} \mathrm{MP}([[N/x]P, Q]) \\ \mathrm{MP}([P, [N/x]Q]) \end{cases} \end{aligned}$$

$R = \mathrm{MP}([[N/x]P, Q])$ のとき。

$$\begin{aligned}
R &= \underline{\mathrm{MP}([N/x]P)} \times \mathrm{MP}(Q) \\
&< (\{\mathrm{MP}((\lambda x.P)N) - (\mathrm{MP}(N) - 1)\}) \times q \quad (\text{I.H. より}) \\
&= (pn - (n-1))q \\
&= (pq - q)n + q \\
&= R' \\
L - R' &= qn - q - n + 1 \\
&= (q-1)(n-1) > 0
\end{aligned}$$

故に $L > R$。$R = \mathrm{MP}([P, [N/x]Q])$ のときも同様。

3. $M \equiv \langle P, Q \rangle$ のとき。

$$\begin{aligned}
L &= \mathrm{MP}((\lambda x.\langle P, Q \rangle)N) - (\mathrm{MP}(N) - 1) \\
&= (p+q) \times n - (n-1) \\
&= (p+q)n - (n-1) \\
R &= \mathrm{MP}([N/x]\langle P, Q \rangle) \\
&= \mathrm{MP}(\langle [N/x]P, [N/x]Q \rangle) \\
&= \underline{\mathrm{MP}([N/x]P)} + \underline{\mathrm{MP}([N/x]Q)} \\
&< \{\mathrm{MP}((\lambda x.P)N) - (\mathrm{MP}(N) - 1)\} + \{\mathrm{MP}((\lambda x.Q)N) - (\mathrm{MP}(N) - 1)\} \\
&\quad (\text{I.H. より}) \\
&= (pn - (n-1)) + (qn - (n-1)) \\
&= (p+q)n - 2(n-1) \\
&= R' \\
L - R' &= n - 1 > 0
\end{aligned}$$

故に $L > R$.

4. それ以外のとき。同様。 □

この補題の '$-(\mathrm{MP}(N) - 1)$' は何を示しているのであろうか。それは、帰納法によって証明するために必要な、命題を十分に強くする値であろう。この値がどこから出てきたかといえば、試行錯誤としかいいようがない。

補題 1.4.40 M を *Contraction* の無いラムダ項とする。

$$M \rhd N \Longrightarrow \mathrm{MP}(M) > \mathrm{MP}(N)$$

(証明) M の構成に関する帰納法。

1. $M \equiv (\lambda x.P)Q$ で $N \equiv [Q/x]P$ のとき。前補題より、$\mathrm{MP}(M) - (\mathrm{MP}(Q) - 1) > \mathrm{MP}(N)$。明らかに $\mathrm{MP}(Q) - 1 > 0$。故に $\mathrm{MP}(M) > \mathrm{MP}(N)$。

2. その他の場合。帰納法により容易に証明できる。 □

定理 1.4.41 (強正規性) *Contraction* の無いラムダ項のリダクションは有限であり、*MP* はその上限を与える。

(証明) M から始まる無限列 $M \equiv M_0 \rhd M_1 \rhd \cdots$ があったとすると、前補題より $MP(M_0) > MP(M_1) > \cdots$ が無限となるが、これは矛盾である。故に Contraction の無いラムダ項は強正規性を持ち、MP はそのリダクション回数の上限を与える。 □

ラムダ項の強正規性を用いると、NFL の各拡張体系の証明図のリダクションの強正規性を示すことができる。ラムダ項の変数は高々加算無限なので、自然数から変数への一対一対応の関数が存在する。これを用いて、ラムダ項から証明図へのリダクションに関する同型写像を定義できる。いまこの同型写像を DT とすると、次の定理が成り立つ。

定理 1.4.42 (NFL の各拡張体系の証明図の強正規性) *NFL* 及びその拡張体系の任意の証明図は強正規性を持つ。

(証明) 今、証明図 $\mathcal{D}_1$ から始まる無限列

$$\mathcal{D}_1 \rhd \mathcal{D}_2 \rhd \cdots$$

が存在したとする。このとき、同型写像 DT により、対応するラムダ項

$$\mathrm{DT}(\mathcal{D}_1) \rhd \mathrm{DT}(\mathcal{D}_2) \rhd \cdots$$

が無限となる。任意のラムダ項は強正規性を持つことからこれは矛盾である。よって、任意の証明図は強正規性を持つ。

1.4.6 ラムダ項のチャーチロッサー性

今、強正規性を示すことができたので、チャーチロッサー性は弱チャーチロッサー性さえ示せば得ることができる。([31] 参照)

しかしながら、高橋の並行リダクション [30] を用いるとチャーチロッサー性を直接証明することはそれほど難しくはないので、ここではチャーチロッサー性を直接証明する。

定義 1.4.43 (β リダクション) $\rhd_\beta$ を $\rhd$ の反射律と推移律の閉包とする。すなわち、

$$\begin{array}{cl} & M \rhd_\beta N \\ \Leftrightarrow & \text{ある自然数 } n \geq 0 \text{ およびあるラムダ項 } M_0, \cdots, M_n \text{ に対して、} M \equiv M_0 \rhd \cdots \rhd M_n \equiv N \end{array}$$

補題 1.4.44

$$M_1 \rhd_\beta N_1 \text{かつ } M_2 \rhd_\beta N_2 \Rightarrow (\lambda x.M_1) \rhd_\beta (\lambda x.N_1) \text{ および } (M_1, M_2) \rhd_\beta (N_1, N_2)$$

((,) ペアリングまたはアプリケーション)

(証明) $M_1 \equiv P_0 \rhd \cdots \rhd P_i \equiv N_1, M_2 \equiv Q_0 \rhd \cdots \rhd Q_j \equiv N_2 \ (i, j \geq 0)$
$(\lambda x.M_1) \equiv (\lambda x.P_0) \rhd \cdots \rhd (\lambda x.P_i) \equiv (\lambda x.N_1)$
$(M_1, M_2) \equiv (P_0, Q_0) \rhd (P_1, Q_0) \rhd \cdots \rhd (P_i, Q_0) \rhd (P_i, Q_1) \rhd \cdots (P_i, Q_j) \equiv (N_1, N_2)$ □

定義 1.4.45 (並行リダクション) 次の $\rhd_p$ の左辺を右辺に置き換えることを**並行リダクション**と呼ぶ。

1. $x \rhd_p x$

$M_1 \rhd_p N_1, M_2 \rhd_p N_2, N_3 \rhd_p M_3$ のとき。

2. $((\lambda x.M_1)M_2) \rhd_p [N_1/x]N_2$

3. $(M_1 \circ_l \langle M_2, M_3\rangle) \rhd_p (N_1 N_2)$

4. $(M_1 \circ_r \langle M_2, M_3\rangle) \rhd_p (N_1 N_3)$

5. $(M_1 \circ [M_2, M_3]) \rhd_p ((N_1 N_2) N_3)$

6. $(\lambda x.M_1) \rhd_p (\lambda x.N_1)$

7. $(M_1, M_2) \rhd_p (N_1, N_2)$ ((,) はペアリングまたはアプリケーション)

例 1.4.46
$(\lambda x.x)y \rhd_p (\lambda x.x)y$
$(\lambda x.x)y \rhd_p [y/x]x \equiv y$

補題 1.4.47

$$M \rhd_p M$$

補題 1.4.48

$$M \rhd N \Rightarrow M \rhd_p N$$

補題 1.4.49

$$M \rhd_p N \Rightarrow M \rhd_\beta N$$

補題 1.4.50

$$M_1 \rhd_p N_1, M_2 \rhd_p N_2 \Rightarrow [M_2/x]M_1 \rhd_p [N_2/x]N_2$$

(証明) M あるいは M_1 の構成に関する帰納法。

定義 1.4.51 M に対して M^* を以下のように定義する。

1. $M \equiv x$ のとき。$M^* \equiv x$

2. $M \equiv ((\lambda x.M_1)M_2)$ のとき。$M^* \equiv [M_1^*/x]M_2^*$

3. $M \equiv (M_1 \circ_l \langle M_2, M_3\rangle)$ のとき。$M^* \equiv (M_1^* M_2^*)$

4. $M \equiv (M_1 \circ_r \langle M_2, M_3\rangle)$ のとき。$M^* \equiv (M_1^* M_3^*)$

5. $M \equiv (M_1 \circ [M_2, M_3])$ のとき。$M^* \equiv ((M_1^* M_2^*) M_3^*)$

6. $M \equiv (\lambda x.M_1)$ のとき。$M^* \equiv (\lambda x.M_1^*)$

7. $M \equiv (M_1, M_2), (M_1, M_1)$ がレデックスでないとき。$M^* \equiv (M_1^*, M_2^*)$ ((,) はペアリングまたはアプリケーション)

すなわち、M^* とは M の中の全てのレデックスを並行にリダションした結果である。

補題 1.4.52

$$\forall N(M \rhd_p N \Rightarrow N \rhd_p M^*)$$

(証明) M の構成に関する帰納法。

1. $M \equiv (\lambda x.M_1)M_1$ のとき。$M_1 \rhd_p N_1, M_2 \rhd_p N_2$ であるような N_1, N_2 をとる。$N \equiv (\lambda x.N_1)N_2$ あるいは $N \equiv [N_2/x]N_1$。I.H. より、$N_1 \rhd_p M_1^*, N_2 \rhd_p M_2^*$。

   $N \equiv (\lambda x.N_1)N_2 \Rightarrow N \rhd_p [M_2^*/x]M_1^* \equiv M^*$

   $N \equiv [N_2/x]N_1 \Rightarrow N \rhd_p [M_2^*/x]M_1^*$ (補題 1.4.50 より) $\equiv M^*$

2. $M \equiv (M_1 \circ_l \langle M_2, M_3\rangle)$ のとき。$M_1 \rhd_p N_1, M_2 \rhd_p N_2, M_3 \rhd_p N_3$ であるような N_1, N_2, N_3 をとる。$N \equiv (N_1 \circ_l \langle N_2, N_3\rangle)$ あるいは $N \equiv (N_1 N_2)$。I.H. より $N_1 \rhd_p M_1^*, N_2 \rhd_p M_2^*, N_3 \rhd_p M_3^*$。

   $N \equiv (N_1 \circ_l \langle N_2, N_3\rangle) \Rightarrow N \rhd_p (M_1^* M_2^*) \equiv M^*$

   $N \equiv (N_1 N_2) \Rightarrow N \rhd_p (M_1^* M_2^*) \equiv M^*$

3. その他の場合は同様である。 □

定理 1.4.53 (Church–Rosser)

$$M \rhd_\beta M_1, M \rhd_\beta M_2 \Rightarrow \exists N(M_1 \rhd_\beta N, M_2 \rhd_\beta N)$$

(証明) $M \equiv M_{0,0} \rhd M_{1,0} \rhd \cdots \rhd M_{i,0} \equiv M_1$、$M \equiv M_{0,0} \rhd M_{0,1} \rhd \cdots \rhd M_{0,j} \equiv M_2$ $(i, j \geq 0)$ とおく。補題 1.4.48 より $M_{0,0} \rhd_p \cdots \rhd_p M_{i,0}$、$M_{0,0} \rhd_p \cdots \rhd_p M_{0,j}$ がいえる。$M_{k,l} \equiv M^*_{k-1,l-1}(k, l \geq 1)$ とおくと、補題 1.4.52 より $M_{k,0} \rhd_p M_{k,1}$。また $M_{k,n} \rhd_p M_{k,n+1}$。 すると $M_1 \equiv M_{i,0} \rhd_p M_{i,1} \rhd_p \cdots \rhd_p M_{i,j}$。 補題 1.4.49 より $M_1 \rhd_\beta M_{i,j}$。同様に $M_2 \rhd_\beta M_{i,j}$。 □

$$\begin{array}{ccccccc}
M_{0,0} & \rhd_p & M_{1,0} & \rhd_p & \cdots & \rhd_p & M_{i,0} \\
\nabla_p & & \nabla_p & & & & \nabla_p \\
M_{0,1} & \rhd_p & M_{1,1} & \rhd_p & \cdots & \rhd_p & M_{i,1} \\
\nabla_p & & \nabla_p & & & & \nabla_p \\
\vdots & & \vdots & & & & \vdots \\
M_{0,j} & \rhd_p & M_{1,j} & \rhd_p & \cdots & \rhd_p & M_{i,j}
\end{array}$$

図 1.5: チャーチロッサー性の証明

1.5 結論

Gentzen は LJ と NJ 論理的に同値であることを示したが、ではシーケント計算と自然演繹の本質的な違いはな何であろうか。その一つの答えが部分構造論理であろう。部分構造論理の研究が答えを指し示してくれると考えられる。

部分構造論理の自然演繹体系としては、[3] や [16] が知られている。しかしながら、いずれも線型論理のみの形式化であり、部分構造論理全体を扱うものではない。本論文により、推論規則への適用条件によって部分構造論理全体を統一的に扱う手法が示された。

また、[3] はシーケント計算風の形式化であり、明示的に weakening 規則や contraction 規則が表われてしまっている。それに対し本論文では、そうした構造規則が出現しない本当の意味での自然演繹の体系になっている。

[16] では対応するラムダ項が導入されているが、それは対応する証明図が一意に定まらないとうカリーハワード同型対応を見たさないものである。本論文では、カリーハワード同型対応を見たすラムダ項を導入することにより、ラムダ項の強正規性から直ちに証明図の強正規性が導かれる。

シーケント計算において、仮定は列として扱われるのに対し、自然演繹では仮定は集合として扱われる。これは一つの本質的な違いである。この性質により、自然演繹では構造規則は隠れてしまう。本論文では、論理式になんらかのインデックスを付加したものを仮定とし、推論規則の適用に条件を付加することにより、隠れていた構造規則が明かになる。

それとは逆に、全ての構造規則をもったシーケント計算では、論理式は集合として扱うことができる。すなわち、$(\supset I)$ 規則では exchange 規則と weakening 規則が、$(\supset E)$ 規則では exchange 規則と contraction 規則が暗に含まれているのである。

そして、本論文で扱った全ての自然演繹の体系は、リダクションが考慮できる限りは強正規性を持つことが示された。特に Contraction の無い体系では、簡単な証明が存在し、その証明に用いた関数がリダクション回数の上限を与えることが分かった。

そして、自然演繹の体系では体系が弱くなればなるほど、適用条件が強くなっていくことが分かった。

この結果の主な内容は、1994 年の応用数学合同研究集会 [17]、1995 年の NLS'95[18] 及び同 1995 年の書き換えシステムの理論とその応用 [19] で発表されている。

部分構造論理については、小野寛晰 [25] を参照のこと。自然演繹体系については、Prawitz[28], van Dalen[4] を参照のこと。

第2章　ラムダ項の展開とBCK–ラムダ項における最短リダクション戦略

本章では、ラムダ項のラムダ項の展開と BCK–ラムダ項における最短リダクション戦略を示す。先の章で部分構造論理に対応したラムダ項を導入したが、その性質をリダクション戦略の観点から眺めるものである。

2.1　はじめに

2.1.1　最短リダクション戦略とは

どんなリダクション系においても、次の二つの根源的問いがある。

- どのようにリダクションを行なうと、最短ステップで手続きが完了するか。
- どのようにリダクションを行なうと、最長ステップで手続きが完了するか。

前者の問いに対する答えを最短リダクション戦略と呼び、後者の問いに対する答えを最長リダクション戦略と呼ぶ。

リダクションのステップ数は計算時間に直結するので、この最短リダクション戦略を求めることの意義は大きい。リダクション系は計算機の関数型言語の計算モデルであるので、この問題を解くことは実際の関数型言語の設計への寄与が大きいと考えられる。

しかしながら、一般のラムダ項では最短リダクション戦略を求める問題は否定的に解かれている。実際、一般のラムダ項では有効な再帰探索法が存在しない ([1] 参照) ことが知られている。

では、制限を加えたラムダ項ではどうであろうか。それが本章における主題である。ラムダ項のラムダ項の展開と BCK–ラムダ項に付いて最短リダクション戦略を見いだす。

2.1.2　背景

ラムダ項の最長リダクション戦略については、図 2.1 に示すようにかなりのことが分かっている。図において斜線部分は最長リダクション戦略が知られていることを表している。弱正規性 (WN) を持たないラムダ項では、どのようにリダクションを行っても正規形とはならないことから、全てのリダクション戦略が最長となる。弱正規性は持つが強正規性を持たないラムダ項では、perpetual なリダクション戦略が無限のリダクションとなる、すなわち最長となることが知られている ([1] 参照)。perpetual なリダクション戦略としては、F_∞ や call–by–value 等がある。そして型付ラムダ項やラムダ項の展開でも、F_∞ リダクション戦略が最長である ([35],[36] 参照)。つまり、今まで分かっているところでは、F_∞ リダクション戦略が全ての場合において最長である。そのため、図 2.1 の空白部分でも、F_∞ リダクション戦略が最長となるのではないかと予想される。

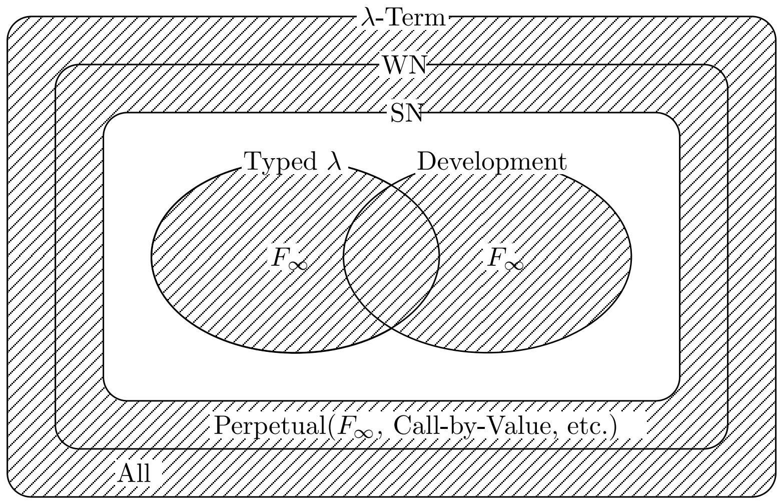


図 2.1: 最長リダクション戦略

ここで、F_∞ リダクション戦略を示しておこう。それは次のようなものである。

定義 2.1.1 (F_∞ リダクション戦略) *M を正規形ではないラムダ項とする。1 ステップの F_∞ リダクション戦略の結果を $F_\infty(M)$ で表すとする。*

1. $M \equiv (\lambda x.M_1)M_2$ のとき。
   (a) $x \in \mathrm{FV}(M_1)$ のとき。
   $F_\infty(M) \equiv [M_2/x]M_1$
   (b) $x \notin \mathrm{FV}(M_1)$ のとき。
   i. M_2 が正規形でないとき。
   $F_\infty(M) \equiv (\lambda x.M_1)F_\infty(M_2)$
   ii. M_2 が正規形のき。
   $F_\infty(M) \equiv [M_2/x]M_1$
2. $M \equiv \lambda x.M_1$ のとき。
   $F_\infty(M) \equiv \lambda x.F_\infty(M_1)$
3. $M \equiv M_1M_2$ で、M_1M_2 自身はレデックスでないとき。
   (a) M_1 が正規形でないとき。
   $F_\infty(M) \equiv F_\infty(M_1)(M_2)$
   (b) M_1 が正規形のき。
   $F_\infty(M) \equiv (M_1)F_\infty(M_2)$

つまり、リダクションをしてラムダ項が消える場合には、代入項が正規形になってからそのレデックスをリダクションするが、消えない場合には、そこからリダクションするという戦略である。

このとき最長リダクションに関する次の 3 つの定理が成り立つ。証明については、それぞれ [1],[35],[36] を参照のこと。

定理 2.1.2 (perpetual な戦略) *M を強正規性を持たないラムダ項とする。このとき、F_∞ リダクション戦略は無限列となる。*

定理 2.1.3 (ラムダ項の展開における最長リダクション戦略) M を λ' 項とする。このとき、β' リダクションにおいて F_∞ リダクション戦略は最長となる。

定理 2.1.4 (型付きラムダ項における最長リダクション戦略) M を型付きラムダ項とする。このとき、F_∞ リダクション戦略は最長となる。

では、最短リダクション戦略はどうであろうか。[15] によれば次の定理がなりたつ。

定理 2.1.5 最左完全並行リダクションは最短リダクションである。

最左完全並行リダクションは並行リダクションの一つであり、最左レデックスをリダクションするが、そのときにコピーされたレデックスも同時にリダクションするものである。この並行リダクションはリダクションを越えたものであるが、最左完全並行リダクションでコピーが生じない場合はリダクションに一致する。すなわち次の系が直ちに成り立つ。

系 2.1.6 *BCK'–ラムダ項では最左リダクションが最短リダクションである。*

ここで BCK'–ラムダ項は次のようなラムダ項である。

定義 2.1.7 (BCK'–ラムダ項)

1. x が変数 $\Rightarrow$ x は *BCK*–ラムダ項
2. M_1, M_2 が *BCK*–ラムダ $\Rightarrow$ (M_1M_2) は *BCK*–ラムダ項
3. M が *BCK*–ラムダ項であり x が変数で、x が M に高々*1*つしか自由出現しない $\Rightarrow$ $(\lambda x.M)$ は *BCK*–ラムダ項

すなわち、リダクションを行ってもラムダ項がコピーされないようなラムダ項である。

以上を図示すると図 2.2 になる。弱正規性を持たないラムダ項では、どのようにリダクションをしても正規形に辿り着かないため、つまり全てのリダクション戦略が最短である。

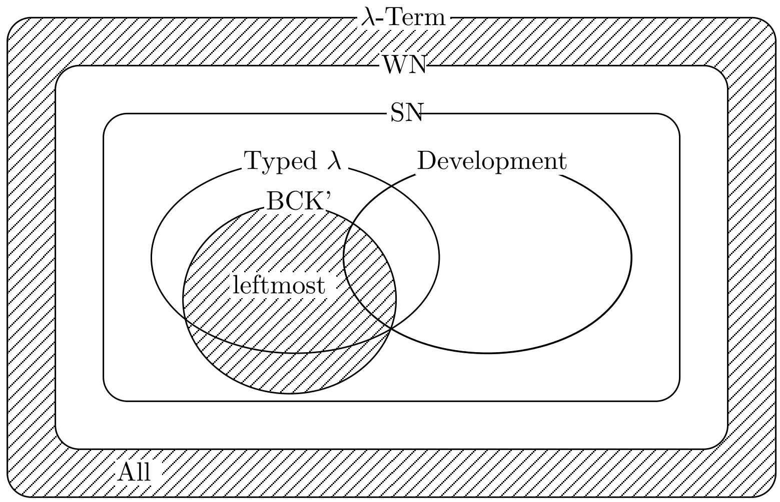


図 2.2: 最短リダクション戦略

ラムダ項が弱正規性を持てば、幅優先探索により最短リダクション戦略を求めることは可能である。しかしながら、その場合は実際にリダクションを行うかそれ以上の手間がかかり実効的[1]とはいえない。我々が欲しいのは実効的なリダクション戦略である。

[1] ここでいう実効的とは実際にリダクションを行うよりは簡単に見い出せるということである。[1] には「比較的簡単に」というあいまいな表現で定義されている。

本章では、ラムダ項の展開において実効的な最短リダクション戦略を与える。同じ証明手法を BCK–ラムダ項に適用し、BCK–ラムダ項に対する最短リダクション戦略の別証明を与える。

2.1.3 証明方法

[35] や [36] によれば、次の二つの関数 h : ラムダ項 $\to$ 自然数 及び $()^*$: ラムダ項 $\to$ ラムダ項 が以下の三つの条件を満たすとき、$()^*$ が最長リダクション戦略を与え、h がそのリダクション回数を与える。

1. M : 正規形 $\Rightarrow h(M) = 0$
2. $M \rhd M' \Rightarrow h(M) > h(M')$
3. M : 正規形ではない $\Rightarrow M \rhd (M)^*$ & $h(M) = h((M)^*) + 1.$

では、逆に最短リダクション戦略ではどうすればいいだろうか。この場合には次の命題を満たす二つの関数 h : ラムダ項 $\to$ 自然数 及び $()^*$: ラムダ項 $\to$ ラムダ項 を見い出せばよい。

命題 2.1.8 関数 h : ラムダ項 $\to$ 自然数 及び $()^*$: ラムダ項 $\to$ ラムダ項 が次の三つの条件を満たすとき、$()^*$ は最短リダクション戦略を与え h はそのステップ数を表す。

1. M : 正規形 $\Leftrightarrow h(M) = 0$

2. $M \rhd M' \Rightarrow h(M) \leq h(M') + 1$

3. M : 正規形ではない $\Rightarrow M \rhd (M)^*$ & $h(M) = h((M)^*) + 1.$

(証明) 今 M において、$h(M)$ よりも短かいステップ数 n で正規形になるリダクション戦略が存在したとする。このリダクション列を $M \equiv M_0 \rhd \cdots \rhd M_n$ とすると、$n < h(M)$。一方、$h(M) = h(M_0) \leq h(M_1)+1 \leq \cdots \leq h(M_n)+n = 0+n = n$ より、$n < n$ となり矛盾。よって、少なくとも $h(M)$ のステップ数で正規形となる。

そして、M に $()^*$ を m 回適用して正規形に辿り着くと仮定する。このとき、$M \rhd (M)^* \rhd \cdots \rhd (M)^{*^m}$ であり、$h(M) = h((M)^*) + 1 = \cdots = h((M)^{*^m}) + m = 0 + m = m$ である。$()^*$ が最短リダクション戦略を与えることになる。

次節から、ラムダ項の展開と BCK–ラムダ項に対する関数 h と $()^*$ を具体的に定義していく。

2.2 ラムダ項の展開における最短リダクション戦略

本節では、ラムダ項の展開における最短リダクション戦略を定義する。ラムダ項の展開とは何か。それは、いくつかのラムダ項のレデックスに印 (λ') を付け、その印の付いたレデックスのみをリダクションするものである。

ラムダ項のチャーチ・ロッサー性を示すときに並行リダクションが用いられるが、ラムダ項の展開とはその 1 ステップの並行リダクションをいわゆるリダクション (あるいはベータリダクション) で捉えたものである。1 ステップの並行リダクションはリダクションでは数ステップになるが、その結果はあるラムダ項の展開の正規形になる。

定義 2.2.1 (λ' 項) Λ をラムダ項全体の集合とする。ラムダ項の展開のラムダ項 *(以下 λ' 項)* の集合 Λ' を以下のように帰納的に定義する。

1. $M \in \Lambda \Rightarrow M \in \Lambda'$
2. $M_1, M_2 \in \Lambda', x \in Var \Rightarrow (\lambda' x.M_1)M_2 \in \Lambda'$

(すなわち、ラムダ項のいくつかのレデックスを $'$ で印付けたものである。)

定義 2.2.2 (λ' リダクション)

1. $(\lambda' x.M_1)M_2 \rhd' M_1[x := M_2]$

   $M_1 \rhd' M_2$ のとき

2. $\lambda x.M_1 \rhd' \lambda x.M_2$
3. $M_1 N \rhd' M_2 N$
4. $NM_1 \rhd' NM_2$

(すなわち、λ' で印付けられたレデックスを一つリダクションした結果である。)

ラムダ項の展開は一般のラムダ項とは異なりラムダ項には含まれないとも考えられる。しかしながら、次のような埋め込み DL を考えることにより、ラムダ項の一部とみなすことができる。

定義 2.2.3 1. $\mathrm{DL}(x) \equiv x$

2. $\mathrm{DL}(MN) \equiv \mathrm{DL}(M)\mathrm{DL}(N)$
3. $\mathrm{DL}(\lambda x'.M) \equiv \lambda x.\mathrm{DL}(M)$
4. $\mathrm{DL}(\lambda x.M) \equiv x'\mathrm{DL}(M)$ *(x' は M に出現しない新しい変数)*

定義 2.2.4 x を束縛変数ではない変数とする。このとき、関数 $\mathrm{VA}_x : \lambda'$ 項 $\rightarrow \{0,1\}$ を以下のように定義する。*(x が束縛変数のときは、x が束縛変数とならないような α 同値な λ' 項と引数を置き換えるものとする。)*

1. $\mathrm{VA}_x(x) = 1$
2. $\mathrm{VA}_x(y) = 0 \ (x \not\equiv y)$
3. $\mathrm{VA}_x((\lambda' y.M_1)M_2) = \mathrm{VA}_x(M_1) \vee (\mathrm{VA}_y(M_1) \wedge \mathrm{VA}_x(M_2))$
4. $\mathrm{VA}_x(M_1 M_2) = \mathrm{VA}_x(M_1) \vee \mathrm{VA}_x(M_2)$ *($(M_1 M_2)$ がレデックスでないとき)*
5. $\mathrm{VA}_x(\lambda y.M_1) = \mathrm{VA}_x(M_1)$

(ここで $A, B \in \{0,1\}$ に対し $A \vee B \equiv A + B - A \times B$, $A \wedge B \equiv A \times B$ と定める。)

どのようにリダクションを行っても M に x が自由出現し続けるとき、$\mathrm{VA}_x(M)$ は 1 となり、いつか x が自由出現しなくなるときには 0 となることを意図して $VA_x(M)$ を定義してある。実際に $(\lambda' y, M_1)M_2$ の場合、M_2 はそのうち y に代入されるが、$(\lambda y'.M_1)M_2$ の中に x が出現し続けるのは、M_1 の中の x が出現し続けるか、あるいは M_1 の中の y が出現し続けるとともに M_2 の中に x が出現し続けるか、のどちらかである。上の項 3 はこのことを表している。

次にどのようにリダクションをしても M に x が出現し続けるならば、実際に $\mathrm{VA}_x(M) = 1$ となることを示す。

補題 2.2.5

$$x \notin \mathrm{FV}(M) \Rightarrow \mathrm{VA}_x(M) = 0$$

(証明) 定義から明らか。

補題 2.2.6 $x \not\equiv y$ とする。

$$\mathrm{VA}_x(M_1[y := M_2]) = \mathrm{VA}_x(M_1) \vee (\mathrm{VA}_y(M_1) \wedge \mathrm{VA}_x(M_2)) \quad (= \mathrm{VA}_x((\lambda' y.M_1)M_2))$$

(証明) M_1 の構造に関する帰納法。$\mathrm{L} = \mathrm{VA}_x(M_1[y := M_2])$ とし $\mathrm{R} = \mathrm{VA}_x(M_1) \vee (\mathrm{VA}_y(M_1) \wedge \mathrm{VA}_x(M_2))$ とする。

1. $M_1 \equiv y$ のとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(M_2) \\
\mathrm{R} &= \mathrm{VA}_x(y) \vee (\mathrm{VA}_y(y) \wedge \mathrm{VA}_x(M_2)) \\
&= 0 \vee (1 \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{VA}_x(M_2)
\end{aligned}$$

2. $M_1 \equiv x$ のとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(x) \\
&= 1 \\
\mathrm{R} &= \mathrm{VA}_x(x) \vee (\mathrm{VA}_y(x) \wedge \mathrm{VA}_x(M_2)) \\
&= 1 \vee (0 \wedge \mathrm{VA}_x(M_2)) \\
&= 1
\end{aligned}$$

3. $M_1 \equiv z$ $(z \not\equiv x, z \not\equiv y)$ のとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(z) \\
&= 0 \\
\mathrm{R} &= \mathrm{VA}_x(z) \vee (\mathrm{VA}_y(z) \wedge \mathrm{VA}_x(M_2) \\
&= 0 \vee (0 \wedge \mathrm{VA}_x(M_2)) \\
&= 0
\end{aligned}$$

4. $M_1 \equiv (\lambda' z.N_1)N_2$ のとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(((\lambda' z.N_1)N_2)[y := M_2]) \\
&= \mathrm{VA}_x((\lambda' z.N_1[y := M_2])N_2[y := M_2]) \\
&= \mathrm{VA}_x(N_1[y := M_2]) \vee (\mathrm{VA}_z(N_1[y := M_2]) \wedge (\mathrm{VA}_x(N_2[y := M_2]))) \quad (\text{定義より}) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(M_2)) \vee ((\mathrm{VA}_z(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \underline{\mathrm{VA}_z(M_2)})) \\
&\qquad \wedge (\mathrm{VA}_x(N_2) \vee (\mathrm{VA}_y(N_2) \wedge \mathrm{VA}_x(M_2)))) \quad (\text{I.H. より}) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(M_2)) \vee (\mathrm{VA}_z(N_1) \wedge (\mathrm{VA}_x(N_2) \vee (\mathrm{VA}_y(N_2) \wedge \mathrm{VA}_x(M_2)))) \\
&\qquad\quad (\mathrm{VA}_z(M_2) = 0 \text{ より}) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(M_2)) \vee (\mathrm{VA}_z(N_1) \wedge \mathrm{VA}_x(N_2)) \vee (\mathrm{VA}_z(N_1) \wedge \mathrm{VA}_y(N_2) \\
&\qquad \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_z(N_1) \wedge \mathrm{VA}_x(N_2)) \vee (\mathrm{VA}_y(N_1) \vee (\mathrm{VA}_z(N_1) \wedge \mathrm{VA}_y(N_2)) \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{VA}_x((\lambda' z.N_1)N_2) \vee (\mathrm{VA}_y((\lambda' z.N_1)N_2) \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{R}
\end{aligned}$$

5. $M_1 \equiv N_1N_2$ で N_1N_2 はレデックスでないとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(N_1[y := M_2]N_2[y := M_2]) \\
&= \mathrm{VA}_x(N_1[y := M_2]) \vee \mathrm{VA}_x(N_2[y := M_2]) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(M_2)) \vee \mathrm{VA}_x(N_2) \vee (\mathrm{VA}_y(N_2) \wedge \mathrm{VA}_x(M_2))(\text{By I.H.}) \\
&= (\mathrm{VA}_x(N_1) \vee \mathrm{VA}_x(N_2)) \vee ((\mathrm{VA}_y(N_1) \vee \mathrm{VA}_y(N_2)) \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{VA}_x(N_1N_2) \vee (\mathrm{VA}_y(N_1N_2) \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{R}
\end{aligned}$$

6. $M_1 \equiv (\lambda z.N_1)$ のとき。

$$\begin{aligned}
\mathrm{L} &= \mathrm{VA}_x(\lambda z.N_1[y := M_2]) \\
&= \mathrm{VA}_x(N_1[y := M_2]) \\
&= \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(M_2))(\text{By I.H.}) \\
&= \mathrm{VA}_x(\lambda z.N_1) \vee (\mathrm{VA}_y(\lambda z.N_1) \wedge \mathrm{VA}_x(M_2)) \\
&= \mathrm{R}
\end{aligned}$$

補題 2.2.7

$$M \rhd' M' \Rightarrow \mathrm{VA}_x(M) = \mathrm{VA}_x(M)$$

(証明) 前補題より明らか。

定理 2.2.8 M' を正規形とし、$M \rhd'_\beta M'$ とする。
$\mathrm{VA}_x(M) = 0 \Rightarrow x \notin \mathrm{FV}(M')$

(証明) 前補題と定義 2.2.4 より明らか。

この VA_x を用いて、$h : \lambda'$項 $\to$ 自然数 を次のように定義する。この h がリダクションの最短ステップ数を与えることになる。

定義 2.2.9 (最短リダクションステップの関数) 以下のように $h:\lambda'$項 $\to$ 自然数 を定義する。

1. $h(x) = 0$
2. $h((\lambda' x.M_1)M_2) = h(M_1) + \mathrm{VA}_x(M_1) \times h(M_2) + 1$
3. $h(M_1M_2) = h(M_1) + h(M_2)$ *((M_1M_2) がレデックスでないとき)*
4. $h(\lambda x.M_1) = h(M_1)$

条項 2 の直観的意味付けは次のようになる。x が M_1 に出現し続けるときには、M_2 もまた出現し続けるであろうから、M_2 のリダクションステップ数をカウントする。x が M_1 でいつか消えるときには、M_2 もいつか消えるのでカウントしない。実際この関数 h が最小リダクションステップ数を与えることを見ていく。

補題 2.2.10

$h(M) = 0 \Leftrightarrow M$ が正規形。

(証明) 定義から明らか。

補題 2.2.11
$h(M_2) = 0 \Rightarrow h(M_1[x := M_2]) = h(M_1)$

(証明) M_1 の構造に関する帰納法。ここでは $M_1 \equiv (\lambda' y.N_1)N_2$ の場合と、$M_1 \equiv N_1N_2$ で M_1 がレデックスではない場合を示す。

$$\begin{aligned}
h(((\lambda' y.N_1)N_2)[x := M_2]) &= h((\lambda' y.N_1[x := M_2])N_2[x := M_2]) \\
&= h(N_1[x := M_2]) + \mathrm{VA}_y(M_1) \times h(N_2[x := M_2]) + 1 \\
&= h(N_1) + \mathrm{VA}_y(M_1) \times h(N_2) + 1 (\text{I.H. より}) \\
&= h((\lambda' y.N_1)N_2)
\end{aligned}$$

$$\begin{aligned}
h((N_1N_2)[x := M_2]) &= h(N_1[x := M_2]N_2[x := M_2]) \\
&= h(N_1[x := M_2]) + h(N_2[x := M_2]) \\
&\qquad (\text{なぜなら } N_1[x := M_2]N_2[x := M_2] \text{ もまたレデックスではないから}) \\
&= h(N_1) + h(N_2) (\text{I.H. より}) \\
&= h(N_1N_2)
\end{aligned}$$

この補題こそがラムダ項の展開に関しては本質的に効いてくる。この補題が成り立つことにより、ラムダ項の展開では最短リダクション戦略の証明が容易となる。$h(M) = 0$ ならば M は正規形であり、この補題から、正規形に正規形を代入した λ' 項はまた正規形となることが直ちに分かる。このように一般のラムダ項ではこのことが成り立たない。例えば、xy と $(\lambda x.x)$ は正規形であるが、$(xy)[x := (\lambda x.x)] \equiv (\lambda x.x)y$ となり正規形ではなくなる。一般のラムダ項では、いわば関数を代入すると新たなレデックスを生じることがある。

補題 2.2.12 $\mathrm{VA}_x(M_1) = 0$ とし x を M_1 の束縛変数ではないとする。このとき、

$$h(M_1[x := M_2]) = h(M_1)$$

(証明) M_1 の構造に関する帰納法。$\mathrm{L} = h(M_1[x := M_2])$ とし $\mathrm{R} = h(M_1)$ とする。

1. $M_1 \equiv y$ $(x \not\equiv y)$ のとき。

$$\begin{aligned}\mathrm{L} &= h(y)\\ &= \mathrm{R}\end{aligned}$$

2. $M_1 \equiv (\lambda' y.N_1)N_2$ $(x \not\equiv y)$ のとき。

$$\begin{aligned}\mathrm{VA}_x((\lambda' y.N_1)N_2) = 0 &\Leftrightarrow \mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(N_2)) = 0\\ &\Leftrightarrow \mathrm{VA}_x(N_1) = 0 \text{ かつ } (\mathrm{VA}_y(N_1) = 0 \text{ or } \mathrm{VA}_x(N_2) = 0)\end{aligned}$$

$$\begin{aligned}\mathrm{L} &= h((\lambda' y.N_1[x := M_2])N_2[x := M_2])\\ &= h(N_1[x := M_2]) + \mathrm{VA}_y(N_1[x := M_2]) \times h(N_2[x := M_2]) + 1\\ &= h(M_1) + (\mathrm{VA}_y(N_1) \vee (\mathrm{VA}_x(N_1) \wedge \mathrm{VA}_y(M_2))) \times h(N_2[x := M_2]) + 1 (\text{I.H. より})\\ &= h(M_1) + (\mathrm{VA}_y(N_1) \times h(N_2[x := M_2]) + 1 (\mathrm{VA}_x(N_1) = 0 \text{ より})\end{aligned}$$

$\mathrm{VA}_y(N_1) = 0$ のとき。

$$\begin{aligned}\mathrm{L} &= h(M_1) + 0 \times h(N_2[x := M_2]) + 1\\ &= h(M_1) + 0 \times h(N_2) + 1\\ &= h(M_1) + (\mathrm{VA}_y(N_1) \times h(N_2)) + 1\\ &= h((\lambda' y.N_1)N_2)\end{aligned}$$

$\mathrm{VA}_x(N_2) = 0$ のとき。

$$\begin{aligned}\mathrm{L} &= h(M_1) + (\mathrm{VA}_y(N_1) \times h(N_2) + 1 (\text{I.H. より})\\ &= h((\lambda' y.N_1)N_2)\end{aligned}$$

3. $M_1 \equiv (N_1 N_2)$ で $(N_1 N_2)$ がレデックスでないとき。

$$\mathrm{VA}_x(N_1 N_2) = 0 \Leftrightarrow \mathrm{VA}_x(N_1) = 0 \text{ かつ } \mathrm{VA}_x(N_2) = 0$$

$$\begin{aligned}\mathrm{L} &= h(N_1[x := M_2]N_2[x := M_2])\\ &= h(N_1[x := M_2]) + h(N_2[x := M_2])\\ &= h(N_1) + h(N_2) (\text{I.H. より})\\ &= h(N_1 N_2)\end{aligned}$$

4. $M_1 \equiv (\lambda y.N_1)$ のとき。

$$\mathrm{VA}_x(\lambda y.N_1) = 0 \Leftrightarrow \mathrm{VA}_x(N_1) = 0$$

$$\begin{aligned}\mathrm{L} &= h(\lambda y.N_1[x := M_2])\\ &= h(N_1[x := M_2])\\ &= h(N_1) (\text{I.H. より})\\ &= h(\lambda y.N_1)\end{aligned}$$

補題 2.2.13 x を M_1 の束縛変数ではないとする。

$$h(M_1[x := M_2]) \geq h(M_1) + \mathrm{VA}_x(M_1) \times h(M_2) \ \ (= h((\lambda' x.M_1)M_2) - 1)$$

(証明) M_1 の構造に関する帰納法。$\mathrm{L} = h(M_1[x := M_2])$ とし $\mathrm{R} = h(M_1) + \mathrm{VA}_x(M_1) \times h(M_2)$ とする。

1. $M_1 \equiv x$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(M_2) \\
\mathrm{R} &= h(x) + \mathrm{VA}_x(x) \times h(M_2) \\
&= 0 + 1 \times h(M_2) \\
&= h(M_2)
\end{aligned}$$

2. $M_1 \equiv y$ $(x \not\equiv y)$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(y) \\
&= 0 \\
\mathrm{R} &= h(y) + \mathrm{VA}_x(y) \times h(M_2) \\
&= 0 + 0 \times h(M_2) \\
&= 0
\end{aligned}$$

3. $M_1 \equiv (\lambda' y.N_1)N_2$ のとき。

$$\begin{aligned}
\mathrm{L} &= h((\lambda' y.N_1[x := M_2])N_2[x := M_2]) \\
&= h(N_1[x := M_2]) + \mathrm{VA}_y(N_1[x := M_2])h(N_2[x := M_2]) + 1 \\
&\geq h(N_1) + \mathrm{VA}_x(N_1)h(M_2) + (\mathrm{VA}_y(N_1) \vee (\mathrm{VA}_x(N_1) \wedge \mathrm{VA}_y(M_2)))(h(N_2) + \mathrm{VA}_x(N_2)h(M_2)) + 1 \\
&\quad (\text{I.H. より}) \\
&= h(N_1) + \mathrm{VA}_x(N_1)h(M_2) + \mathrm{VA}_y(N_1)(h(N_2) + \mathrm{VA}_x(N_2)h(M_2)) + 1 (\text{By } \mathrm{VA}_y(M_2) = 0) \\
&= \mathrm{L}' \\
\mathrm{R} &= h((\lambda' y.N_1)N_2) + \mathrm{VA}_x((\lambda' y.N_1)N_2)h(M_2) \\
&= h(N_1) + \mathrm{VA}_y(N_1)h(N_2) + 1 + (\mathrm{VA}_x(N_1) \vee (\mathrm{VA}_y(N_1) \wedge \mathrm{VA}_x(N_2)))h(M_2) \\
&= h(N_1) + \mathrm{VA}_y(N_1)h(N_2) + (\mathrm{VA}_x(N_1) + \mathrm{VA}_y(N_1)\mathrm{VA}_x(N_2) - \mathrm{VA}_x(N_1)\mathrm{VA}_y(N_1)\mathrm{VA}_x(N_2)) \\
&\qquad \times h(M_2) + 1 \\
&= h(N_1) + \mathrm{VA}_x(N_1)h(M_2) + \mathrm{VA}_y(N_1)(h(N_2) + \mathrm{VA}_x(N_2)h(M_2) - \mathrm{VA}_x(N_1)\mathrm{VA}_x(N_2)h(M_2)) + 1 \\
\mathrm{L}' - \mathrm{R} &= \mathrm{VA}_y(N_1)\mathrm{VA}_x(N_1)\mathrm{VA}_x(N_2)h(M_2) \geq 0
\end{aligned}$$

4. $M_1 \equiv (N_1N_2)$ ((N_1N_2) はレデックスでない) のとき。

$$\begin{aligned}
\mathrm{L} &= h(N_1[x := M_2]N_2[x := M_2]) \\
&= h(N_1[x := M_2]) + h(N_2[x := M_2]) \\
&\geq h(N_1) + \mathrm{VA}_x(N_1)h(M_2) + h(N_2) + \mathrm{VA}_x(N_2)h(M_2) (\text{I.H. より}) \\
&= h(N_1N_2) + \mathrm{VA}_x(N_1N_2)h(M_2) \\
&= \mathrm{R}
\end{aligned}$$

5. $M_1 \equiv (\lambda y.N_1)$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(\lambda y.N_1[x := M_2]) \\
& h(N_1[x := M_2]) \\
& \geq h(N_1) + \mathrm{VA}_x(N_1)h(M_2)(\text{I.H. より}) \\
& = h(\lambda y.N_1) + \mathrm{VA}_x(\lambda y.N_1)h(M_2) \\
& = \mathrm{R}
\end{aligned}$$

補題 2.2.14 (主補題 1)

$M \rhd' M' \Rightarrow h(M) \leq h(M') + 1$

(証明) M の構成に関する帰納法。$\mathrm{L} = h(M)$ とし $\mathrm{R} = h(M') + 1$ とする。

1. $M \equiv (\lambda' x.N_1)N_2, M' \equiv N_1[x := N_2]$ のとき。

$$\begin{aligned}
\mathrm{L} &= h((\lambda' x.N_1)N_2) \\
&= h(N_1) + \mathrm{VA}_x(N_1)h(N_2) + 1 \\
&\leq h(N_1[x := N_2]) + 1(\text{前補題より}) \\
&= \mathrm{R}
\end{aligned}$$

2. $M \equiv (\lambda' x.N_1)N_2, N_1 \rhd' N_1' M' \equiv (\lambda' x.N_1')N_2$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(N_1) + \mathrm{VA}_x(N_1)h(N_2) + 1 \\
&\leq h(N_1') + 1 + \mathrm{VA}_x(N_1)h(N_2) + 1(\text{I.H. より}) \\
&= (\lambda' x.N_1')N_2 + 1 \\
&= \mathrm{R}
\end{aligned}$$

3. $M \equiv (\lambda' x.N_1)N_2, N_2 \rhd' N_2' M' \equiv (\lambda' x.N_1)N_2'$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(N_1) + \mathrm{VA}_x(N_1)h(N_2) + 1 \\
&\leq h(N_1) + 1 + \mathrm{VA}_x(N_1)h(N_2') + 1(\text{I.H. より}) \\
&= (\lambda' x.N_1)N_2' + 1 \\
&= \mathrm{R}
\end{aligned}$$

4. $M \equiv N_1 N_2$ で $N_1 N_2$ がレデックスでなく、さらに $N_1 \rhd' N_1' M' \equiv N_1' N_2$ のとき。

$$\begin{aligned}
\mathrm{L} &= h(N_1) + h(N_2) \\
&\leq h(N_1') + 1 + h(N_2)(\text{I.H. より}) \\
&= h(N_1' N_2) + 1 \\
&= \mathrm{R}
\end{aligned}$$

5. $M \equiv N_1 N_2$ で $N_1 N_2$ がレデックスでなく、さらに $N_2 \rhd' N_2' M' \equiv N_1 N_2'$ のとき。前の場合と同様。

6. $M \equiv (\lambda x.N_1), N_1 \rhd' N_1' M' \equiv (\lambda x.N_1')$ のとき。

$$
\begin{aligned}
\mathrm{L} &= h(N_1) \\
&\leq h(N_1') + 1 (\text{I.H. より}) \\
&= h(\lambda x.N_1') + 1 \\
&= \mathrm{R}
\end{aligned}
$$

以上で h が命題 2.1.8 の条項 2 が成り立つことを示した。次は条項 3 を満たす関数 $()^*$ の導入である。
[Semi C.b.V. リダクション戦略]

定義 2.2.15 ($*$-操作) M を正規形ではないとする。このとき、$(M)^*$ を以下のように定義する。

1. $M \equiv (\lambda' x.M_1)M_2$ のとき。

(a) $\mathrm{VA}_x(M_1) = 0$ のとき。
$M^* \equiv M_1[x := M_2]$

(b) $\mathrm{VA}_x(M_1) = 1$ のとき。

i. M_2 が正規形でないとき。
$M^* \equiv (\lambda' x.M_1)(M_2)^*$

ii. M_2 が正規形のとき。
$M^* \equiv M_1[x := M_2]$

2. $M \equiv M_1 M_2$ で $M_1 M_2$ はレデックスでないとき。

(a) M_1 が正規形でないとき。
$M^* \equiv (M_1)^* M_2$

(b) M_1 が正規形のとき。
$M^* \equiv M_1(M_2)^*$

3. $M \equiv \lambda x.M_1$ のとき。
$M^* \equiv \lambda x.(M_1)^*$

このリダクション戦略を、偽値呼び出しリダクション戦略と呼ぶ。

補題 2.2.16 M を正規形ではないとする。

$M \rhd' (M)^*$

(証明) 定義より明らか。

補題 2.2.17 (主補題 2) M を正規形ではないとする。

$$h(M) = h((M)^*) + 1$$

(証明) M の構成に関する帰納法。

1. $M \equiv (\lambda' x.M_1)M_2$ のとき。

(a) $\mathrm{VA}_x(M_1) = 0$ のとき。$(M)^* \equiv M_1[x := M_2]$ であり、

$$\begin{aligned}
h(M) &= h(M_1) + M_x(M_1)h(M_2) + 1 \\
&= h(M_1) + 1 \\
&= h(M_1[x := M_2]) + 1 (\text{補題 } 2.2.12 \text{ より}) \\
&= h((M)^*) + 1
\end{aligned}$$

(b) $\mathrm{VA}_x(M_1) = 1$ のとき。

i. M_2 が正規形のとき。$(M)^* \equiv \mathrm{VA}_x[x := M_2]$ であり、

$$\begin{aligned}
h(M) &= h(M_1) + M_x(M_1)h(M_2) + 1 \\
&= h(M_1) + 1 \\
&= h(M_1[x := M_2]) + 1 (\text{補題 } 2.2.11 \text{ より}) \\
&= h((M)^*) + 1
\end{aligned}$$

ii. M_2 が正規形でないとき。$(M)^* \equiv (\lambda' x.M_1)(M_2)^*$ であり、

$$\begin{aligned}
h(M) &= h(M_1) + M_x(M_1)h(M_2) + 1 \\
&= h(M_1) + M_x(M_1)(h((M_2)^*) + 1) + 1 (\text{I.H. より}) \\
&= h(M_1) + M_x(M_1)h((M_2)^*) + 1 + \mathrm{VA}_x(M_1) \\
&= h((\lambda' x.M_1)(M_2)^*) + 1 (\mathrm{VA}_x(M_1) = 0) \text{ より} \\
&= h((M)^*) + 1
\end{aligned}$$

2. $M \equiv M_1 M_2$ で $M_1 M_2$ はレデックスでないとき。

(a) M_1 が正規形でないとき。$(M)^* \equiv (M_1)^* M_2$ であり、

$$\begin{aligned}
h(M) &= h(M_1) + h(M_2) \\
&= h((M_1)^*) + 1 + h(M_2) (\text{I.H. より}) \\
&= h((M_1)^* M_2) + 1 \\
&= h((M)^*) + 1
\end{aligned}$$

(b) M_1 が正規形のとき。$(M)^* \equiv M_1(M_2)^*$ であり、

$$\begin{aligned}
h(M) &= h(M_1) + h(M_2) \\
&= h(M_1) + h((M_2)^*) + 1 (\text{I.H. より}) \\
&= h(M_1(M_2)^*) + 1 \\
&= h((M)^*) + 1
\end{aligned}$$

3. $M \equiv (\lambda x.M_1)$ のとき。$(M)^* \equiv \lambda x.(M_1)^*$ であり、

$$\begin{aligned}
h(M) &= h(M_1) \\
&= h((M_1)^*) + 1 (\text{I.H. より}) \\
&= h(\lambda x.(M_1)^*) + 1 \\
&= h((M)^*) + 1
\end{aligned}$$

定理 2.2.18 (最短リダクション戦略) M が正規形ではないとき、$(M)^*$ は最短リダクション戦略となる。

(証明) 次のリダクション列を考える。

$$M \rhd' (M)^* \rhd' ((M)^*)^* \cdots \rhd' (\ldots((M)^*)\ldots)^*$$

$()^*$ を適用する度に h の値は 1 だけ小さくなる。正規形では h は 0 であるから、M に $()^*$ は高々$h(M)$ 回適用可能である。$M \equiv M_0, (M)^* \equiv M_1, \ldots, (\ldots((M)^*)\ldots)^* \equiv M_m (M_m$ は正規形) とおくと、

$$M_0 \rhd' M_1 \rhd' \cdots \rhd' M_m$$

これは、m 回で正規形に辿りつくリダクション列である。今 m より小さい n ステップで正規形に辿りつくリダクション列があったとする。すなわち

$$M \equiv N_0 \rhd' N_1 \rhd' \cdots \rhd' N_n. (n < m)$$

であるが、

$$m = h(M) = h(N_0) \leq_g h(N_1) + 1 \leq \cdots \leq h(N_m) + n = n \Rightarrow m \leq n.$$

となり矛盾する。故に $(M)^*$ は最短リダクション戦略である。

2.3 BCK–ラムダ項における最短リダクション戦略

本節では、BCK–ラムダ項における最短リダクション戦略を考える。

2.3.1 背景

BCK–ラムダ項は前章の 1.3.3 節で定義したラムダ項のうち Contraction のない体系に対応したラムダ項である。この体系をコンビネータで捉えると、B,C,K の三つのコンビネータで表わせるので、BCK–ラムダ項と呼ばれ [13] で導入された。形式的定義は次のようになる。

定義 2.3.1 (BCK–ラムダ項)

1. x が変数 $\Rightarrow$ x は *BCK*–ラムダ項
2. M_1, M_2 が *BCK*–ラムダ項であり $\mathrm{FV}(M_1) \cap \mathrm{FV}(M_2) = \emptyset \Rightarrow (M_1 M_2)$ は *BCK*–ラムダ項
3. M が *BCK*–ラムダ項であり x が変数 $\Rightarrow$ $(\lambda x.M)$ は *BCK*–ラムダ項

このとき、次の性質を持つ。

定理 2.3.2 全ての *BCK*–ラムダ項は型を持つ。

(証明) [13] を参照。

すなわち、BCK–ラムダ項は型付きラムダ項でもある。一般の型付きラムダ項全体に対する最短リダクション戦略は複雑になりすぎて手に負えないので、それらのうち BCK–ラムダ項について最短リダクション戦略を考察することにする。まず、いくつかの定義を行う。

2.3.2 ラベル付き関数族

ラムダ項は値としての性質と関数としての性質を合わせ持つという二面性を有している。このことが、ラムダ項のモデルの構成を難しくしている。

では、型付きラムダ項ではどうなるであろうか。この場合には一つ大きな違いがある。それは型付きラムダ項では、アトミックな型を持つときには関数として振る舞わないということである。M^a(a はアトミック) のときには、MN となる N は存在しない。このことを利用すると、アトミックなラムダ項を出発点にして、一種のモデルを組み立てることが可能となる。それが [36] で導入されたラベル付き関数族 (labelled functionals) である。

定義 2.3.3 (ラベル付き関数族) a を型変数とし α, β を型とする。型が α であるラベル付き関数系の集合 L_α を以下のように定義する。

1. $L_a = \omega$
2. $L_{\alpha\to\beta} = (L_\alpha \to L_\beta) \times \omega.$

*(*但し ω は自然数全体の集合であり、$\times$ は直積である。また、$L_\alpha \to L_\beta$ は L_α から L_β への関数全体の集合を表す。*)*

そして、ラムダ項からこの関数族への解釈を考える。この解釈を用いて、あるリダクション戦略が最短となることを証明する。

表記 2.3.4 $f = \langle f', m\rangle \in L_{\alpha\to\beta}$ に対し、関数族上の操作 $*$ を $f* = m$ 定める。さらに、これを自然数上にも拡張して $m* = m$ とする。また、$g \in L_\alpha$ に対し fg を $f'g$ により定める。L は関数族全体の集合を表すものとする。つまり $L = \cup_\alpha L_\alpha$ である。一般に fg と書いたときには、ある α, β が存在して、$f \in L_{\alpha\to\beta}$ 及び $g \in L_\alpha$ が成り立つものとする。

次に関数族に自然数を加える操作を定義する。

定義 2.3.5 (+ 操作) $n \in \omega$ とし、$f +_\alpha n \in L_\alpha$ を次のように定義する。

1. $m +_a n = m + n$
2. $f +_{\beta\to\gamma} n = \langle \Lambda g \in L_\beta . fg +_\gamma n, f * + n\rangle$

単に $+$ と書いたときは $+ = \cup_\alpha +_\alpha$ である。

補題 2.3.6

1. $(f + n)g = fg + n$
2. $(f + n)* = f * + n$
3. $f + 0 = f$
4. $(f + m) + n = f + (m + n)$

(証明)

$$
\begin{aligned}
1.\ (f+n)g &= \langle \Lambda h \in L_\alpha, fh +_\beta n, f * + n\rangle g \\
&= (\Lambda h \in L_\alpha, fh +_\beta n) g \\
&= fg + n
\end{aligned}
$$

$$
\begin{aligned}
4.\ f + m + n &= \langle \Lambda g \in L_\alpha, fg +_\beta m, f * + m\rangle + n \\
&= \langle \Lambda g \in L_\alpha, (fg +_\beta m) +_\beta n, (f * + m) + n\rangle \\
&= \langle \Lambda g \in L_\alpha, fg +_\beta (m+n), f * + (m+n)\rangle \qquad \text{(I.H. より)} \\
&= f + (m+n)
\end{aligned}
$$

定義 2.3.7 (最小の累積関数) 任意の $n \in \omega$ に対して**最小の累積関数 (minimally cumulative functionals)** $c_n^\alpha \in L_\alpha$ を以下のように α に対して帰納的に定義する。

1. $c_n^a = n$
2. $c_n^{\alpha\to\beta} = \langle \Lambda f \in L_\alpha . c^\beta_{n+f*}, n\rangle$

補題 2.3.8

1. $c_n^\alpha f_1 \dots f_m = c^\alpha_{n+f_1*+\dots f_m*}$
2. $(c_0^\alpha f_1 \dots f_m)* = f_1 + \dots f_m$
3. $c_n^\alpha + m = c^\alpha_{n+m}$

(証明) 1 と 2 は定義より明らか。3 は α の構成に関する帰納法。

$$\begin{aligned} c_n^{\alpha\to\beta} + m &= \langle \Lambda f \in L_\alpha . c^\beta_{n+f*}, n \rangle + m \\ &= \langle \Lambda f \in L_\alpha . c^\beta_{n+f*} + m, n+m \rangle \\ &= \langle \Lambda f \in L_\alpha . c^\beta_{n+f*+m}, n+m \rangle && \text{(I.H. より)} \\ &= \langle \Lambda f \in L_\alpha . c^\beta_{n+m+f*}, n+m \rangle \\ &= c^{\alpha\to\beta}_{n+m} \end{aligned}$$

これらの定義を用いて、BCK–ラムダ項の関数族上への解釈を定義していく。

2.3.3 BCK–ラムダ項の解釈

定義 2.3.9 (割り当て) 任意の x^α から L_α 上への割り当てが v で与えられているとする。このとき $v(x/f)$ は、$v(x/f)(x) = f$ であり $v(y/f)(x) = v(x)\ (x \not\equiv y)$ であるとする。また、$v(x/f)(x/g)(x) = v(x/g)(x) = g$ であるとする。

定義 2.3.10 (解釈) 任意の割り当て v に対して、BCK–ラムダ項の解釈 $[M^\alpha]_v \in L_\alpha$ を以下のように定義する。

1. $[x]_v = v(x)$

2. $[M_1 M_2]_v = [M_1]_v [M_2]_v$

3. $[\lambda x^\alpha . M_1]_v = \langle \Lambda f \in L_\alpha . [M_1]_v + 1, [M_1]_v * \rangle \quad (x \notin \mathrm{FV}(M_1))$

4. $[\lambda x^\alpha . M_1]_v = \langle \Lambda f \in L_\alpha . [M_1]_{v(x/f))} + 1, [M_1]_{v(x/c_0^\alpha)} * \rangle \quad (x \in \mathrm{FV}(M_1))$

この解釈が、命題 2.1.8 の h に一致すること、すなわち $\underline{\forall x (v(x) = c_0) \Rightarrow h(M) = [M]_v *}$ が成り立つことをこれから証明していく。先走っていってしまえば、BCK–ラムダ項では最左リダクション戦略が最短となり、この解釈は任意の BCK–ラムダ項に対して最左リダクション戦略を施したときのリダクションステップ数を数え上げる関数になっている。

補題 2.3.11 $x^\alpha \notin \mathrm{FV}(M)$ のとき、

$$\forall f \in L_\alpha ([M]_v = [M]_{v(x/f)})$$

(証明) M の構成に関する帰納法。$M \equiv \lambda x . M_1$ の場合のみを示す。但し $y \in \mathrm{FV}(M_1)$ かつ $x \not\equiv y$ とする。

$$\begin{aligned} [\lambda y . M_1]_v &= \langle \Lambda g . [M_1]_{v(y/g)} + 1, [M_1]_{v(y/c_0)} \rangle \\ &= \langle \Lambda g . [M_1]_{v(y/g)(x/f)} + 1, [M_1]_{v(y/c_0)(x/f)} \rangle && \text{(I.H. より)} \\ &= \langle \Lambda g . [M_1]_{v(x/f)(y/g)} + 1, [M_1]_{v(x/f)(y/c_0)} \rangle && (x \not\equiv y \text{ より}) \\ &= [\lambda y . M_1]_{v(x/f)} \end{aligned}$$

この補題は、定義 2.3.10 における条項 3 を次の 3’ ように条項 4 と同じ形に置き換えてもよいことを示している。

3’. $[\lambda x . M_1]_v = \langle \Lambda f \in L_\alpha . [M_1]_{v(x/f)} + 1, [M_1]_{v(x/c_0^\alpha)} * \rangle \quad (x \notin \mathrm{FV}(M_1))$

補題 2.3.12 (代入)

$$[M[x := N]]_v = [M]_{v(x/[N]_v)}$$

(証明) M の構成に関する帰納法。$M \equiv (\lambda y^\alpha.M_1)$ で $y \in \mathrm{FV}(M_1))$ の場合のみを示す。但し $y \notin \mathrm{FV}(N)$ とする。

$$
\begin{aligned}
&[(\lambda y.M_1)[x := s]]_v \\
&= [\lambda y.M_1[x := s]]_v \\
&= \langle \Lambda f.[M_1[x := s]]_{v(y/f)} + 1, [M_1[x := s]]_{v(y/c_0)}*\rangle \\
&= \langle \Lambda f.[M_1]_{v(y/f))(x/[s]_{v(y/f)})} + 1, [M_1]_{v(y/c_0)(x/[s]_{v(y/c_0)})}*\rangle \quad \text{(I.H. より)} \\
&= \langle \Lambda f.[M_1]_{v(y/f)(x/[s]_v)} + 1, [M_1]_{v(y/c_0)(x/[s]_v)}*\rangle \quad (y \notin \mathrm{FV}(s) \text{ より}) \\
&= [\lambda y.M_1]_{v(x/[s]_v)}
\end{aligned}
$$

2.3.4 遺伝的単調関数

2.3.3 で導入された解釈が命題 2.1.8 の h の条件、

$$M \rhd M' \Rightarrow h(M) \leq h(M') + 1$$

を満たすことを証明するために、関数族 L 上に関係 $\leq$ を導入する。関数族 L 全体に関係 $\leq$ を導入することはうまくいかないので、L_α の部分集合である遺伝的単調関数 (hereditarity monotone functionals) 上に関係 $\leq$ を導入する。

定義 2.3.13 L_α の部分集合である遺伝的単調関数 C_α とその上の関係 $\leq_\alpha$ を以下のように同時に定義する。

1. $C_a = L_a$ とし $m \leq_a n \Leftrightarrow m \leq n$
2. $C_{\alpha\to\beta}$ を次の *2a*〜*2c* を満たすような $f(\in L_{\alpha\to\beta})$ 全体の集合とする。
   (a) $\forall g \in C_\alpha (fg \in C_\beta)$
   (b) $\forall g, g' \in C_\alpha (g \leq_\alpha g' \Rightarrow fg \leq_\beta fg')$ (単調性)
   (c) $\forall g \in C_\alpha (f(g + n) \leq fg + n)$
3. $\forall f, f' \in C_{\alpha\to\beta}$ ($f \leq_{\alpha\to\beta} f' \Leftrightarrow \forall g \in C_\alpha fg \leq_\beta f'g \wedge f* \leq f'*$)

補題 2.3.14 (C_α の性質)

1. $\forall f \in C_\alpha, \forall m \geq 0 (f + m \in C_\alpha)$ (C は $+$ の下で閉じている)
2. $\forall f, f' \in C_\alpha, \forall m \in \omega (f \leq_\alpha f' \Rightarrow f + m \leq_\alpha f' + m)$
3. $m \leq m' \Rightarrow \forall f \in C_\alpha (f + m \leq_\alpha f + m')$
4. $\forall f \in C_\alpha, \forall m \geq 0 (f \leq_\alpha f + m)$
5. $\forall f \in C_\alpha (f \leq_\alpha f)$

(証明) 1 と 2 は α に関する同時帰納法。3 は α に関する帰納法。4 と 5 は 3 から得られる。

補題 2.3.15

1. $\forall \alpha, \forall n \in \omega (c_n^\alpha \in C_\alpha)$
2. $n \leq n' \Rightarrow c_n^\alpha \leq_\alpha c_{n'}^\alpha$

(証明) α に関する帰納法。

定義 2.3.16 (C-割り当て) $\forall x^\alpha(v(x) \in C_\alpha)$ のとき、割り当て v は C-割り当てであるという。

補題 2.3.17 (C–補題) 任意の C-割り当て v に関して、

1. $[M]_v \in C$

2. $f \leq f' \Rightarrow [M]_{v(x/f)} \leq [M]_{v(x/f')}$

3. $[M]_{v(x/f+n)} \leq [M]_{v(x/f)} + n$

(証明) M に関する同時帰納法。ここでは $M \equiv (\lambda y.M_1)$ $(y \in \mathrm{FV}(M_1))$ の場合のみを示す。

1. $[(\lambda y.M_1)^{\beta\to\gamma}]_v \equiv \langle \Lambda g^\beta.[M_1^\gamma]_{v(x/g)} + 1, [M_1^\gamma]_{v(x/c_0)}*\rangle$

2a. $\forall g \in C_\beta([(\lambda x.M_1)]_v g = [M_1^\gamma]_{v(x/g)} + 1 \in C^\gamma)$

2b. $\forall g, g' \in C_\beta(g \leq g')$ $[(\lambda x.M_1)]_v g = [M_1^\gamma]_{v(x/g)} + 1 \leq [M_1^\gamma]_{v(x/g')} + 1 = [(\lambda x.M_1)]_v g'$ (I.H. より)

2c. $\forall g \in \beta([(\lambda x.M_1)]_v(g + m) = [M_1^\gamma]_{v(x/g+m)} + 1 \leq [M_1^\gamma]_{v(x/g)} + m + 1 = [M_1^\gamma]_{v(x/g)} + 1 + m = [(\lambda x.M_1)]_v g + m)$

2. $f \leq f'$ とすると、

$$\begin{aligned}
[\lambda y.M_1]_{v(x/f)} &= \langle \Lambda g.[M_1]_{v(x/f)(y/g)} + 1.[M_1]_{v(x/f)(y/c_0)}*\rangle \\
&= \langle \Lambda g.[M_1]_{v(y/g)(x/f)} + 1.[M_1]_{v(y/c_0)(x/f)}*\rangle \\
&\leq \langle \Lambda g.[M_1]_{v(y/g)(x/f')} + 1.[M_1]_{v(y/c_0)(x/f')}*\rangle \quad \text{(I.H. より)} \\
&= \langle \Lambda g.[M_1]_{v(x/f')(y/g)} + 1.[M_1]_{v(x/f')(y/c_0)}*\rangle \\
&= [\lambda y.M_1]_{v(x/f')}
\end{aligned}$$

3.

$$\begin{aligned}
[\lambda y.M_1]_{v(x/f+n)} &= \langle \Lambda g.[M_1]_{v(x/f+n)(y/g)} + 1.[M_1]_{v(x/f+n)(y/c_0)}*\rangle \\
&= \langle \Lambda g.[M_1]_{v(y/g)(x/f+n)} + 1.[M_1]_{v(y/c_0)(x/f+n)}*\rangle \\
&\leq \langle \Lambda g.[M_1]_{v(y/g)(x/f)} + 1 + n.([M_1]_{v(y/c_0)(x/f)} + n)*\rangle \quad \text{(I.H. より)} \\
&= \langle \Lambda g.[M_1]_{v(x/f)(y/g)} + 1 + n.([M_1]_{v(x/f)(y/c_0)} + n)*\rangle \\
&= [\lambda y.M_1]_{v(x/f)} + n
\end{aligned}$$

定義 2.3.18 (最小の累積割り当て)
v が最小の累積割り当て *(minimal cumurative assignment,* v_{c_0} と表記する*)* であるとは、$\forall x(v(x) = c_0)$。

補題 2.3.19 (正規形)
M が正規形 $\Rightarrow [M]_{v_{c_0}} * = 0$

(証明) M の構成に関する帰納法。$M \equiv xM_1 \dots t_n$ $(M_1, \dots, M_n : normal\ form)$ の場合のみを示す。

$$\begin{aligned}
[xM_1 \dots t_n]_{v_{c_0}} * &= ([x]_{v_{c_0}}[M_1]_{v_{c_0}} \dots [M_n]_{v_{c_0}}) * \\
&= (c_0[M_1]_{v_{c_0}} \dots [M_n]_{v_{c_0}}) * \\
&= (c_{0+[M_1]_{v_{c_0}}*+\cdots+[M_n]_{v_{c_0}}*}) * \\
&= (0 + [M_1]_{v_{c_0}} * + \cdots + [M_n]_{v_{c_0}} *) \\
&= (0 + 0 + \cdots + 0) = 0
\end{aligned}$$

補題 2.3.20 (レデックス)
M が正規形でない $\Rightarrow \exists f \in C_\alpha([M]_{v_{c_0}} = f+1)$ *(すなわち、*$[M]_{v_{c_0}}* \neq 0$*)*

(証明) M の構成に関する帰納法。

1. $M \equiv M_1M_2$ で M_1M_2 それ自身はレデックスではなく、M_1 が正規形で M_2 にレデックスが存在するとき。
   $M_1 \equiv xs_1\ldots s_n$ $(s_1,\ldots,s_n$ は正規形) とおくと、
   $[M_1M_2]_{v_{c_0}} = [M_1]_{v_{c_0}}[M_2]_{v_{c_0}} = [xs_0\ldots s_n]_{v_{c_0}}(f'+1) = c_0(f'+1) = c_{0+(f'+1)*} = c_{f'*}+1$
   よって $f = c_{f'*}$ を取ればよい。

2. $M \equiv (\lambda x.M_1)$ $(x \in \mathrm{FV}(M_1))$ で M_1 にレデックスが存在するとき。
   $[(\lambda x.M_1)]_{v_{c_0}} = \langle \Lambda g.[M_1]_{v_{c_0}(x/g)}+1, [M_1]_{v_{c_0}}*\rangle = \langle \Lambda g.[M_1]_{v_{c_0}(x/g)}+1, (f'+1)*\rangle = \langle \Lambda g.[M_1]_{v_{c_0}(x/g)}, f'*\rangle + 1$
   $f = \langle \Lambda g.[M_1]_{v_{c_0}(x/g)}, f'*\rangle$ と取ると、補題 2.3.14 より $f \in C_\alpha$ である。

系 2.3.21
$[M]_{v_{c_0}}* = 0 \Rightarrow M$ は正規形

補題 2.3.22 (リダクション)
$M \rhd N \Rightarrow [M]_v \leq [N]_v + 1$

(証明) M の構成に関する帰納法。
$M \equiv (\lambda x.M_1)$ で $M_1 \rhd N_0$ and $N = (\lambda x.N_0)$ の場合、

$$\begin{aligned}
[(\lambda x.M_1)]_v &= \langle \Lambda f.[M_1]_{v(x/f)}+1, [M_1]_{v(x/c_0)}*\rangle \\
&\leq \langle \lambda f.[N_0]_{v(x/f)}+1+1, ([N_0]_{v(x/c_0)}+1)*\rangle \quad \text{(I.H. と定義 2.3.13 より)} \\
&= \langle \lambda f.[N_0]_{v(x/f)}+1, [N_0]_{v(x/c_0)}*\rangle + 1 \\
&= [N]_v + 1
\end{aligned}$$

系 2.3.23
$M \rhd N \Rightarrow [M]_{v_{c_0}}* \leq [N]_{v_{c_0}}*+1$

この系により、$h(M) = [M]_{v_{c_0}}*$ とすれば h は命題 2.1.8 の条件を満たすことが示された。次に 3 番目の条件を満たす関すう $()^*$ を導入する。

定義 2.3.24 (最左リダクション戦略) M が正規形でないとき、$(M)^*$ を以下のように帰納的に定義する。

1. $M \equiv (\lambda x.M_1)M_2$ のとき。 $M^* \equiv M_1[x := M_2]$

2. $M \equiv M_1M_2$ で、M_1M_2 それ自身がレデックスではないとき。

   (a) M_1 正規形でないとき。$M^* \equiv (M_1)^*M_2$

   (b) M_1 が正規形のとき。$M^* \equiv M_1(M_2)^*$

3. $M \equiv (\lambda x.M_1)$ のとき。$(M)^* \equiv (\lambda x.M_1^*)$

BCK–ラムダ項の場合、この最左リダクションが最短リダクション戦略になることを示す。

補題 2.3.25 (最左リダクション戦略) M が正規形ではないとする。

1. M が関数抽象の形をしていないとき。$[M]_{v_{c_0}} = [M^*]_{v_{c_0}} + 1$

2. M が関数抽象の形をしているとき。$[M]_{v_{c_0}}* = [M^*]_{v_{c_0}} * + 1$

(証明) M の構成に関する同時帰納法。

1. $M \equiv (\lambda x.M_1)M_2$ のとき。

$$\begin{aligned}[(\lambda x.M_1)M_2]_{v_{c_0}} &= \langle \Lambda f.[M_1]_{v_{c_0}(x/f)} + 1, [M_1]_{v_{c_0}(x/c_0)}\rangle [M_2]_{v_{c_0}} \\ &= [M_1]_{v_{c_0}(x/[M_2]_{v_{c_0}})} + 1 \\ &= [M_1]_{v_{c_0}}[x := [M_2]_{v_{c_0}}] + 1 \\ &= [M^*] + 1\end{aligned}$$

2. $M \equiv \lambda x.M_1$ のとき。

$$\begin{aligned}[\lambda x.M_1]_{v_{c_0}}* &= [M_1]_{v_{c_0}(x/c_0)} * \\ &= [M_1]_{v_{c_0}} * \\ &= [M_1^*]_{v_{c_0}} * + 1 (\text{I.H. より}) \\ &= [M_1]_{v_{c_0}}[x := [M_2]_{v_{c_0}}] + 1 \\ &= [M^*] + 1\end{aligned}$$

3. $M \equiv M_1M_2$ で M_1M_2 それ自身がレデックスではなく M_1 が正規形のとき。$M^* \equiv M_1^*M_2$ であり、M_1 は関数抽象の形をしてはいない。

$$\begin{aligned}[M_1M_2]_{v_{c_0}} &= [M_1]_{v_{c_0}}[M_2]_{v_{c_0}} \\ &= ([M_1^*]_{v_{c_0}} + 1)[M_2]_{v_{c_0}} (\text{I.H. より}) \\ &= [M_1^*]_{v_{c_0}}[M_2]_{v_{c_0}} + 1 (\text{補題 2.3.6 より}) \\ &= [M_1^*M_2]_{v_{c_0}} + 1 \\ &= [M^*]_{v_{c_0}} + 1\end{aligned}$$

4. $M \equiv M_1M_2$ で M_1M_2 それ自身がレデックスではなく M_1 が正規形のとき。$M^* \equiv M_1M_2^*$ であり、$M_1 \equiv xN_1 \ldots N_n$ ($N_1, \ldots, N_n$ は正規形) とおく。

   (a) M_2 が関数抽象の形をしていないとき、

$$\begin{aligned}[M_1M_2]_{v_{c_0}} &= [M_1]_{v_{c_0}}[M_2]_{v_{c_0}} \\ &= [xN_1 \ldots N_n]_{v_{c_0}}([M_2^*]_{v_{c_0}} + 1)(\text{I.H. より}) \\ &= c_0([M_2^*]_{v_{c_0}} + 1) \\ &= c_{([M_2^*]_{v_{c_0}}+1)*} \\ &= c_{[M_2^*]_{v_{c_0}}*} + 1 \\ &= [xN_1 \ldots N_n]_{v_{c_0}}[M_2^*]_{v_{c_0}} + 1 \\ &= [M_1]_{v_{c_0}}[M_2^*]_{v_{c_0}} + 1 \\ &= [M^*]_{v_{c_0}} + 1\end{aligned}$$

(b) M_2 が関数抽象の形をしているとき、

$$\begin{aligned}[M_1M_2]_{v_{c_0}} &= [M_1]_{v_{c_0}}[M_2]_{v_{c_0}} \\ &= [xN_1 \ldots N_n]_{v_{c_0}}([M_2]_{v_{c_0}}) \\ &= c_0([M_2]_{v_{c_0}}) \\ &= c_{[M_2]_{v_{c_0}}*} \\ &= c_{([M_2^*]_{v_{c_0}}*+1)}(\text{I.H. より}) \\ &= c_{[M_2^*]_{v_{c_0}}*}+1 \\ &= [xN_1 \ldots N_n]_{v_{c_0}}[M_2^*]_{v_{c_0}}+1 \\ &= [M_1]_{v_{c_0}}[M_2^*]_{v_{c_0}}+1 \\ &= [M^*]_{v_{c_0}}+1\end{aligned}$$

系 2.3.26 (最左リダクション戦略) t が正規形でないとき、
$[M]_{v_{c_0}}* = [M^*]_{v_{c_0}} *+1$

定理 2.3.27 *BCK–*ラムダ項では最左リダクション戦略が最短リダクション戦略である。

(証明) 補題 2.3.19 と系 2.3.21 より BCK–ラムダ項の命題 2.1.8 の条項 1 を満たす。また、補題 2.3.22 より解釈は命題 2.1.8 の条項 2 を満たすことが分かる。そして、補題 2.3.25 より最左リダクション戦略及び解釈は条項 3 を満たすから、BCK–ラムダ項において最左リダクション戦略は最短リダクション戦略となる。

2.4 シンプルなリダクション戦略

本節ではシンプルなリダクション戦略を定義し、ラムダ–I 項と BCK–ラムダ項のリダクション上の関係について探る。

$(\lambda x.M)N$ というレデックスがあったとき、束縛変数の有無といった M や N の形を考慮せず、単純にリダクション戦略を定めるとすると次の 3 通りの優先順位を付けることができる。

1. このレデックスをリダクションする。
2. M が正規形でないとき、M の中をリダクションする。
3. N が正規形でないとき、N の中をリダクションする。

この場合 $3 \times 2 \times 1 = 6$ 通りとなる。さらにレデックスではない場合は左からか右からかの 2 通りであるから合計 $6 \times 2 = 12$ 通りとなる。このリダクション戦略は次のように定式化される。

定義 2.4.1 (シンプルなリダクション戦略) レデックス $(\lambda x.M)N$ に対して、$\underline{(\lambda x.\overline{M}^{(2)})\overline{N}^{(3)}}_{(1)}$ と番号付けてこの (1)〜(3) のうちどこからリダクションを行うかを *3* つ組で表わす。例えば (2) → (3) → (1) という優先順位を付けた場合にこれを $(2, 3, 1)$ と表す。同様にレデックスではないラムダ項 MN に対して $\underline{M}_{(1)}\underline{N}_{(2)}$ と番号付けてどこからリダクションを行うかを *2* つ組で表す。この *3* つ組と *2* つ組を並べて *12* 通りのリダクション戦略を定義する。

例 2.4.2 最左リダクション戦略は $(1, 2, 3)(1, 2)$ あるいは $(1, 3, 2)(1, 2)$ となる。値呼び出しリダクション戦略は $(2, 1, 3)(1, 2)$ となる。

ラムダ項の展開において、これらのシンプルなリダクション戦略が最短・最長リダクション戦略となりうるだろうか。残念ながら次のように全ての場合で反例が見つかる。

	最短リダクション戦略	最長リダクション戦略
$(1, 2, 3)(1, 2)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(1, 2, 3)(2, 1)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(1, 3, 2)(1, 2)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(1, 3, 2)(2, 1)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(2, 1, 3)(1, 2)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(2, 1, 3)(2, 1)$	$(\lambda' a.aa)((\lambda' x.x)x)$	$(\lambda' a.b)((\lambda' x.x)x)$
$(2, 3, 1)(1, 2)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$
$(2, 3, 1)(2, 1)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$
$(3, 1, 2)(1, 2)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$
$(3, 1, 2)(2, 1)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$
$(3, 2, 1)(1, 2)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$
$(3, 2, 1)(2, 1)$	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$

ではさらに制限を加えるとどうなるであろうか。BCK–ラムダ項の条件とラムダ– I 項の条件を加えた場合を考える。

	BCK–ラムダ項		ラムダ–I 項	
	最短	最長	最短	最長
$(1,2,3)(1,2)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(1,2,3)(2,1)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(1,3,2)(1,2)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(1,3,2)(2,1)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(2,1,3)(1,2)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(2,1,3)(2,1)$	○	$(\lambda' a.b)((\lambda' x.x)x)$	$(\lambda' a.aa)((\lambda' x.x)x)$	○
$(2,3,1)(1,2)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$
$(2,3,1)(2,1)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$
$(3,1,2)(1,2)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$
$(3,1,2)(2,1)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$
$(3,2,1)(1,2)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$
$(3,2,1)(2,1)$	$(\lambda' a.b)((\lambda' x.x)x)$	○	○	$(\lambda' x.xx)((\lambda' x.x)x)$

この表で '○' が付いているのは最短か最長リダクション戦略になっていることを示している。またラムダ項は反例を示している。このように制限を加えると、全てのリダクション戦略で BCK またはラムダ– I 項が最短か最長リダクション戦略になっていることが分かる。

では一般の BCK–ラムダ項やラムダ–I 項ではどうなるであろうか。それは次の表のようになる。

	BCK–ラムダ項		ラムダ–I 項	
	最短	最長	最短	最長
$(1,2,3)(1,2)$	○	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	○
$(1,2,3)(2,1)$	$((\lambda x.x)(\lambda a.b))((\lambda x.x)x)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	$((\lambda x.x)(\lambda a.aa))((\lambda x.x)x)$
$(1,3,2)(1,2)$	○	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	○
$(1,3,2)(2,1)$	$((\lambda x.x)(\lambda a.b))((\lambda x.x)x)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	$((\lambda x.x)(\lambda a.aa))((\lambda x.x)x)$
$(2,1,3)(1,2)$	$(\lambda y.y((\lambda x.x)x))(\lambda a.b)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	$(\lambda y.y((\lambda x.x)x))(\lambda a.aa)$
$(2,1,3)(2,1)$	$(\lambda y.y((\lambda x.x)x))(\lambda a.b)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda a.aa)((\lambda x.x)x)$	$(\lambda y.y((\lambda x.x)x))(\lambda a.aa)$
$(2,3,1)(1,2)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$
$(2,3,1)(2,1)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$
$(3,1,2)(1,2)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$
$(3,1,2)(2,1)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$
$(3,2,1)(1,2)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$
$(3,2,1)(2,1)$	$(\lambda a.b)((\lambda x.x)x)$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.b)(yz))$	$(\lambda x.x(\lambda x.x))(\lambda y.(\lambda a.aa)(yz))$	$(\lambda a.aa)((\lambda x.x)x)$

$(1,2,3)(1,2)$ と $(1,3,2)(1,2)$ はともに最左リダクション戦略であるが、この場合を除き全ての場合で反例が見つかる。そして興味深いのは BCK–ラムダ項では $\lambda a.b$ というラムダ項を含んでいて、これを $\lambda a.aa$ に置き換えるとそれがラムダ-I 項の反例になるということである。型の付いた反例を見つけるとすれば $\lambda a.a(ab)$ を取ればいい。そして、BCK–ラムダ項の最短リダクション戦略にはラムダ–I 項の最長リダクション戦略が、BCK–ラムダ項の最長リダクション戦略にはラムダ–I 項の最短リダクション戦略が対応している。これは偶然であろうか。

このことを考えるために、まず BCI–ラムダ項を定義する。

定義 2.4.3 (BCI–ラムダ項)

1. x が変数 $\Rightarrow$ x は *BCI*–ラムダ項

2. M_1, M_2 が *BCI*–ラムダ項であり $\mathrm{FV}(M_1) \cap \mathrm{FV}(M_2) = \emptyset \Rightarrow (M_1 M_2)$ は *BCI*–ラムダ項

3. M が *BCI*–ラムダ項であり x が変数で x が M に自由出現する $\Rightarrow (\lambda x.M)$ は *BCI*–ラムダ項

このBCI–ラムダ項ではリダクションにより記号λがちょうど一つ減るため、どのようにリダクションを行っても同じステップ数で正規形に到る。すなわち、全てのリダクションが最短であり最長になる。

このBCI–ラムダ項の拡張を考えてみると、2. で条件「$\mathrm{FV}(M_1) \cap \mathrm{FV}(M_2) = \emptyset$」を省くか、3. で条件「$x$が$M$に自由出現する」を省くかのどちらかが自然であろう。つまり、BCI–ラムダ項を不動点としてBCK–ラムダ項とラムダ–I項にはリダクションに関して双対性があるのではないかと考えられる。

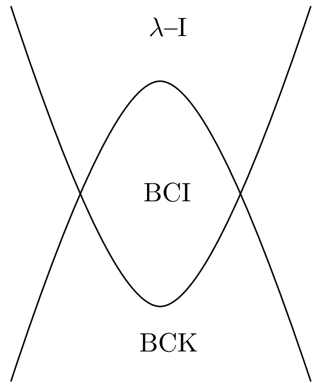


図 2.3: BCK と λ–I の関係

2.5 結論

以上の最短リダクション戦略に関する結果を図示すると次のようになる。

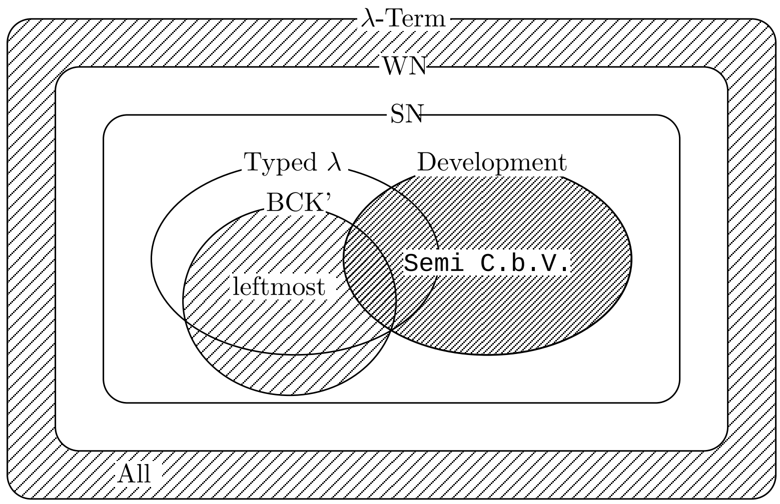


図 2.4: 最短リダクション戦略

濃い斜線で示されている部分が本論文で最短リダクション戦略が見つかったところである。BCK–ラムダ項では最左リダクション戦略が最短となり、ラムダ項の展開については偽値呼び出しリダクション戦略が最短となる。(図中では Semi C.b.V. と呼んでいる。) この部分をさらに広げることは今後の課題である。

最短リダクション戦略については [15] が知られている。この論文は並行変換を基にある種の最左リダクション戦略が最短となることを示した。並行変換は複数のリダクションを一度に行うリダクションであり、ラムダ項における最短リダクション戦略を示すものではないが、BCK–ラムダ項に制限を加えた場合には最左リダクション戦略が最短となることがこの論文から導かれる。しかしながら、この証明は permutation に基く複雑なものである。それに対して本論文ではラムダ項から自然数への関数を提示し、リダクションにより 1 減ることから最短であることを示すという見通しのよい証明である。

また、ラムダ項の展開ではより簡単な関数により、証明されることを示した。

この結果の主な内容は、the 32nd MLG meeting[20] で発表されている。

ラムダ項に関しては、高橋正子 [31], H.P. Barendregt[1], J.Roger Hindley and Jonathan P.Seldin[12] で詳しく論じられている。

第3章　証明反駁アルゴリズム(proof or refutation algorithm)

本章では、いくつかの命題様相論理に対する証明反駁アルゴリズムを考える。証明反駁アルゴリズムとは与えられた論理式が証明可能なときには証明図を、証明不可能なときにはその反例を与えるアルゴリズムである。

さらに、その証明反駁アルゴリズムを xpe(X window system Proof Editor) 上に実装したので、その結果についても述べる。([21],[22],[23] 及び、本論文の付録を参照)

3.1　はじめに

論理式がある論理体系で証明不可能なとき、その反例となるモデルを具体的に構成できるであろうか。反例を作ることが可能な体系もいくつか存在する。しかしながら、完全性により証明不可能なときに反例が存在することが保証されていたとしても、具体的に構成することが困難な体系も数多く存在する。実際に、有限モデル性により有限なモデルの存在が保証され、さらに決定可能性が示されているとしても、具体的に反例を構成する実行可能な方法が与えられているとは限らない。

多くの体系ではシュッテの手法により反例を構成できるが知られている。しかしながら、この方法は指数時間という莫大なコストがかかり、現実的ではない。

そこで考えられたのが、決定手続きから直接反例を抽出する方法である。そのような手続きが存在すれば、効率的に反例を作れることができる。

通常の定理自動証明システム (theorem prover) は、一般には与えられた証明可能か否かを判定するのみであり、証明図や反例が出力されるわけではない。論理を研究では、それだけでは不便である。実際に論理の研究を進める際に必要となるのは、証明可能な論理式ではどのように証明図が構成されるのか、そして証明不可能な論理式ではどのように反例が構成されるのかという分析である。

こうした、証明可能なときには証明図を、証明不可能なときには反例を出力するアルゴリズムを証明反駁アルゴリズム (proof or refutation algorithm) と呼ぶことにする。

ある体系に対してこの証明反駁アルゴリズムが存在するならば、一般に次の 4 つ性質が直ちに得られる。

- 決定可能性
- Cut 除去
- 完全性
- 有限モデル性

このように、論理学における重要な性質を同時に示すことができる有用な手法でもあるである。

この章の 3.2 節から 3.4 節では古典命題論理、様相論理 K 及び KT に対する我々の証明反駁アルゴリズムについて述べる。これらのアルゴリズムは xpe 上に既に実装されている。([21],[22],[23] 及び、本論文の付録を参照)

直観主義命題論理に対しては [5] 及び [11] でループなしの決定手続きが示された。そして、この体系を元に 1995 年に [27] によって、ループなしの証明反駁アルゴリズムが示された。これらの様相論理 K,KT 及び直観主義命題論理で

は、反例 (より正確に言えば反例を与えるクリプキフレーム) として木構造をしているものを取ることができる。そのためこれらに対する証明反駁アルゴリズムにおいては、証明の探索に失敗した際に得られる探索木 (反駁木) の構造そのままに反例を作ることができる。

それに対し S4 では有限の反例を作るためには (自明でない) クラスターが必要になる。ただし、クラスターとはフレームの到達可能関係が作る同値類のことである。したがって探索木の構造からさらにクラスターを導き、それを用いて反例を作り出す工夫が必要になる。3.5 節では、探索木内に二種類のブロックの概念とそれらのブロックを結ぶ二種類の関係を導入することにより得られる効率的な証明反駁アルゴリズムについて述べる。ループチェックを効率的に行うために、そのアルゴリズムにおいては新に補助的な記号 ■ と、履歴という概念を用いる工夫がなされている。このアルゴリズムは RPC'01([24]) で発表され、また既に xpe 上に実装されている。様相論理及びキリプキのセマンティクスについては、[26] を参照して欲しい。

3.2 古典命題論理

本節では次節以降の議論の基礎となる古典命題論理に対する証明反駁アルゴリズムの説明を行う。

古典命題論理の形式化には 3 通りある。1. シーケント計算 2. 自然演繹 3. ヒルベルト流の形式化である。このうち、決定手続きでは一般にシーケント計算を元にした体系が用いられる。古典命題論理をシーケント計算により形式化した体系は LK と呼ばれ、図 3.1 のようになる。

$$p \to p$$

$$\frac{\Gamma_1 \to \Delta_1, A \quad \Gamma_2, B \to \Delta_2}{\Gamma_1, \Gamma_2, A \supset B \to \Delta_1, \Delta_2} (\supset L) \qquad \frac{\Gamma, A \to \Delta, B}{\Gamma \to \Delta, A \supset B} (\supset R)$$

$$\frac{\Gamma, A \to \Delta}{\Gamma, A \wedge B \to \Delta} (\wedge L) \qquad \frac{\Gamma, B \to \Delta}{\Gamma, A \wedge B \to \Delta} (\wedge L)$$

$$\frac{\Gamma \to \Delta, A \quad \Gamma \to \Delta, B}{\Gamma \to \Delta, A \wedge B} (\wedge R) \qquad \frac{\Gamma, A \to \Delta \quad \Gamma, B \to \Delta}{\Gamma, A \vee B \to \Delta} (\vee L)$$

$$\frac{\Gamma \to \Delta, A}{\Gamma \to \Delta, A \vee B} (\vee R) \qquad \frac{\Gamma \to \Delta, B}{\Gamma \to \Delta, A \vee B} (\vee R)$$

$$\frac{\Gamma, A \to \Delta}{\Gamma \to \Delta, \neg A} (\neg R) \qquad \frac{\Gamma \to A, \Delta}{\Gamma, \neg A \to \Delta} (\neg L)$$

$$\frac{\Gamma \to \Delta}{A, \Gamma \to \Delta} (Weakening) \qquad \frac{\Gamma \to \Delta}{\Gamma \to \Delta, A} (Weakening)$$

$$\frac{A, A, \Gamma \to \Delta}{A, \Gamma \to \Delta} (Contraction) \qquad \frac{\Gamma \to \Delta, A, A}{\Gamma \to \Delta, A} (Contraction)$$

$$\frac{\Gamma_1, B, A, \Gamma_2 \to \Delta}{\Gamma_1, A, B, \Gamma_2 \to \Delta} (Exchange) \qquad \frac{\Gamma \to \Delta_1, B, A, \Delta_2}{\Gamma \to \Delta_1, A, B, \Delta_2} (Exchange)$$

$$\frac{\Gamma_1 \to \Delta_1, A \quad A, \Gamma_2 \to \Delta_2}{\Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2} (Cut)$$

図 3.1: 古典命題論理 LK

LK で与えられた論理式が証明可能か否かを判断しようとすると、論理式にこれらの推論規則のうち適用可能なものを適用していき、すべてのシーケントが始式になるかどうかで判断することになる。論理結合子に関する推論規則の場合には、規則を適用するとシーケントに出現する論理結合子の数が減るため、これの規則を適用しても無限の長さの証明図は生じない。

ところが、残りの構造規則には問題がある。

Cut $\cdots$ 上式の論理式 A(cut 論理式) を任意に取れるため、無限の長さの証明図を派生する可能性がある。

Contraction $\cdots$ 上式が下式よりも長くなっているため、繰り返し Contraction を適用すると無限の長さの証明図を派生してしまう。

Exchange $\cdots$ 上式の B, A にさらに Exchange 規則を適用すると A, B という並びになり、簡単にループを生じる。

Weakening $\cdots$ 消去する A に証明に必要なものを取ってしまうと証明図が完成しなくなり、証明を再検索する必要がある。

Cut の問題は、LK では Cut 除去定理により Cut の無い体系も元の LK と強さが同じであることが知られているので、Cut の無い体系を考えればよい。Exchange に関しては、シーケントの両辺を多重集合 (multi–set) と考えることで、論理式の並ぶ順序を無視することができる。Contraction は次の $(\supset L)$ のように必要な Contraction を含んだ形に推論規則を書き替えることで乗り越えることができる。そして、Weakening は次の始式の形のように最後に適用することで、余分な Weakening の適用を無くすことができる。

$$\Gamma, p \to p, \Delta$$

$$\frac{\Gamma \to \Delta, A \quad \Gamma, B \to \Delta}{\Gamma, A \supset B \to \Delta}\ (\supset L) \qquad \frac{\Gamma, A \to \Delta, B}{\Gamma \to \Delta, A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \to \Delta}{\Gamma, A \wedge B \to \Delta}\ (\wedge L) \qquad \frac{\Gamma \to \Delta, A \quad \Gamma \to \Delta, B}{\Gamma \to \Delta, A \wedge B}\ (\wedge R)$$

$$\frac{\Gamma, A \to \Delta \quad \Gamma, B \to \Delta}{\Gamma, A \vee B \to \Delta}\ (\vee L) \qquad \frac{\Gamma \to \Delta, A, B}{\Gamma \to \Delta, A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \to \Delta}{\Gamma \to \Delta, \neg A}\ (\neg R) \qquad \frac{\Gamma \to A, \Delta}{\Gamma, \neg A \to \Delta}\ (\neg L)$$

図 3.2: 古典命題論理の証明反駁アルゴリズムの体系 (Wang の体系 [38])

この手続きでは、上式は下式よりも論理結合子が減っているので証明探索の手続きは必ず停止する。この体系によって得られる木を**探索木**と呼ぶ。探索木の全ての枝が $\Gamma, p \to p, \Delta$ の形に到るとき、この探索木を**証明木**とよぶ。それとは逆に、枝のうち一つでも共通の命題変数を持たない $p_1, \cdots, p_n \to q_1, \cdots, q_m$ という形に到るならば、この探索木を**反駁木**と呼ぶ。この体系での証明木が見つかった場合には、その証明木からたやすく LK の証明図を作り出すことができる。すなわち、

定理 3.2.1 古典命題論理の証明反駁アルゴリズムの体系において、与えられた論理式に対する証明木が存在するならば、その論理式は LK で証明可能である。

また、証明木から簡単に LK の証明図を起こせることから、証明木を (古典命題論理の) 証明図と考えても差し支えない。

そして逆を示すために次の二つの補題を用いる。

補題 3.2.2 探索木において、始式の両辺に共通の命題変数が存在しないときには、その始式を偽にするモデルが存在する。

補題 3.2.3 探索木に現れる各推論規則おいて、上式のうち少なくとも一つに対して反例が存在するとき、その反例は下式の反例にもなっている。

そして次の二つの定理により、定理 3.2.1 の逆が示される。

定理 3.2.4 ある論理式に対する反駁木が存在するならば、反例が存在する。

定理 3.2.5 LK で証明可能な論理式はトーロジーである。

つまり、先に示した古典論理の証明反駁アルゴリズムでは、証明可能なときにはその証拠として証明図が得られ、証明不可能なときにはその証拠として反例が得られる。このような、証明反駁アルゴリズムを様相論理でも見い出していく。

3.3 様相論理 K

様相論理 K は次の左側の □ 規則を古典命題論理の LK に加えることで形式化される。K に対する証明反駁アルゴリズムの体系は右側の □ 規則のを Wang の体系に加えることで得られる。この推論規則は Weakening 規則を含んだ形になっている。

$$\frac{\Gamma \to A}{\Box\Gamma \to \Box A}\ (\Box) \qquad \frac{\Gamma \to A}{\Delta, \Box\Gamma \to \Box A, \Sigma}\ (\Box)$$

図 3.3: LK と Wang の体系に対する □ 規則

与えられた論理式に対するこの体系での証明木が存在するならば、様相論理 K の体系でも証明可能であることはほぼ自明であろう。

ここでモデルについて定義しておく。様相論理 K に対するモデルは次のようなクリプキモデルである。

定義 3.3.1 (クリプキモデル) クリプキモデルは *3* 個組 $(M, R, \models)$ である。但し、

$M \cdots$ 可能世界 *(*空でない集合*)*

$R \cdots$ 到達可能関係 *(*M 上の二項関係*)*

$\models \cdots$ 付値

証明不可能なとき反例を作るようにするためには、図 3.4 のように完全に論理結合子を分解してアトミックな形になってから適用する必要がある。

$$\Gamma, p \to p, \Delta$$

$$\frac{\Gamma \to \Delta, A \quad \Gamma, B \to \Delta}{\Gamma, A \supset B \to \Delta}\ (\supset L) \qquad \frac{\Gamma, A \to \Delta, B}{\Gamma \to \Delta, A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \to \Delta}{\Gamma, A \wedge B \to \Delta}\ (\wedge L) \qquad \frac{\Gamma \to \Delta, A \quad \Gamma \to \Delta, B}{\Gamma \to \Delta, A \wedge B}\ (\wedge R)$$

$$\frac{\Gamma, A \to \Delta \quad \Gamma, B \to \Delta}{\Gamma, A \vee B \to \Delta}\ (\vee L) \qquad \frac{\Gamma \to \Delta, A, B}{\Gamma \to \Delta, A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \to \Delta}{\Gamma \to \Delta, \neg A}\ (\neg R) \qquad \frac{\Gamma \to A, \Delta}{\Gamma, \neg A \to \Delta}\ (\neg L)$$

$$\genfrac{}{}{2pt}{0}{\Gamma \to A_1 \quad \cdots \quad \Gamma \to A_m}{p_1, \cdots, p_n, \Box\Gamma \to \Box A_1, \cdots, \Box A_m, q_1, \cdots q_l}\ (\Box)$$

図 3.4: 様相論理 K の証明反駁アルゴリズム

図 3.4 で □ 規則が 2 重線 (=) になっているのは、□ 規則が OR–分岐であるからである。すなわち $(\supset L)$,$(\wedge R)$,$(\vee L)$ 規則においては二つの上式がともに証明可能ならば下式も証明可能であるが、これに対し □ 規則では上式 $\Gamma \to A_1$ から $\Gamma \to A_n$ までのうち一つでも証明可能ならば下式が証明可能と考える。証明の探索木を考えるときには、OR–分岐を導入しておく方が便利である。しかも反例を作り出すためにはこのような規則の導入が必要である。それに対し、証明不可能なときに反例を出力しないような決定手続きでは、OR–分岐はバックトラックとして処理される。

そして、証明不可能なときには、上式の n 個の情報を全てを用いて反例を構成する。この作り方を例を用いて説明する。下図のように $\Box C \vee \Box B \wedge D \to \Box(\neg A \vee \Box C) \vee \Box B$ の反駁木を考える。

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A, A \to}{A \to \neg A}\,(\neg R) \qquad A \to B}{\Box A, C \to \Box\neg A, \Box B}\,(\Box) \qquad \Box B, C \to \Box\neg A, \Box B}{\Box A \vee \Box B, C \to \Box\neg A, \Box B}\,(\vee L)}{\Box A \vee \Box B, C \to \Box\neg A \vee \Box B}\,(\vee R)}{\Box A \vee \Box B \wedge C \to \Box\neg A \vee \Box B}\,(\wedge L)
\qquad
\begin{array}{cc} \{A, A\} & \{A\} \\ \vdots & \\ \{C\} & \end{array}
$$

図 3.5: K の反駁木と反例

これは xpe の出力であり、証明図の右側のものが反例を示している。これは $\{C\}, \{A, A\}, \{A\}$ の 3 点からなるクリプキモデルであり、$\{C\}$ から $\{A, A\}$ に、$\{C\}$ から $\{A\}$ にそれぞれ関係が付いている。つまり、

$$
\begin{array}{ccc} \{A, A\} & & \{A\} \\ & \nwarrow \quad \nearrow & \\ & \{C\} & \end{array}
$$

とうクリプキモデルを表現している。

次に手続きについて説明する。まず論理結合子 $\wedge, \vee, \wedge, \supset$ を分解し、シーケントが始式になっているかどうかを検討する。それから $\Box$ 規則を適用する、という手続きを繰り返す。

そして、もうこれ以上規則を適用できなくなり、かつ証明できないときには最も上のシーケントは $p_1, \cdots, p_n, \Box\Gamma \to q_1, \cdots, q_n$ という形をしている。対応するクリプキモデルとして反射律の成り立たない一点からなるフレームを取り、そこでの付値を $\{p_1, \cdots, p_n\}$ と定めると、この可能世界で元のシーケントは偽となる。

$\Box$ 規則では、上式の $\Gamma \to A_1, \cdots \Gamma \to A_n$ に対応するクキプケモデルの付値を $\{\Gamma, \neg A_1\}, \cdots, \{\Gamma, \neg A_n\}$ とする。このとき、下式の $p_1, \cdots, p_n, \Box\Gamma \to \Box A_1, \cdots, \Box A_m, q_1, \cdots, q_l$ 対応するクリプキモデルとして、

$$
\begin{array}{ccc} \{\Gamma, \neg A_1\} & \cdots & \{\Gamma, \neg A_m\} \\ & \nwarrow \quad \nearrow & \\ & \{p_1, \cdots, p_n\} & \end{array}
$$

を取ればこの下式はこのモデルの $\{p_1, \cdots, p_n\}$ で偽となる。

このようにフレームと付値を定めれば、証明不可能な論理式はそれを偽にするクリプケモデルが構成できることが分かる。つまり、証明反駁アルゴリズムで反駁木が得られたならば、その反駁木の═══と═══で挟まれた部分が一つの可能世界に対応すると考えることにより反例が作れるのである。先の例では次のような対応関係が取れる。

$$
\dfrac{A, A \to}{A \to \neg A}\,(\neg R) \qquad \cdots \{A, A\}
$$

$$
A \to B \qquad \cdots \{A\}
$$

$$
\dfrac{\dfrac{\dfrac{\Box A, C \to \Box\neg A, \Box B \qquad \Box B, C \to \Box\neg A, \Box B}{\Box A \vee \Box B, C \to \Box\neg A, \Box B}\,(\vee L)}{\Box A \vee \Box B, C \to \Box\neg A \vee \Box B}\,(\vee R)}{\Box A \vee \Box B \wedge C \to \Box\neg A \vee \Box B}\,(\wedge L) \qquad \cdots \{C\}
$$

図 3.6: 反駁木と付値との対応

この例において $(\wedge L)$ の右上式のシーケント $\Box B, D \to \Box\neg A, \Box B$ は証明可能なので、$\Box A, C \to \Box\neg A, \Box B$ から $\{C\}$ という付値を定める。

3.4 様相論理 KT

様相論理 KT は次の図 3.7 の一番左の形の T 規則を様相論理 K に加えた体系として形式化される。証明反駁アルゴリズムでは真中のように Contraction を加えた形を用いる。しかしこのままではループを生じてしまう。そこで証明探索を効率的に行うために、T 規則を適用するごとに対象となる $\Box$ 論理式に印を付けて、二度と同じ論理式には T 規則を適用しないようにすればよい。具体的には、一番右の形の規則ように $\Box A$ に T 規則を適用して上式を作ると $\Box A$ は $\blacksquare A$ に変わり、この $\blacksquare A$ には T 規則を適用しないようにするのである。

$$\frac{A,\Gamma \to \Delta}{\Box A,\Gamma \to \Delta}\ (T) \qquad \frac{\Box A, A,\Gamma \to \Delta}{\Box A,\Gamma \to \Delta}\ (T) \qquad \frac{\blacksquare A, A,\Gamma \to \Delta}{\Box A,\Gamma \to \Delta}\ (T)$$

図 3.7: 様相論理 KT の T 規則

このことに応じて、K で用いた $\Box$ 規則は次のように $\blacksquare$ に対して適用することになる。

$$\frac{\Gamma \to A_1 \quad \cdots \quad \Gamma \to A_m}{p_1,\cdots,p_n,\blacksquare\Gamma \to \Box A_1,\cdots,\Box A_m,q_1,\cdots q_l}\ (\Box)$$

図 3.8: 様相論理 KT の $\Box$ 規則

様相論理 KT に対するモデルは次のようなクリプキモデルである。

定義 3.4.1 (KT モデル) $(M,R,\models)$ がクリプキモデルであり、R が反射律を満たすとき、$(M,R,\models)$ は KT モデルである。

証明不可能なシーケントの反駁木から反例を構成する方法は、様相論理 K の場合とほぼ同じである。ただし、T があるために到達可能関係反射的になることに注意する必要がある。

完全に分解が完了したとき、最も上のシーケントは $p_1,\cdots,p_n,\blacksquare\Gamma \to q_1,\cdots,q_n$ という形をしている。これに対して、$\{p_1,\cdots,p_n\}$ を真にするクリプキモデルを対応させると、このシーケントは偽になる。ここで $\blacksquare\Gamma$ の解釈が問題になるが、これは自分自身以外のそこから辿り着ける可能世界で真と解釈すればよい。形式的には、

$w \models \blacksquare A \Leftrightarrow \forall v(wRv)(v \neq w \Rightarrow v \models A)$

である。そうすれば、T 規則では、

$$\begin{aligned} w \models \blacksquare A \text{ かつ } w \models A &\Leftrightarrow \forall v(wRv)(v \neq w \Rightarrow v \models A) \text{ かつ } w \models A \\ &\Leftrightarrow \forall v(wRv)(v \neq w \Rightarrow v \models A) \text{ かつ } (v = w \Rightarrow v \models A) \\ &\Leftrightarrow \forall v(wRv)(v \models A) \\ &\Leftrightarrow w \models \Box A \end{aligned}$$

となり、下式も偽になることがわかる。

$\Box$ 規則でも様相論理 K の場合と同様に、対応するモデルとして、

$$\begin{array}{ccc} \{\Gamma, \overset{\curvearrowright}{\neg A_1}\} & \cdots & \{\Gamma, \overset{\curvearrowright}{\neg A_m}\} \\ \nwarrow & \curvearrowright & \nearrow \\ & \{p_1,\cdots,p_n\} & \end{array}$$

をとれば、下式も偽になる。

次に例として $\Box\Box B, C \to \Box(\neg A \vee \Box C)$ を挙げる。xpe は次のような反駁木と反例を出力する。

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \to C}{\blacksquare B, B, B, A \to \Box C}\,(\Box)}{\Box B, B, A \to \Box C}\,(T)}{\Box B, B \to \neg A, \Box C}\,(\neg R)}{\Box B, B \to \neg A \lor \Box C}\,(\lor R)}{\blacksquare \Box B, \blacksquare B, B, C \to \Box(\neg A \lor \Box C)}\,(\Box)}{\blacksquare \Box B, \Box B, C \to \Box(\neg A \lor \Box C)}\,(T)}{\Box \Box B, C \to \Box(\neg A \lor \Box C)}\,(T)
\qquad\qquad
\begin{array}{c}
\{B\} \\ \vdots \\ \{B, B, A\} \\ \vdots \\ \{B, C\}
\end{array}
$$

図 3.9: 様相論理 KT の反駁木と反例

そうすると様相論理 K の場合と同様に次のような関係が取れる。

$$
B \to C \quad \cdots \{B\}
$$

$$
\dfrac{\dfrac{\dfrac{\blacksquare B, B, B, A \to \Box C}{\Box B, B, A \to \Box C}\,(T)}{\Box B, B \to \neg A, \Box C}\,(\neg R)}{\Box B, B \to \neg A \lor \Box C}\,(\lor R) \quad \cdots \{B, B, A\}
$$

$$
\dfrac{\dfrac{\blacksquare \Box B, \blacksquare B, B, C \to \Box(\neg A \lor \Box C)}{\blacksquare \Box B, \Box B, C \to \Box(\neg A \lor \Box C)}\,(T)}{\Box \Box B, C \to \Box(\neg A \lor \Box C)}\,(T) \quad \cdots \{C, B\}
$$

図 3.10: 反駁木と付値との対応

3.5 様相論理 S4

3.5.1 証明反駁アルゴリズム

様相論理 S4 の形式化は様相論理 KT の $\Box$ 規則を次の左側の $\Box$ 規則で置き換えることによって得られる。決定手続きでは右側の $\Box$ 規則で置き換で置き換えればよい。

$$\frac{\Box\Gamma \to A}{\Box\Gamma \to \Box A}\ (\Box) \qquad \frac{\Box\Gamma \to A_1 \quad \cdots \quad \Box\Gamma \to A_m}{p_1, \cdots, p_n, \blacksquare\Gamma \to \Box A_1, \cdots, \Box A_m, q_1, \cdots q_l}\ (\Box)$$

図 3.11: 様相論理 S4 の $\Box$ 規則

ところが、この S4 の体系では証明探索でループ生ずることがある。例えば、次のような場合である。

$$\cfrac{\cfrac{\cfrac{\vdots}{\Box\neg\Box A \to A}\ (\Box)}{\blacksquare\neg\Box A \to \Box A, A}\ (\neg R)}{\cfrac{\blacksquare\neg\Box A, \neg\Box A \to A}{\Box\neg\Box A \to A}\ (T)}$$

そのため、$\Box$ 規則を適用する度に探索木に同じシーケントが出現しているかをチェックする必要がある。それでは効率が悪いため、Heuerding ら [9] によって、履歴 (history) を用いる体系が提出された。元の論文 [9] ではこのアイディアはタブローで定義された体系に対して用いられているが、これをシーケント計算の体系に適用し、さらに証明不可能な論理式に対して反駁木を出力するようにしている。その体系の規則は図 3.12 である。この体系では効率的にループをチェックすることができる。

この様相論理 S4 に対するモデルは次のようなクリプキモデルである。

定義 3.5.1 (S4 モデル) $(M, R, \models)$ がクリプキモデルであり、R が反射律と推移律を満たすとき、$(M, R, \models)$ は *S4* モデルである。

S4 の証明反駁アルゴリズムでも $\blacksquare$ が出現するが、この解釈は KT の場合とは異なっている。S4 の場合には単純に無視する論理式、あるいは $\top$ と考える。

次に履歴について説明する。一般の体系のシーケントは $\Gamma \to \Delta$ のように二つの論理式の列を $\to$ で結んだものであるが、Heuerding らが提出した体系では、

$$\Gamma \to \Delta\langle\Box\Pi|\Box\Sigma\rangle$$

というように、さらに $\Box$ 論理式の集合のペア $\langle\Box\Pi|\Box\Sigma\rangle$ を通常のシーケントに付けたものをシーケントと考えるのである。この $\Box$ 論理式の集合のペアを履歴と呼ぶ。左の $\Box\Pi$ が妥当な履歴 (valid history) であり、右の $\Box\Sigma$ が非妥当な履歴 (invalid hisotry) である。本論文で S4 の反例はクラスターをノードとするような木構造を持っているものとして与えられるが、妥当な履歴は対応クラスターに属するすべての可能世界で妥当となり、非妥当な履歴は対応するクラスターに属するすべての可能世界で妥当ではない。

$$\dfrac{}{\blacksquare\Gamma, p_1, \cdots, p_n \to q_1, \cdots, q_m, \Box\Delta\ \langle\Box\Gamma|\Box\Sigma\rangle}\ (init)_s\ (\Box\Delta \subset \Box\Sigma\text{のとき})$$

$$\dfrac{}{\blacksquare\Gamma, p_1, \cdots, p_n \to q_1, \cdots, q_m\ \langle\Box\Gamma|\epsilon\rangle}\ (init)_t$$

$$\dfrac{\Gamma \to \Delta, A\ \langle\Box\Pi|\Box\Sigma\rangle \quad \Gamma, B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \supset B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\supset L) \qquad \dfrac{\Gamma, A \to \Delta, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \to \Delta, A \supset B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\supset R)$$

$$\dfrac{\Gamma, A, B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \wedge B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\wedge L) \qquad \dfrac{\Gamma \to \Delta, A\ \langle\Box\Pi|\Box\Sigma\rangle \quad \Gamma \to \Delta, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \to \Delta, A \wedge B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\wedge R)$$

$$\dfrac{\Gamma, A \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle \quad \Gamma, B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \vee B \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\vee L) \qquad \dfrac{\Gamma \to \Delta, A, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \to \Delta, A \vee B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\vee R)$$

$$\dfrac{\Gamma \to A, \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma\neg A \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\neg L) \qquad \dfrac{\Gamma, A \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \to \Delta, \neg A\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\neg R)$$

$$\dfrac{\Box\Gamma \to A_1\ \langle\Box\Gamma|\Box\Sigma, \Box\Theta\rangle \quad \cdots \quad \Box\Gamma \to A_m\ \langle\Box\Gamma|\Box\Sigma, \Box\Theta\rangle}{\blacksquare\Gamma, p_1, \cdots, p_n \to \Box A_1, \cdots, \Box A_m, \Box\Delta, q_1, \cdots q_l\ \langle\Box\Gamma|\Box\Sigma\rangle}\ (\Box)_s$$
$$(\Box\Theta \equiv \Box A_1, \cdots \Box A_n,\ \Box\Theta \cap \Box\Sigma = \emptyset,\ \Box\Delta \subset \Box\Sigma)$$

$$\dfrac{\Box\Gamma \to A_1\ \langle\Box\Gamma|\Box\Theta\rangle \quad \cdots \quad \Box\Gamma \to A_m\ \langle\Box\Gamma|\Box\Theta\rangle}{\blacksquare\Gamma, p_1, \cdots, p_n \to \Box A_1, \cdots, \Box A_m, q_1, \cdots q_l\ \langle\Box\Gamma|\epsilon\rangle}\ (\Box)_t$$
$$(\Box\Theta \equiv \Box A_1, \cdots, \Box A_n)$$

$$\dfrac{\blacksquare A, A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Box A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (T)\ (\Box A \in \Box\Pi\text{のとき}) \qquad \dfrac{\blacksquare A, A, \Gamma \to \Delta\ \langle\Box A, \Box\Pi|\epsilon\rangle}{\Box A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (T)_t\ (\Box A \notin \Box\Pi\text{のとき})$$

図 3.12: 様相論理 S4 の証明反駁アルゴリズムの体系

この体系について補足説明する。まず、始式 $init$, $\Box$, T 規則がそれぞれ二つあることを注意しておこう。二つの $\Box$ 規則以外では、シーケントの両辺は多重集合として扱う。$\Box$ 規則を適用いて上式を求めるときには、同じ論理式を取り除く。例えば、

$$\dfrac{\Box A, \Box B \to C\langle\Box A, \Box B|\Box C, \Box D\rangle \quad \Box A, \Box B \to D\langle\Box A, \Box B|\Box C, \Box D\rangle}{\blacksquare A, \blacksquare B, \blacksquare A, p \to \Box C, \Box C, \Box D, q\langle\Box A, \Box B|\epsilon\rangle}\ (\Box)$$

というようにである。

そして、様相論理 K や KT と同様に $\Box$ 規則は OR–分岐である。すなわち、上式のうち一つでも証明可能ならば、下式が証明可能と考える。普通の AND–分岐の規則と区別するため、$\Box$ 規則は 2 重線 (=) で表わす。

[10] で示された体系は、我々の体系から $(\Box)_t$ 規則と $(init)_t$ 規則を取り除くことにより得られる。それは反例を出力することを要求しないない決定手続きとして眺めた場合には、我々の体系と本質的に同じである。つまり、次の定理が直ちに成り立つ。

定理 3.5.2 図 *3.12* で示されている体系は様相論理 *S4* に対する決定手続きを与える。

(証明) [10] 参照。

3.5.2 反駁木からの反例の構成

本節では、反例の構成法を示す。この方法で作られたモデルが実際に反例になっていることは次節で示すことにする。クリプキモデルを構成するために、次の図 3.13 で示されている反駁木を考える。

$$\dfrac{}{\blacksquare\Gamma, p_1, \cdots, p_n \not\to q_1, \cdots, q_m, \Box\Delta\ \langle\Box\Gamma|\Box\Sigma\rangle}\ (init)_s (\Box\Delta \subset \Box\Sigma\text{のとき})$$

$$\dfrac{}{\blacksquare\Gamma, p_1, \cdots, p_n \not\to q_1, \cdots, q_m\ \langle\Box\Gamma|\epsilon\rangle}\ (init)_t$$

$$\dfrac{\Gamma \not\to \Delta, A\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \supset B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\supset L)_1 \quad \dfrac{\Gamma, B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \supset B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\supset L)_2 \quad \dfrac{\Gamma, A \not\to \Delta, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \not\to \Delta, A \supset B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\supset R)$$

$$\dfrac{\Gamma, A, B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \wedge B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\wedge L) \quad \dfrac{\Gamma \not\to \Delta, A\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \not\to \Delta, A \wedge B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\wedge R)_1 \quad \dfrac{\Gamma \not\to \Delta, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \not\to \Delta, A \wedge B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\wedge R)_2$$

$$\dfrac{\Gamma, A \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \vee B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\vee L)_1 \quad \dfrac{\Gamma, B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, A \vee B \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\vee L)_2 \quad \dfrac{\Gamma \not\to \Delta, A, B\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \not\to \Delta, A \vee B\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\vee R)$$

$$\dfrac{\Gamma \not\to A, \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma, \neg A \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\neg L) \quad \dfrac{\Gamma, A \not\to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Gamma \not\to \Delta, \neg A\ \langle\Box\Pi|\Box\Sigma\rangle}\ (\neg R)$$

$$\dfrac{\Box\Gamma \not\to A_1\ \langle\Box\Gamma|\Box\Sigma, \Box\Theta\rangle \quad \cdots \quad \Box\Gamma \not\to A_m\ \langle\Box\Gamma|\Box\Sigma, \Box\Theta\rangle}{\blacksquare\Gamma, p_1, \cdots, p_n \not\to \Box A_1, \cdots, \Box A_m, \Box\Delta, q_1, \cdots q_l\ \langle\Box\Gamma|\Box\Sigma\rangle}\ (\Box)_s$$

$$(\Box\Theta \equiv \Box A_1, \cdots \Box A_n,\ \Box\Theta \cap \Box\Sigma = \emptyset,\ \Box\Delta \subset \Sigma)$$

$$\dfrac{\Box\Gamma \not\to A_1\ \langle\Box\Gamma|\Box\Theta\rangle \quad \cdots \quad \Box\Gamma \not\to A_m\ \langle\Box\Gamma|\Box\Theta\rangle}{\blacksquare\Gamma, p_1, \cdots, p_n \not\to \Box A_1, \cdots, \Box A_m, q_1, \cdots q_l\ \langle\Box\Gamma|\epsilon\rangle}\ (\Box)_t$$

$$(\Box\Theta \equiv \Box A_1, \cdots, \Box A_n)$$

$$\dfrac{\blacksquare A, A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}{\Box A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (T)\, (\Box A \in \Box\Pi\text{のとき}) \quad \dfrac{\blacksquare A, A, \Gamma \to \Delta\ \langle\Box A, \Box\Pi|\epsilon\rangle}{\Box A, \Gamma \to \Delta\ \langle\Box\Pi|\Box\Sigma\rangle}\ (T)_t\, (\Box A \notin \Box\Pi\text{のとき})$$

図 3.13: S4 の反駁木の導出規則

証明反駁アルゴリズムの体系と反駁木の違いは一つである。それは、AND–分岐での上式が片方になることである。反駁木ではこのように片方の上式を取り外してもよい。なぜなら、AND–分岐では二つの上式のうちどちらか一つが証明不可能ならば、下式が証明不可能であるからである。反例を作るときには、その証明不可能なシーケントのみを用いる。それは $\supset$, $\wedge$, $\vee$ 規則での分岐がなくなることであり、その結果、シーケントの適用規則の入れ替えをし、一定の順序に並べ替えることができるようになる。

まず、ブロックの概念を定義し、それから反例の構成法を定義する。

定義 3.5.3 (ブロック) ブロックとは、反駁木のうち始式か $\Box$ 規則で始まり $\Box$ 規則か終式で終わって、その間には $\Box$ 規則を持たない部分である。すなわち、ブロックとは反駁木の部分木のうち、*2* 重線 *(=)* から隣合う *2* 重線で囲まれた部分である。

1. **対称的ブロック** $(\Box)_s$ 規則か $(init)_s$ の始式で始まるブロック。

2. **推移的ブロック** $(\Box)_t$ 規則か $(init)_t$ の始式で始まるブロック。

また、始式で始まるブロックを**イニシャルブロック** という。イニシャルブロックは次の二つに分けられる。

1. **対称的イニシャルブロック** $(init)_s$ の始式で始まるブロック。

2. **推移的イニシャルブロック** $(init)_t$ の始式で始まるブロック。

定義 3.5.4 (反例) 反駁木に対応するクリプキモデルと可能世界を木の構成に関して帰納的に以下のように定義する。

1. $\blacksquare\Gamma, p_1, \cdots, p_n \not\rightarrow q_1, \cdots, q_m, \Box\Delta\langle\Box\Gamma|\Box\Sigma\rangle$ で始まる対称的イニシャルブロックのとき。対応する (M, w) として、$M = (W, R, \models)$, $W = \{w\}, R = (w, w)$, $w \models p_1 \wedge \cdots \wedge p_n$ を取る。

2. 推移的イニシャルブロックのとき。対称的イニシャルブロックと同じ。

3. シンメトリックブロックで、

$$\frac{\cdots \quad \Box\Gamma \not\rightarrow S_i\ \langle\Box\Gamma|\Box\Sigma,\Box\Theta\rangle \quad \cdots \quad \Box\Gamma \not\rightarrow T_j\ \langle\Box\Gamma|\Box\Sigma,\Box\Theta\rangle \quad \cdots}{\blacksquare\Gamma, p_1, \cdots, p_n \not\rightarrow \Box S_1, \cdots, \Box S_m, \Box T_1, \cdots, \Box T_l, \Box\Delta, q_1, \cdots q_k\ \langle\Box\Gamma|\Box\Sigma\rangle}\ (\Box)_s \quad (1 \leq i \leq m, 1 \leq j \leq l)$$

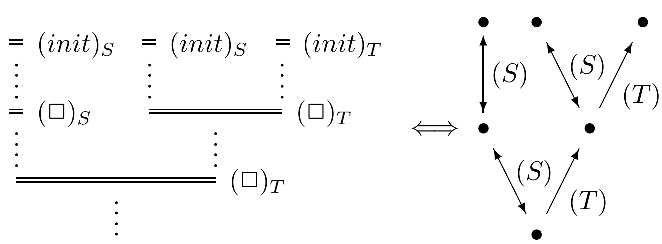
で始まっているとき。

$\Box\Gamma \not\rightarrow S_1\langle\Box\Gamma|\Box\Sigma,\Theta\rangle, \cdots, \Box\Gamma \not\rightarrow S_m\langle\Box\Gamma|\Box\Sigma,\Theta\rangle$ を上部の対称的ブロックの終式とし、それぞれが (M_{S_i}, w_{S_i}) に対応するとする。但し、$M_{S_i} = (W_{S_i}, R_{S_i}, \models)$ とする。一方、$\Box \not\rightarrow T_1\langle\Box\Gamma|\Box\Sigma,\Theta\rangle, \cdots, \Box \not\rightarrow T_l\langle\Box\Gamma|\Box\Sigma,\Theta\rangle$ を上部の推移的ブロックの終式とし、それぞれが (M_{T_i}, w_{T_i}) に対応するとする。但し、$M_{T_i} = (W_{T_i}, R_{S_i}, \models)$ とする。

このとき、この下式の対称的ブロックには (M, w) を対応させる。但し、$M = (W, R \models, w)$, $W = \{w\} \cup W_{S_1} \cup \cdots \cup W_{S_m} \cup W_{T_1} \cup \cdots \cup W_{T_l}$。また $R' = \{(w,w), (w, w_{S_1}), \cdots, (w, w_{S_m}), (w, w_{T_1}), \cdots, (w, w_{T_l})\} \cup \{(w_{S_1}, w), \cdots, (w_{S_m}, w)\} \cup R_{S_1} \cup \cdots \cup R_{S_m} \cup R_{T_1} \cup \cdots \cup R_{T_l}$ たとき、R は R' の推移的閉包とする。また $w \models p_1 \wedge \cdots \wedge p_n$ とする。

4. 推移的ブロックのとき、対称的ブロックの場合と同じ。

例 3.5.5 (反例) 反駁木から反例を作るためには、まず $(init)_s$, $(init)_t$, $(\Box)_s$,$(\Box)_t$ 規則の出現を見てクリプキフレームを作る。例えば、次のようである。



次に、$(init)_s$, $(init)_t$, $(\Box)_s$,$(\Box)_t$ 規則が適用された個所それぞれのの下式を見て付値を定める。例えば次のようになる。

$$\frac{}{\blacksquare\Gamma, \overrightarrow{p_3}\rightarrow\overrightarrow{q_3}, \Box B, \Box C\langle\Box\Gamma|\Box A, \Box B, \Box C\rangle}\ (init)_S \qquad \frac{}{\blacksquare\Gamma, \blacksquare P, \overrightarrow{p_5}\rightarrow\overrightarrow{q_5}, \Box B, \Box D\langle\Box\Gamma, \Box P|\Box B, \Box D\rangle}\ (init)_S \qquad \frac{}{\blacksquare\Gamma, \blacksquare P, \blacksquare Q, \overrightarrow{p_6}\rightarrow\overrightarrow{q_6}, \langle\Box\Gamma, \Box P, \Box Q|\epsilon\rangle}\ (init)_T$$

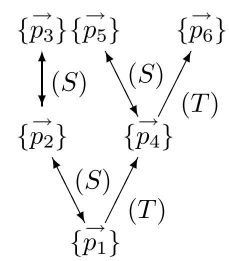
$$\frac{\Box\Gamma \rightarrow C\langle\Box\Gamma|\Box A, \Box B, \Box C\rangle}{\blacksquare\Gamma, \overrightarrow{p_2}\rightarrow\overrightarrow{q_2}, \Box A, \Box C\langle\Box\Gamma|\Box A, \Box B\rangle}\ (\Box)_S \qquad \frac{\Box\Gamma, \Box P \rightarrow B\langle\Box\Gamma, \Box P|\Box B, \Box D\rangle \qquad \Box\Gamma, \Box P \rightarrow D\langle\Box\Gamma, \Box P|\Box B, \Box D\rangle}{\blacksquare\Gamma, \blacksquare P, \overrightarrow{p_4}\rightarrow\overrightarrow{q_4}, \Box B, \Box D\langle\Box\Gamma, \Box P|\epsilon\rangle}\ (\Box)_T$$

$$\frac{\Box\Gamma \rightarrow A\langle\Box\Gamma|\Box A, \Box B\rangle \qquad \Box\Gamma \rightarrow B\langle\Box\Gamma|\Box A, \Box B\rangle}{\blacksquare\Gamma, \overrightarrow{p_1}\rightarrow\overrightarrow{q_1}, \Box A, \Box B\langle\Box\Gamma|\epsilon\rangle}\ (\Box)_T$$

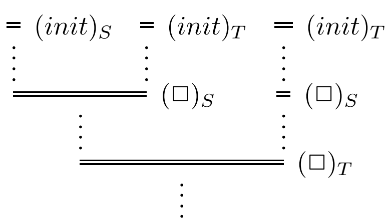
$$\vdots$$

$$\Delta \rightarrow \Sigma\langle\epsilon|\epsilon\rangle$$

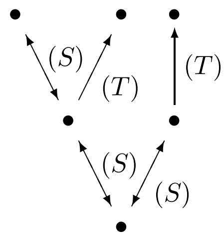
に対しては、次のクリプキモデルが対応する。



もう一つ例を示そう。



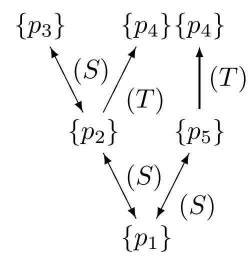
$\Longleftrightarrow$



さらに付値を定めると。

$$
\begin{prooftree}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_S$}
\UnaryInfC{$\blacksquare\Gamma, p_3 \to q_3, \Box A, \Box C$}
\noLine
\UnaryInfC{$\vdots$}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_T$}
\UnaryInfC{$\blacksquare\Gamma, \blacksquare P, p_4 \to q_4$}
\noLine
\UnaryInfC{$\vdots$}
\doubleLine
\RightLabel{$(\Box)_S$}
\BinaryInfC{$\blacksquare\Gamma, p_2 \to q_2, \Box C, \Box D$}
\noLine
\UnaryInfC{$\vdots$}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_T$}
\UnaryInfC{$\blacksquare\Gamma, \blacksquare P, p_4 \to q_4$}
\noLine
\UnaryInfC{$\vdots$}
\doubleLine
\RightLabel{$(\Box)_S$}
\UnaryInfC{$\blacksquare\Gamma, p_5 \to q_5, \Box D$}
\noLine
\UnaryInfC{$\vdots$}
\doubleLine
\RightLabel{$(\Box)_T$}
\BinaryInfC{$\blacksquare\Gamma, p_1 \to q_1, \Box A, \Box B$}
\noLine
\UnaryInfC{$\vdots$}
\end{prooftree}
$$

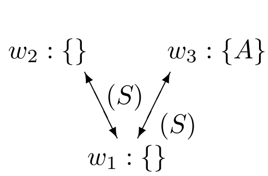
$\Longleftrightarrow$



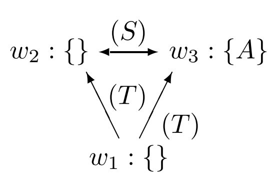
次に我々が導入した手法と、Goré 等が [9] で用いている反例の構成法との比較を行う。その違いを見るために $\to \neg\Box\neg(\Box A \vee \Box\neg A)$ に対する反駁木を考えてみる。以下 B は $\neg(\Box A \vee \Box\neg A)$ を表わす。

$$
\begin{prooftree}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_s$}
\UnaryInfC{$\blacksquare B \to A, \Box A, \Box\neg A\langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\to\vee)$}
\UnaryInfC{$\blacksquare B \to A, \Box A \vee \Box\neg A\langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\neg\to)$}
\UnaryInfC{$\blacksquare B, B \to A\langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box B \to A\langle \Box B | \Box A, \Box\neg A\rangle$}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_s$}
\UnaryInfC{$\blacksquare B, A \to \Box A, \Box\neg A\langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\to\vee)$}
\UnaryInfC{$\blacksquare B, A \to \Box A \vee \Box\neg A\langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\neg\to)$}
\UnaryInfC{$\blacksquare B, B, A \to \langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box B, A \to \langle \Box B | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\to\neg)$}
\UnaryInfC{$\Box B \to \neg A\langle \Box B | \Box A, \Box\neg A\rangle$}
\doubleLine
\RightLabel{$(\Box)_t$}
\BinaryInfC{$\blacksquare B \to \Box A, \Box\neg A\langle \Box B | \epsilon\rangle$}
\RightLabel{$(\to\vee)$}
\UnaryInfC{$\blacksquare B \to \Box A \vee \Box\neg A\langle \Box B | \epsilon\rangle$}
\RightLabel{$(\neg\to)$}
\UnaryInfC{$\blacksquare B, B \to \langle \Box B | \epsilon\rangle$}
\RightLabel{$(T)_t$}
\UnaryInfC{$\Box B \to \langle \epsilon | \epsilon\rangle$}
\RightLabel{$(\to\neg)$}
\UnaryInfC{$\to \neg\Box B(\equiv \neg\Box\neg(\Box A \vee \Box\neg A))\langle \epsilon | \epsilon\rangle$}
\end{prooftree}
$$

本手法では次の左側のモデルを与えるが、文献 [9] の手法では右側となる。



本手法



[9] の手法

[9] の構成法は次のようになものである。$(\Box)_t$ 規則の下式に $\Box A$ が存在し、$(\Box)_t$ 規則の左上式に A が存在するので、w_1 から w_2 に関係を付ける。同様に、w_1 から w_3 に関係を付ける。次に、左の $(init)_s$ に $\Box\neg A$ が存在し、$\neg A$ が右のシンメトリックイニシャルブロックに存在するので、w_2 から w_3 に関係を付ける。同様にして、w_3 から w_2 に関係が付けられる。そして、最後に推移律と反射律の閉包を取る。

このように、[9] の手法では、クリプキフレームを構成するためにはシーケントを構成する論理式に立ち入って調べる必要がある。このため、規則の種類のみからフレームを作る本手法よりも遥かに手間がかかる。

オーダーを考えてみれば、本手法ではブロックのサイズを n とすると、ブロックの先頭の推論規則が何であるかを調べればよいため、そのオーダーは $O(n)$ である。それに対して [9] の手法では、論理式がどこに出現しているかを探す必要があるため $O(n^2)$ 程度である。二分木を使えばさらに速くすることもできるが、それでも $O(nlog(n))$ 程度である。

3.5.3 完全性

前節のクリプキモデルは終式を反駁するので、様相論理 S4 の完全性の別証明が得られることになる。このことを詳しく見てみよう。

完全性を示すために、まずいくつかの部分論理式を $\perp$ で置き換えたシーケントが偽となることを示し、それから置き換えた部分論理式を少しずつ元に戻していく。そうすると最終的に元のシーケントが反駁されることが示される。

その証明はクラスターの構成に関する帰納法である。まずクラスターを定義する。

定義 3.5.6 (クラスター) クリプキモデル $(M, R, \models)$ の M の部分集合 M' が**クラスター**であるのは、
$\forall m, n \in M'(mRn$ かつ $nRm)$ かつ $\forall m \in (M \backslash M'), \forall m' \in M'(\neg(mRm'$ かつ $m'Rm))$ が成り立つときである。

あるいは、クラスターは次のように定義できる。*S4* クリプキモデル $(M, R, \models)$ に対して、関係 sin を $x, y \in M$ を、$x \sin y \Leftrightarrow xRy$ かつ yRx と定める。このとき、sin は同値関係となる。そうしたときに、sin の定める同値類一つ一つをクラスターと呼ぶ。

そして、$(M, R, \models)$ のクラスター M' が**イニシャルクラスター**であるとは、$\forall m \in (M \backslash M'), \forall m' \in M'(\neg(m'Rm))$ が成り立つことである。つまり、このクラスターから他のクラスターへの到達可能関係がないとき、イニシャルクラスターと呼ぶ。

クリプキモデルの可能世界がクラスターの**始点**であるとは、対応するブロックが推移的ブロックであるときである。

例 3.5.7 (クラスター)

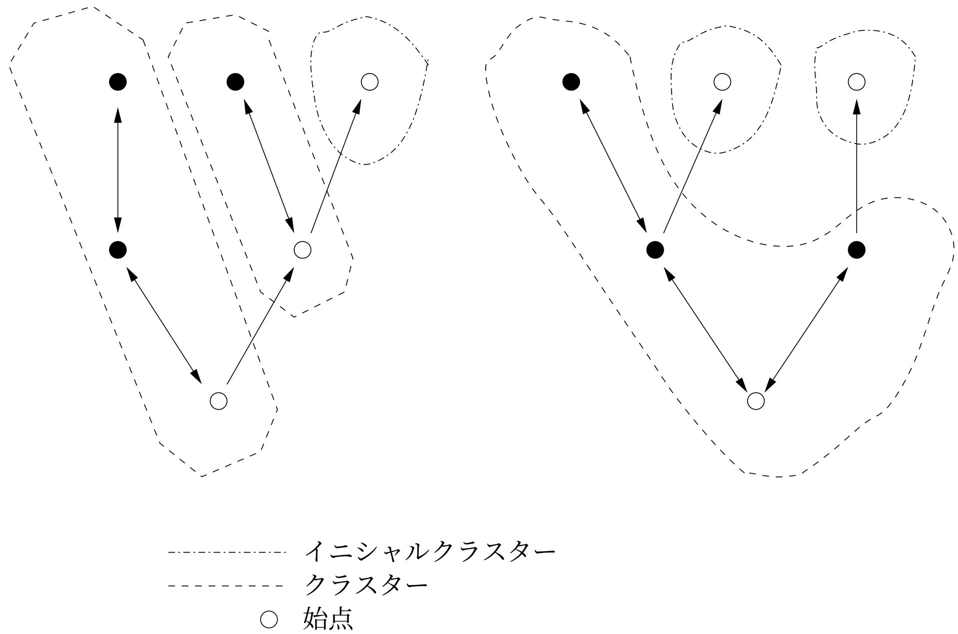


図 *3.14:* クラスターの例

図 3.14 のように、S4 の反駁木から構成された S4 モデルはクラスターの木構造を持ち、そのクラスターには必ず一つの始点がある。

定義 3.5.8 (ソートされた論理式列, ソートされたブロック) 論理式列 $\vec{A} = A_1, \cdots, A_n$ が**ソート**されているとは、$\forall i, j((i \geq j \Rightarrow len(A_i) \geq len(A_j))$ かつ $(i \neq j \Rightarrow A_i \neq A_j))$ が成り立つことである。

ブロックが $\Box\vec{A}$ によってソートされているとは、ブロック中の全ての T 規則

$$\frac{\blacksquare\Gamma, \blacksquare B, B, \Theta \not\to \Delta\langle\Box\Pi|\Box\Sigma\rangle}{\blacksquare\Gamma, \Box B, \Theta \not\to \Delta\langle\Box\Pi|\Box\Sigma\rangle}\ (T)$$

に対して、$\forall\blacksquare C \in \blacksquare\Gamma, \exists i(\Box C \equiv \Box A_i$ かつ $\exists j(\Box B \equiv A_j$ かつ $i \geq j))$ が成り立つことである。

補題 3.5.9 (規則の入れ換え) 上の規則の主論理式が下の規則の主論理式よりも長いときには、適用している推論規則の順序を入れ替えることができる。

(証明) 下図のように上の規則が $(\to\ \wedge)_1$ で下の規則が $(\to\supset)$ の場合を考える。$\mathrm{len}(A \wedge B) \leq \mathrm{len}(C \wedge D)$ かつ $\mathrm{len}(A) < \mathrm{len}(A \wedge B)$ かつ $\mathrm{len}(B) < \mathrm{len}(A \wedge B)$ であるから、$\mathrm{len}(A) < \mathrm{len}(C \wedge D)$ かつ $\mathrm{len}(B) < \mathrm{len}(C \wedge D)$ が成り立つ。すなわち、$A \neq C \wedge D$ かつ $B \neq C \wedge D$ が成り立つ。故に、この場合には下図のように規則を入れ替えることができる。他の規則の場合も同様である。

$$\cfrac{\cfrac{\Gamma, A \not\to B, C, \Delta}{\Gamma, A \not\to B, C \wedge D, \Delta}\ (\to\wedge)_1}{\Gamma \not\to A \supset B, C \wedge D, \Delta}\ (\to\supset) \quad \underset{\Longrightarrow}{\text{入れ換え}} \quad \cfrac{\cfrac{\Gamma, A \not\to B, C, \Delta}{\Gamma, \not\to A \supset B, C, \Delta}\ (\to\supset)}{\Gamma \not\to A \supset B, C \wedge D, \Delta}\ (\to\wedge)_1$$

補題 3.5.10 (ソートされたブロック) あるブロックが $\blacksquare\Gamma, p_1, \cdots, p_n \not\to q_1, \cdots, q_m, \Box\Delta\ \langle\Box\Pi, \Box\Sigma\rangle$ で始まり $\Gamma' \to \Delta'\langle\Box\Pi, \Box\Sigma\rangle$ で終わるとする。このとき、ソートされた論理式列 $\Box\vec{A}\ (\supset \Box\Gamma)$ に対して $\blacksquare\Gamma, p_1, \cdots, p_n \not\to q_1, \cdots, q_m, \Box\Delta\ \langle\Box\Pi, \Box\Sigma\rangle$ で始まり $\Gamma' \to \Delta'\langle\Box\Pi, \Box\Sigma\rangle$ で終わるソートされたブロックが存在する。

(証明) 補題 3.5.9 より、主論理式の長さが下式よりも上式が短かくなるように並べ替えることができる。また、主論理式の長さが同じであるときにも、規則を入れ替えることができるので、ソートされた $\Box\overrightarrow{A}$ ($\supset \Box\Gamma$) に対してソートされているように並べ替えることができる。

定義 3.5.11 (ポジティブ代入とネガティブ代入) $\overrightarrow{A}\equiv A_1,\cdots,A_n,\ \overrightarrow{B}\equiv B_1,\cdots,B_n,\ [\overrightarrow{A}{:=}\overrightarrow{B}]\equiv[A_1:=B_1,\cdots,A_n:=B_n]$ とする。このとき、 **ポジティブ代入** $C\,pos[\overrightarrow{A}{:=}\overrightarrow{B}]$ と**ネガティブ代入** $C\,neg[\overrightarrow{A}{:=}\overrightarrow{B}]$ を以下のように定義する。

ある i に対して $C\equiv A_i$ のとき、
$C\,pos[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv B_i$

それ以外のときは以下のように C に対して帰納的に定義する。

1. $C\equiv p$ のとき。
$p\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv p$
$p\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv p$

2. $C\equiv D\wedge E$ のとき。
$(D\wedge E)\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\wedge E\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(D\wedge E)\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\wedge E\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]$

3. $C\equiv D\vee E$ のとき。
$(D\wedge E)\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\vee E\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(D\wedge E)\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\vee E\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]$

4. $C\equiv D\supset E$ のとき。
$(D\supset E)\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\supset E\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(D\supset E)\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv D\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\supset E\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]$

5. $C\equiv\neg D$ のとき。
$(\neg D)\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv\neg D\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(\neg D)\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv\neg D\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]$

6. $C\equiv\Box D$ のとき。
$(\Box D)\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv\Box D\,\mathrm{pos}[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(\Box D)\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]\equiv\Box D\,\mathrm{neg}[\overrightarrow{A}{:=}\overrightarrow{B}]$

定義 3.5.12 (シーケントに対するポジティブ代入とネガティブ代入) シーケント $C_1,\cdots,C_n\to D_1,\cdots,D_m$ に対してポジティブ代入とネガティブ代入を以下のように拡張する。
$(C_1,\cdots C_n\to D_1,\cdots D_m)pos[\overrightarrow{A}{:=}\overrightarrow{B}]$
$\equiv C_1neg[\overrightarrow{A}{:=}\overrightarrow{B}],\cdots,C_nneg[\overrightarrow{A}{:=}\overrightarrow{B}]\to D_1pos[\overrightarrow{A}{:=}\overrightarrow{B}],\cdots,D_mpos[\overrightarrow{A}{:=}\overrightarrow{B}]$
$(C_1,\cdots C_n\to D_1,\cdots D_m)neg[\overrightarrow{A}{:=}\overrightarrow{B}]$
$\equiv C_1pos[\overrightarrow{A}{:=}\overrightarrow{B}],\cdots,C_npos[\overrightarrow{A}{:=}\overrightarrow{B}]\to D_1neg[\overrightarrow{A}{:=}\overrightarrow{B}],\cdots,D_mneg[\overrightarrow{A}{:=}\overrightarrow{B}]$

例 3.5.13 (ポジティブ代入とネガティブ代入)
$(A\wedge B,C\to C)neg[A\wedge B:=D',A:=E',C:=F']$
$\equiv(A\wedge B)pos[A\wedge B:=D',A:=E',C:=F'],Cpos[A\wedge B:=D',A:=E',C:=F']\to Cneg[A\wedge B:=D',A:=E',C:=F']$
$\equiv D',F'\to C$

補題 3.5.14 *(i)* $w \not\models A$ のとき $w \not\models A\,\mathrm{pos}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。
(ii) $w \models A$ のとき $w \models A\,\mathrm{neg}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。

(証明) $A \equiv B$ で $w \not\models A$ のとき。$A\,\mathrm{pos}[\overrightarrow{B}{:=}\overrightarrow{\bot}] \equiv \bot$。ゆえに $w \not\models A$。
$A \equiv B$ で $w \models A$ のとき。$A\,\mathrm{neg}[\overrightarrow{B}{:=}\overrightarrow{\bot}] \equiv A$。ゆえに $w \models A$。

$A \not\equiv B$ のとき、A の構造に関する (i),(ii) の同時帰納法で証明する。
(i) で $A \equiv \Box C$ のとき。$w \not\models \Box C$ であるから、$\exists v(w \to v) v \not\models C$。帰納法の仮定より、$v \not\models C\,\mathrm{pos}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。よって、$w \not\models \Box(C\,\mathrm{pos}[\overrightarrow{B}{:=}\overrightarrow{\bot}])$。故に $w \not\models (\Box C)\,\mathrm{pos}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。

(ii) で $A \equiv \Box C$ のとき。$w \models \Box C$ であるから、$\forall v(w \to v) v \models C$。帰納法の仮定より、$v \models C\,\mathrm{neg}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。よって、$w \models \Box(C\,\mathrm{neg}[\overrightarrow{B}{:=}\overrightarrow{\bot}])$。故に、$w \models (\Box C)\,\mathrm{neg}[\overrightarrow{B}{:=}\overrightarrow{\bot}]$。

その他の場合は明らか。

補題 3.5.15 クリプキモデルのある可能世界が $\wedge, \vee, \supset, \neg$ 規則の上式を反駁するとき、その可能世界はまた下式も反駁する。

(証明) 自明。

定理 3.5.16 本手法により構成されたクリプキモデルは反例となる。

(証明) クラスターの構成に関する帰納法。

一点からなるイニシャルクラスターのとき。この場合対応するブロックは推移的イニシャルブロックとなり、始式は $\blacksquare\Gamma, p_1, \cdots, p_n \to q_1, \cdots, q_m$ という形をしている。このシーケントは対応する可能世界 $\{p_1, \cdots, p_n\}$ で反駁される。なぜなら、$\blacksquare\Gamma$ は解釈上無視され、このシーケントの右辺には $\Box$ 論理式が出現しないからである。前補題より、T 規則以外では下式も順次反駁される。そして、クラスターはイニシャルクラスターとなるので、この可能世界からの他の可能世界への到達可能関係はない。よって、T 規則の下式も反駁される。
故に、このブロックの終式も反駁される。

二点以上からなるイニシャルクラスターのとき。

1. このクラスターに対応するブロックの妥当な履歴を Γ とし、この Γ をソートしたものを $\overrightarrow{\Gamma}$ で表す。
2. 対応する全てのブロックを $\overrightarrow{\Gamma}$ でソートする。(補題 3.5.10 より)
3. 対応するブロックの非妥当な履歴の和集合を Σ とする。
4. 対応するブロックに出現する全てのシーケントに $pos[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}]$ を代入したブロックを考える。すると、ブロックの始式は対応する可能世界で反駁される。なぜなら、代入前の始式は $\blacksquare\Gamma, \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\Box A}$ という形をしていて、$(\blacksquare\Gamma, \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\Box A})\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}] \equiv \blacksquare\Gamma\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}], \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\bot}$ となり、右辺の $\Box$ 論理式はすべて $\bot$ で置き換えられるからである。
5. $\wedge, \vee, \supset, \neg$ 規則のときは、補題 3.5.15 より、下式も反駁される。
6. (T), $(T)_t$ 規則のときは、$\dfrac{\blacksquare\Gamma', \blacksquare B, B, \Delta \to \Pi}{\blacksquare\Gamma', \Box B, \Delta \to \Pi}\ (T)$ という形をしている。このとき、各ブロックはソートされているので、全て対応する可能世界で B は妥当である。すなわち、そのクラスターで B が妥当であるので、$\Box B$ も妥当である。故に、T 規則の下式は反駁される。
7. 5. と 6. を繰り返すと、$\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}]$ を代入した各終式が反駁されることが分かる。

8. Σ の中で最も複雑さが小さい論理式を C とする。このとき、C は対応するブロックの終式の右辺のいずれかに出現する。また、$C\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}] \equiv C$ である。なぜなら、C は全ての $\Box\Sigma$ より小さいから代入が生じないからである。よって、C はいずれかの可能世界で反駁される。

9. $\Box\Sigma$ として $\Box\Sigma\backslash\{\Box C\}$ をとり、4. に戻ってこの手続きを繰り返す。

10. $\Sigma \equiv \epsilon$ となったとき、全てのブロックの終式は反駁されることが示される。

一点からなるイニシャルではないクラスターのとき。対応するブロックは、

$$\cfrac{\Box\Gamma \not\to A_1 \ \langle \Box\Gamma | \Box\Theta \rangle \quad \cdots \quad \Box\Gamma \not\to A_m \ \langle \Box\Gamma | \Box\Theta \rangle}{\blacksquare\Gamma, p_1, \cdots, p_n \not\to \Box A_1, \cdots, \Box A_m, q_1, \cdots q_l \ \langle \Box\Gamma | \epsilon \rangle} \ (\Box)_t$$

から始まる推移的ブロックであり、可能世界の付値は $\{p_1, \cdots, p_n\}$ である。このとき、このブロックの始式はこの可能世界で反駁される。なぜなら、到達可能関係が付いた可能世界で $A_1, \cdots, A_m$ が反駁されているので、この可能世界では $\Box A_1, \cdots, \Box A_m$ が反駁されるからである。

$\wedge, \vee, \supset, \neg$ 規則のときは、補題 3.5.15 より、下式も反駁される。

(T) あるいは $(T)_t$ 規則のときは、$\dfrac{\blacksquare\Gamma, \blacksquare A, A, \Delta \to \Pi}{\blacksquare\Gamma, \Box A, \Delta \to \Pi}\ (T)$ という形をしているが、到達可能関係が付いた可能世界では $\Box\Gamma, \Box A$ が妥当であるから Γ, A も妥当。よって、この可能世界でも $\Box A$ が妥当となり、結局下式が反駁される。

故に、終式が反駁される。

二点以上からなるイニシャルではないクラスターのとき。

1. このクラスターに対応するブロックの妥当な履歴を Γ とし、この Γ をソートしたものを $\overrightarrow{\Gamma}$ で表す。

2. 対応する全てのブロックを $\overrightarrow{\Gamma}$ でソートする。(補題 3.5.10 より)

3. 対応するブロックの非妥当な履歴の和集合を Σ とする。

4. I.H. より、このクラスターから到達可能な他のクラスターの各シーケントは対応する可能世界で反駁される。

5. 対応するブロックに出現する全てのシーケントに $\mathrm{pos}[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}]$ を代入したブロックを考える。このクラスターから到達可能な他のクラスターに対応したブロックの各シーケントに対しては、補題 3.5.14 より、代入したシーケントも反駁される。よって、ブロックの始式は対応する可能世界で反駁される。なぜなら、代入前の始式は $\blacksquare\Gamma, \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\Box A}$ という形をしていて、$(\blacksquare\Gamma, \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\Box A})\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}] \equiv \blacksquare\Gamma\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}], \overrightarrow{p} \to \overrightarrow{q}, \overrightarrow{\bot}, \overrightarrow{\Box A'}$ $(\overrightarrow{\Box A'} \subset \overrightarrow{\Box A}\ \mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}])$ となり、$\overrightarrow{\Box A'}$ は到達可能な他のクラスターに対応したブロックの終式に出現するからである。

6. $\wedge, \vee, \supset, \neg$ 規則のときは、補題 3.5.15 より、下式も反駁される。

7. (T), $(T)_t$ 規則のときは、$\dfrac{\blacksquare\Gamma', \blacksquare B, B, \Delta \to \Pi}{\blacksquare\Gamma', \Box B, \Delta \to \Pi}\ (T)$ という形をしている。このとき、各ブロックはソートされているので、全て対応する可能世界で B は妥当である。すなわち、そのクラスターで B が妥当であるので、$\Box B$ も妥当である。故に、T 規則の下式は反駁される。

8. 6. と 7. を繰り返すと、$\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}]$ を代入した各終式が反駁されることが分かる。

9. Σ の中で最も複雑さが小さい論理式を C とする。このとき、C は対応するブロックの終式の右辺のいずれかに出現する。また、$C\mathrm{pos}\,[\overrightarrow{\Box\Sigma}{:=}\overrightarrow{\bot}] \equiv C$ である。なぜなら、C は全ての $\Box\Sigma$ より小さいから代入が生じないからである。よって、C はいずれかの可能世界で反駁される。

10. $\Box\Sigma$ として $\Box\Sigma\backslash\{\Box C\}$ をとり、5. に戻ってこの手続きを繰り返す。

11. $\Sigma \equiv \epsilon$ となったとき、全てのブロックの終式は反駁されることが示される。

3.6 結論

本論文では、古典命題論理、様相論理 K, KT, S4 では反駁木から効率的に反例がを求める証明反駁アルゴリズムを与えた。

Goré[9] 等にある飽和 (saturated) を用いた完全性の証明では、K, KT では本論文と同じく反例は木構造で与えられるが、S4 では反例はより複雑なグラフとなり、それ以上の特質を与えることはできない。また、反駁木における論理式の出現を追い掛ける必要があるため、多くの手間が掛かってしまう。

それに対して本論文では、始式と (□) 規則を s と t の二つに分け、このどちらが適用されたかによって、反例のフレームを構成していく。つまり、反駁木に出現する適用規則から直接的にフレームを構成するのである。この結果、効率的に反例が構成できると同時に、クラスターの木構造で構成できるという新な反例の性質を示すことでできた。

それは、次のようなクラスターの木構造である。クラスターの高さが高くなればなるほど、妥当な □ 論理式が増加するという構造である。このことを図示すると図 3.15 のようになる。

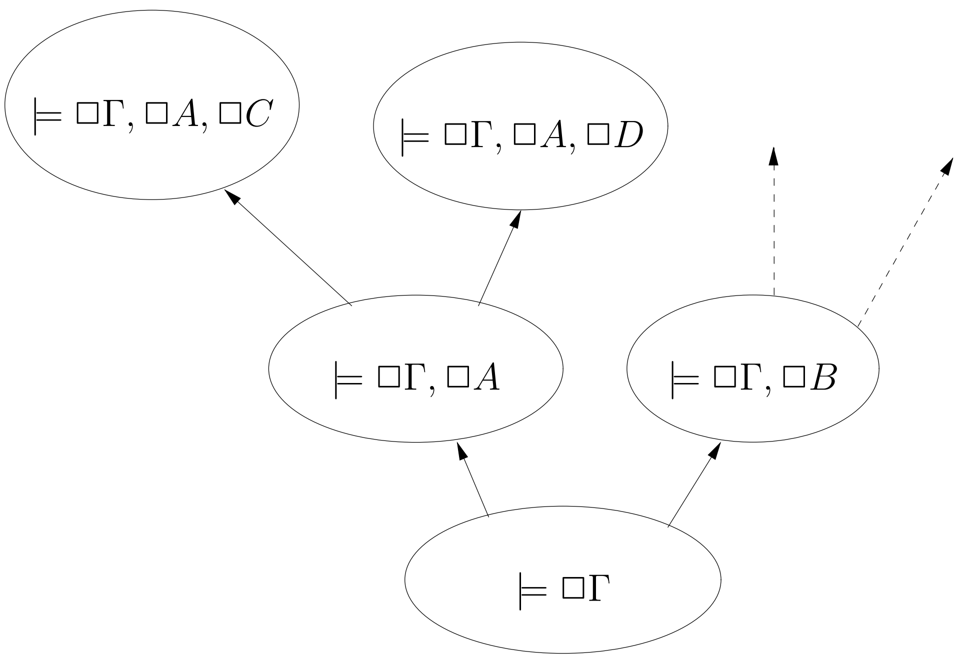


図 3.15: S4 の反例の構造

この結果の主な内容のうち、様相論理 K および KT については 2001 年の the 34th MLG meeting[21] で発表され、同じく 2001 年の Bulletin of Section of Logic[23] に掲載されている。また、様相論理 S4 については同じく 2001 年の RPC '01[24] で発表されている。

様相論理については、小野寛晰 [26] を参照のこと。証明反駁アルゴリズム、及び決定手続きに関しては Alain Heuerding[10]、Lincoln A. Wallen[37]、R. Goré[9] で詳しく論じられている。

第4章　証明支援システムxpe

本章では証明支援システム xpe の概要を示す。詳細は付録の xpe のマニュアルを参照のこと。

4.1　はじめに

論理学の計算機科学への貢献は大きい。人工知能を始めとし様々な分野に応用されている。それどころか、計算機そのものが論理回路で構成されていることから、計算機科学そのものが論理学の上に成り立っているといっても過言ではないだろう。しかしながら、その逆はどうであろうか。すなわち、計算機はどの程度論理学に貢献しているのだろうか。

定理自動証明などの分野では、計算機によって様々な未解決問題が解かれてきている。しかしながら、それらは研究対象として定理自動証明の実装を行っているのであり、論理学者が道具として使える実用化がなされているとは思えない。

この場合の道具としての実用化とは何であるか。今、電卓を例えに挙げる。何か計算を行うとき、電卓はボタンさえ押せば答えを出してくれる。関数電卓を用いれば、さらに複雑な計算さえこなしてくれる。同じようなことが論理学に対して計算機で行えないだろうか。つまり論理学を研究する上での計算や証明を肩代わりする計算機である。

本章で述べる**証明支援システム xpe** は、そんな論理学者にとって実用に耐えうる道具を提供するものである。この xpe の機能の概要を説明していく。

4.2 証明図の作成

TEX で証明図を描くにはどうするであろうか。一般には龍田真氏作成のマクロ proof.sty が用いられる。このマクロを用いれば、どんあ複雑な証明図も TEX 上で描くことが可能ではあるが、実際には大きな問題を腫らんでいる。それは、証明図という木構造を括弧 $\{\cdots\}$ を用いて表現するため、証明図が複雑になるにつれて括弧が複雑になりすぎて手に負えなくなってくる。

例を上げよう。次の LK の排中律の証明図を考える。

$$\dfrac{\dfrac{\dfrac{\dfrac{A \to A}{\to A, \neg A}\,(\neg R)}{\to A, A \lor \neg A}\,(\lor R)}{\to A \lor \neg A, A \lor \neg A}\,(\lor R)}{\to A \lor \neg A}\,(Co)$$

このとき、TEX のソースは次のようになる。

```
\infer[(Co)]{\to A \lor \lnot A}
  {\infer[(\lor R)]{\to A \lor \lnot A, A \lor \lnot A}
    {\infer[(\lor R)]{\to  A, A \lor \lnot A}
      {\infer[(\lnot R)]{\to  A, \lnot A}
        {A \to  A}
      }
    }
  }
```

このようにソースでは括弧 $\{\cdots\}$ が数多く用いられている。今、括弧のどれかが欠けてしまった場合を考えてみよう。この場合その括弧を復元することは多くの場合困難である。それは、括弧が多いこと、そしてエラーが別の行で報告されることが多いためである。また、証明図では出現する論理式の上の行と下の行で違いがわずかであるため、プレビューで間違いを発見したとしても、その行を特定することが困難になる。このため、証明図を証明図のまま見ながらエディットできるソフトウェアが必要であった。

こうした要求を満たすべく開発されたのが xpe である。本ソフトウェアにより、証明図の作成がたやすくなり、また再利用性が高まった。

xpe 上でこの排中律の証明図を描くには次のようにすればよい。

1. xpe の初期画面は  となっている。ここに`\to A \lor \lnot A` を入力し Return–key を押す。そうすると画面は  となる。

2.  の上部を左クリックする。

   画面: 

3. `\to A \lor \lnot A, A \lor \lnot A` を入力し Return を押す。

   画面:  。

4. 以下同様に、

   `\to \lnot A, A \lor \lnot A`

```
\to \lnot A, A
```

```
A \to A
```

を入力していくと、

画面: 

という LK の排中律の証明図が完成しているはずである。

5. カットアンドペーストでエディタに貼り付けると次のテキストが挿入される。

```
\infer{\to A \lor \lnot A}
  {\infer{\to A \lor \lnot A, A \lor \lnot A}
    {\infer{\to \lnot A, A \lor \lnot A}
      {\infer{\to \lnot A, A}
        {A \to A}
      }
    }
  }
```

このように、普段紙に証明図を書くように計算機上で証明図を作成できる。それどころか、補完が効くので紙に書くよりも効率的な部分がある。

さらに定理自動証明や証明反駁アルゴリズム等を xpe 上に実装することにより、より実用的な証明支援システムとして発展しつつある。

4.3 定理自動証明

一般的な定理自動証明システムでは、与えられた論理式に対して証明可能か否かのみを返すようになっている。しかしこれでは論理学を研究する上では不便である。証明可能なときにはその証拠、すなわち証明図がどのように構成されるのか、また証明不可能なときにはその証拠すなわち、反例がどのような構造をしているのかといった分析が重要である。

そこで、xpe 上に実装される定理自動証明は証明図の出力を主眼に置いている。証明可能な論理式に対しては証明図の出力と伴に証明可能性を判断するようになっている。また次節で示すように、一部の体系では証明反駁システムとして実装し、証明不可能なときに反例をも出力する。

xpe 上に実装されている定理自動証明は次の通りである。いずれも命題論理である。

- 古典論理
- 直観主義論理
- 様相論理 (K,KT,S4,S5,K5,KD5)
- 部分構造論理 (FLe,FLew,FLec)
- ラムダ項の型付けシステム

これらは完全に外部コマンドとして実装されている。次にあるのはディレクトリ`/usr/X11R6/lib/X11/xpe/prover` の `ls` の結果である。

```
$ cd /usr/X11R6/lib/X11/xpe/prover
$ ls -l
合計 960
drwxr-xr-x    2 root     root           85 Sep 29 17:10 check/
-rwxr-xr-x    1 root     root       101846 Sep 29 17:10 classical*
-rwxr-xr-x    1 root     root         1722 Sep 29 17:10 clprover*
-rwxr-xr-x    1 root     root        99371 Sep 29 17:10 fle*
-rwxr-xr-x    1 root     root       111397 Sep 29 17:10 flec*
lrwxrwxrwx    1 root     root            3 Sep 29 17:10 flew -> fle*
-rwxr-xr-x    1 root     root       107403 Sep 29 17:10 il*
-rwxr-xr-x    1 root     root       125391 Sep 29 17:10 k*
-rwxr-xr-x    1 root     root       100620 Sep 29 17:10 k+*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 k5 -> k*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 k5-like -> k*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 kd5 -> k*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 kd5-like -> k*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 kt -> k*
-rwxr-xr-x    1 root     root       105878 Sep 29 17:10 kt+*
-rwxr-xr-x    1 root     root         1533 Sep 29 17:10 llprover*
drwxr-xr-x    2 root     root           59 Sep 29 17:10 parser/
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 s4 -> k*
-rwxr-xr-x    1 root     root       113472 Sep 29 17:10 s4+*
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 s5 -> k*
```

```
lrwxrwxrwx    1 root     root            1 Sep 29 17:10 s5-like -> k*
drwxr-xr-x    2 root     root          131 Sep 29 17:10 trace/
lrwxrwxrwx    1 root     root            9 Sep 29 17:10 truth-table -> classical*
-rwxr-xr-x    1 root     root        81187 Sep 29 17:10 type-assign*
```

このように定理自動証明の体系の名前が付いた実効可能なファイルがあることが分かる。それぞれがその体系に対応した定理自動証明のプログラムである。一部はシンボリックリンクになっているが、それは 1 つのプログラムをどういう名前で呼び出されたかによって体系を切り替えているのである。特に様相論理では、推論規則を加えるか否かで別の体系になるため、全てが 'k' という一つのプログラムへのシンボリックリンクになっている。

尚 'k+' のように'+' が付いたものは前章で説明した証明反駁アルゴリズムの実装である。

これら定理自動証明プログラムは xpe とは完全に独立している。xpe からは入力された論理式をこれらのプログラムに渡し、返り値を xpe に戻すという単純なものである。例えば、排中律 $A \vee \neg A$ は `A \lor \lnot A` と TEX で書けるが、次のように古典論理の定理自動証明プログラム `classical` を実効してこの論理式を入力してみると、

```
$ ./classical
A \lor \lnot A ^D
%%It's provable.
\infer [(\lor R)]{\to A \lor \lnot A }
 {\infer [(\lnot R)]{\to A , \lnot A }
 {A \to A }
}
```

というように、証明図

$$\dfrac{\dfrac{A \to A}{\to A, \neg A}\,(\neg R)}{\to A \vee \neg A}\,(\vee R)$$

の TEX のソースが返ってくる。xpe ではこのように入力された論理式を独立した自動証明プログラムに渡し返り値を評価することでシステム全体が構成されている。そのため拡張は用意であり、また既存の定理自動証明系との接続も用意である。実際、神戸大の田村直之氏の線型論理の自動証明プログラムとインターネットを通して接続され、xpe 上からシームレスに利用できる。

実装されていない体系での証明を行うとき、途中までの証明図を手で書き残りを自動証明させたいときがある。例えば、様相論理 K に公理 $\Box(\Box A \supset A) \supset \Box A$ を加えた体系 (この論理は証明可能性の論理と呼ばれる) で、$\Box\Box A \to \Box$(4 の公理) が証明可能であることを調べる場合である。このとき、

$$\dfrac{\Box A \vdash \Box(\Box(A \wedge \Box A) \supset A \wedge \Box A) \qquad \dfrac{\dfrac{\Box(\Box A \supset A) \vdash \Box A}{\Box(\Box(A \wedge \Box A) \supset A \wedge \Box A) \vdash \Box(\Box A \wedge A)}\,(Sub) \qquad \Box(\Box A \wedge A) \vdash \Box\Box A}{\Box(\Box(A \wedge \Box A) \supset (A \wedge \Box A)) \vdash \Box\Box A}\,(Cut)}{\Box A \vdash \Box\Box A}\,(Cut)$$

と Cut を適用した証明図を xpe に入力する。後は $\Box A \vdash \Box(\Box(A \wedge \Box A) \supset A \wedge \Box A)$ と $\Box(\Box A \wedge A) \vdash \Box\Box A$ が閉じていないので、ここをクリックして K の自動証明に掛ければ、次のように証明図全体が完成する。

$$\dfrac{\dfrac{\dfrac{\dfrac{A, \Box(A \wedge \Box A) \to A \qquad \dfrac{\dfrac{A, \Box A \to A}{A \wedge \Box A \to A}\,(\wedge L)}{A, \Box(A \wedge \Box A) \to \Box A}\,(\Box)}{A, \Box(A \wedge \Box A) \to A \wedge \Box A}\,(\wedge R)}{A \to \Box(A \wedge \Box A) \supset A \wedge \Box A}\,(\supset R)}{\Box A \to \Box(\Box(A \wedge \Box A) \supset A \wedge \Box A)}\,(\Box) \qquad \dfrac{\dfrac{\Box(\Box A \supset A) \vdash \Box A}{\Box(\Box(A \wedge \Box A) \supset A \wedge \Box A) \vdash \Box(\Box A \wedge A)}\,(Sub) \qquad \dfrac{\dfrac{\Box A, A \to \Box A}{\Box A \wedge A \to \Box A}\,(\wedge L)}{\Box(\Box A \wedge A) \to \Box\Box A}\,(\Box)}{\Box(\Box(A \wedge \Box A) \supset (A \wedge \Box A)) \vdash \Box\Box A}\,(Cut)}{\Box A \vdash \Box\Box A}\,(Cut)$$

このように対話的に定理自動証明を利用することができる。

次に個々の定理自動証明プログラムについて簡単に説明する。

4.3.1 古典論理

ワンの体系 [38] で実装されている。[26] では分解と呼ばれている。この体系で証明不可能なときはそのまま反駁木になるため、証明反駁プログラムでもある。

4.3.2 直観主義論理

シーケントの両辺を多重集合として考え、推論規則を適用するごとに同じシーケントが現れていないかをチェックしている。そのため、入力するシーケントによっては遅くなるときがある。将来的には、[5],[11] に示されているループ無しの体系に移行する予定である。

4.3.3 様相論理 K,KT,S4,S5,K5,KD5

シーケントの両辺を集合として考え、論理式の読み込み時に証明図に出現する論理式のデータベースを作成し、シーケントをビット列で表現することで最適化している。

また、証明可能性の判定よりも、速度を犠牲にしても実際に証明図を出力するアルゴリズムを採用している。例えば S5 では S4 への埋め込みにより高速に証明可能性の判定が可能である。しかしながら、それでは S5 の証明図は得られない。論理学を研究する上では証明可能なときには、実際にどのように証明図が構成されるのかという分析が重要である。そこで、高野の手法 [32] を用いて Cut 規則が出現するアルゴリズムを採用している。このアルゴリズムでは Cut を取る際に同じシーケントが出現していないかを調べる必要があるため、それほど速くはないが可読性の高い証明図を出力する。例えば、$A \to \Box\neg\Box\neg A$ を証明するときには Cut 規則が必要であるが、xpe は次のような Cut 規則を含む証明図を出力する。

$$\dfrac{\dfrac{\dfrac{\Box\neg A \to \Box\neg\Box\neg A, \Box\neg A}{\to \Box\neg\Box\neg A, \neg\Box\neg A, \Box\neg A}\,(\neg R)}{A \to \Box\neg\Box\neg A, \Box\neg A}\,(5) \quad \dfrac{\dfrac{\Box\neg A, A \to \Box\neg\Box\neg A, A}{\Box\neg A, \neg A, A \to \Box\neg\Box\neg A}\,(\neg L)}{\Box\neg A, A \to \Box\neg\Box\neg A}\,(T)}{A \to \Box\neg\Box\neg A}\,(Cut)$$

4.3.4 ラムダ項の型付けシステム

ラムダ項を入力すると、型付け可能なときはその分解木と型を返し、型付け不能なときは分解木と単一化不能な型を指し示す。

一般的な定理自動証明系では読み込みの制限から扱えるのは論理式のみであり、ラムダ項を同じ枠組みで扱うことはできない。しかしながら、xpe ではユーザーインターフェイスと定理自動証明プログラムは独立であるため、TeX で扱えるような数式ならばどんな数式でも扱うことができる。

入力として $\lambda xyz.xz(yz)$ すなわち`\lambda xyz.xz(yz)`を入力したとき、その分解木と型として次の出力を得る。

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[x]^1 : A \to B \to C \quad [z]^2 : A}{xz : B \to C}\,(\to E) \quad \dfrac{[y]^3 : A \to B \quad [z]^2 : A}{yz : B}\,(\to E)}{xz(yz) : C}\,(\to E)}{\lambda z.xz(yz) : A \to C}\,(\to I)^2}{\lambda yz.xz(yz) : (A \to B) \to A \to C}\,(\to I)^3}{\lambda xyz.xz(yz) : (A \to B \to C) \to (A \to B) \to A \to C}\,(\to I)^1}{}$$

4.4 証明反駁アルゴリズム

論理学の研究では、ある論理式が証明不可能なときにその反例がどのように構成されるかという分析が重要である。しかしながら、一般的には証明不可能な論理式に対してそれを反駁するような反例をつねに具体的に構成できるわけではない。完全性定理により反例の存在が保証され、有限モデル性により有限ノードの反例の存在が保証されていても、その反例をしらみつぶしに探すのでは実用的なものとはならない。そこでこれらの性質の証明とは別に、反例を構成するようなアルゴリズムを提出する必要がある。

証明可能な論理式に対してその証拠、すなわち証明図を出力し、証明不可能な論理式に対してその証拠、すなわち反例を出力するアルゴリズムを証明反駁アルゴリズムと呼ぶ。このアルゴリズムは決定手続きの特別な場合であり、この実装はまた定理自動証明プログラムの特別な場合にもなっている。

一般的にいえば、決定手続きにより証明探索に失敗したときには、与えられた論理式が証明できない証拠を含んでいるはずである。そこから反例を抽出することができないであろうか。これが証明反駁アルゴリズムを考える動機付けの一つである。しかしながら、一般的には抽出することはできない。なぜなら、決定手続きの証明では元となる形式体系との同値性が示される場合が多く、モデルとの関係が示されないからである。

そこで前章のように古典論理、様相論理 K,KT,S4 に対して証明反駁アルゴリズムを与えた。本節では、その出力を簡単に眺めていく。

4.4.1 古典論理

パースの定理 $((p \supset q) \supset p) \supset p$ すなわち入力 `((p \supset q) \supset p) \supset p` に対しては次のような証明図と伴に `It's provable.` と出力する。

$$\dfrac{\dfrac{\dfrac{p \to p, q}{\to p, p \supset q}\ (\supset R) \qquad p \to p}{(p \supset q) \supset p \to p}\ (\supset L)}{\to ((p \supset q) \supset p) \supset p}\ (\supset R)$$

これに少し変更を加えて証明不可能な論理式 $((p \supset q) \supset r) \supset p$ すなわち `((p \supset q )\supset r )\supset p` では次のような反駁木と伴に反例として `Counter: r = t / p = f` を出力する。これは r が true、p が false のとき (q が ture であっても false であっても) この論理式が false になることを意味している。

$$\dfrac{\dfrac{\dfrac{p \to p, q}{\to p, p \supset q}\ (\supset R) \qquad r \to p}{(p \supset q) \supset r \to p}\ (\supset L)}{\to ((p \supset q) \supset r) \supset p}\ (\supset R)$$

4.4.2 様相論理 K

メニューの'k+' が様相論理 K の証明反駁プログラムである。論理式 $\Box(A \land B) \to \Box A \lor \Box C$ すなわち `\Box (A \land B )\to \Box A \lor \Box C` に対しては次の証明図と `Provable` という出力を得る。

$$\dfrac{\dfrac{\dfrac{A, B \to A}{A \land B \to A}\ (\land L) \qquad \dfrac{A, B \to C}{A \land B \to C}\ (\land L)}{\Box(A \land B) \to \Box A, \Box C}\ (\Box)}{\Box(A \land B) \to \Box A \lor \Box C}\ (\lor R)$$

ここで $(\Box)$ 規則は OR–分岐であり、$A \land B \to C$ は証明不可能であるが、$A \land B \to A$ が証明可能であるため全体として証明可能になる。

さらに変更を加えて $\Box(A \land B) \to \Box(A \land D) \lor \Box C$ すなわち `\Box (A \land B )\to \Box (A \land D )\lor \Box C` の場合には次のような出力と伴に `Not provable.  Write with the counter model.` が出力される。

$$\dfrac{\dfrac{\dfrac{A, B \to A}{A \land B \to A}\,(\land L) \quad \dfrac{\dfrac{\dfrac{A, B, C \to}{A \land B, C \to}\,(\land L)}{A \land B \to \neg C}\,(\neg R)}{}}{A \land B \to A \land \neg C}\,(\land R) \quad \dfrac{A, B \to C}{A \land B \to C}\,(\land L)}{\dfrac{\Box(A \land B) \to \Box(A \land \neg C), \Box C}{\Box(A \land B) \to \Box(A \land \neg C) \lor \Box C}\,(\lor R)}\,(\Box) \qquad \begin{array}{ccc} \{A,B,C\} & & \{A,B\} \\ \vdots & & \\ \{\} & & \end{array}$$

これは反駁木と反駁木から抽出される反例を示している。右側の

$$\begin{array}{ccc} \{A,B,C\} & & \{A,B\} \\ \vdots & & \\ \{\} & & \end{array}$$

は次のようなクリプキモデルを示している。

$$\begin{array}{ccc} \{A,B,C\} & & \{A,B\} \\ & \nwarrow \quad \nearrow & \\ & \{\} & \end{array}$$

4.4.3 様相論理 KT

$\neg\Box(A \land \neg\Box A)$ すなわち`\lnot \Box (A \land \lnot \Box A )`に対しては次の証明図と`Provable`という出力を得る。

$$\dfrac{\dfrac{\blacksquare(A \land \neg\Box A), A \land \neg\Box A \to}{\Box(A \land \neg\Box A) \to}\,(T)}{\to \neg\Box(A \land \neg\Box A)}\,(\neg R)$$

$$\blacksquare(A \land \neg\Box A), A \land \neg\Box A \to \;\text{は}\; \dfrac{\dfrac{\dfrac{\dfrac{A, \neg\Box A \to A}{A \land \neg\Box A \to A}\,(\land L)}{\blacksquare(A \land \neg\Box A), A \to \Box A}\,(\Box)}{\blacksquare(A \land \neg\Box A), A, \neg\Box A \to}\,(\neg L)}{\blacksquare(A \land \neg\Box A), A \land \neg\Box A \to}\,(\land L)$$

そして $\neg\Box(A \land \neg\Box(A \land B))$ すなわち`\lnot \Box (A \land \lnot \Box (A \land B ))`の場合には次のような出力と伴に`Not provable. Write with the counter model.`が出力される。

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A, \neg\Box(A \land B) \to A \quad \dfrac{\dfrac{\dfrac{\to A \quad \to B}{\to A \land B}\,(\land R)}{A \to B, \Box(A \land B)}\,(\Box)}{A, \neg\Box(A \land B) \to B}\,(\neg L)}{A, \neg\Box(A \land B) \to A \land B}\,(\land R)}{A \land \neg\Box(A \land B) \to A \land B}\,(\land L)}{\blacksquare(A \land \neg\Box(A \land B)), A \to \Box(A \land B)}\,(\Box)}{\blacksquare(A \land \neg\Box(A \land B)), A, \neg\Box(A \land B) \to}\,(\neg L)}{\dfrac{\dfrac{\blacksquare(A \land \neg\Box(A \land B)), A \land \neg\Box(A \land B) \to}{\Box(A \land \neg\Box(A \land B)) \to}\,(T)}{\to \neg\Box(A \land \neg\Box(A \land B))}\,(\neg R)}\,(\land L) \qquad \begin{array}{c} \{\} \\ \vdots \\ \{A\} \\ \vdots \\ \{A\} \end{array}$$

これは反駁木と反駁木から抽出される反例を示している。右側が次のようなクリプキモデルを示している。

$$\begin{tikzcd} \{A\} \\ \{A\} \arrow[u] \\ \{A\} \arrow[u] \end{tikzcd}$$

4.4.4　様相論理 S4

S4 の証明反駁アルゴリズムはループを持つため [10] で導入された履歴 (history) を用いて効率的にループをチェックする。$\Box A \wedge \Box B \to \Box(\Box\Box A \wedge B)$ に対する証明図は次のようになる。

$$\begin{prooftree}
\AxiomC{$\Box A, \Box B \to \Box A\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\doubleLine
\RightLabel{$(\Box)_s$}
\UnaryInfC{$\blacksquare A, A, \blacksquare B, B \to \Box\Box A\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\blacksquare A, A, \Box B \to \Box\Box A\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box A, \Box B \to \Box\Box A\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\AxiomC{$\blacksquare A, A, \blacksquare B, B \to B\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\blacksquare A, A, \Box B \to B\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box A, \Box B \to B\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\RightLabel{$(\wedge R)$}
\BinaryInfC{$\Box A, \Box B \to \Box\Box A \wedge B\langle \Box B, \Box A | \Box(\Box\Box A \wedge B), \Box\Box A\rangle$}
\doubleLine
\RightLabel{$(\Box)_t$}
\UnaryInfC{$\blacksquare A, A, \blacksquare B, B \to \Box(\Box\Box A \wedge B)\langle \Box B, \Box A | \epsilon\rangle$}
\RightLabel{$(T)_t$}
\UnaryInfC{$\blacksquare A, A, \Box B \to \Box(\Box\Box A \wedge B)\langle \Box A | \epsilon\rangle$}
\RightLabel{$(T)_t$}
\UnaryInfC{$\Box A, \Box B \to \Box(\Box\Box A \wedge B)\langle \epsilon | \epsilon\rangle$}
\RightLabel{$(\wedge L)$}
\UnaryInfC{$\Box A \wedge \Box B \to \Box(\Box\Box A \wedge B)\langle \epsilon | \epsilon\rangle$}
\end{prooftree}$$

$\langle$ と $\rangle$ で囲まれた部分が履歴である。この履歴は証明可能な場合よりも、むしろ証明不可能な場合で有効的である。$\neg\Box\neg(\Box A \vee \Box\neg A)$ では、次のような出力とともに、`Not provable.  Write with the counter model.` と出力される。

$$\begin{prooftree}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_s$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A) \nrightarrow A, \Box A, \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\vee R)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A) \nrightarrow A, \Box A \vee \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\neg L)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A), \neg(\Box A \vee \Box\neg A) \nrightarrow A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box\neg(\Box A \vee \Box\neg A) \nrightarrow A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\AxiomC{}
\doubleLine
\RightLabel{$(init)_s$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A), A \nrightarrow \Box A, \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\vee R)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A), A \nrightarrow \Box A \vee \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\neg L)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A), \neg(\Box A \vee \Box\neg A), A \nrightarrow \langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{(T)}
\UnaryInfC{$\Box\neg(\Box A \vee \Box\neg A), A \nrightarrow \langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\RightLabel{$(\neg R)$}
\UnaryInfC{$\Box\neg(\Box A \vee \Box\neg A) \nrightarrow \neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \Box A, \Box\neg A\rangle$}
\doubleLine
\RightLabel{$(\Box)_t$}
\BinaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A) \nrightarrow \Box A, \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \epsilon\rangle$}
\RightLabel{$(\vee R)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A) \nrightarrow \Box A \vee \Box\neg A\langle \Box\neg(\Box A \vee \Box\neg A) | \epsilon\rangle$}
\RightLabel{$(\neg L)$}
\UnaryInfC{$\blacksquare\neg(\Box A \vee \Box\neg A), \neg(\Box A \vee \Box\neg A) \nrightarrow \langle \Box\neg(\Box A \vee \Box\neg A) | \epsilon\rangle$}
\RightLabel{$(T)_t$}
\UnaryInfC{$\Box\neg(\Box A \vee \Box\neg A) \nrightarrow \langle \epsilon | \epsilon\rangle$}
\RightLabel{$(\neg R)$}
\UnaryInfC{$\nrightarrow \neg\Box\neg(\Box A \vee \Box\neg A)\langle \epsilon | \epsilon\rangle$}
\end{prooftree}$$

$$\dfrac{\underline{\{\}}\qquad\underline{\{A\}}}{\{\}}$$

始式の右辺の $\Box A, \Box\neg A$ がともに非妥当な履歴に出現しているので、これ以上は規則が適用されず、これが反駁木となる。右側が次のようなクリプキモデルを示している。

$$\begin{tikzcd} \{\} & & \{A\} \\ & \{\} \arrow[ul, leftrightarrow] \arrow[ur, leftrightarrow] & \end{tikzcd}$$

4.5 教育用としての xpe

xpe は教育用としても有効である。一般的な定理自動証明システムとは異なり証明図の出力に配慮しているため、ユーザが実際に証明図を追い掛けることで、証明図の構成法を学び取ることができる。

2000 年に北陸先端科学技術大学院大学で開催されたサマースクールでは、xpe と用いて次のような演習を行った。

> exchange を含む直観主義部分構造論理 (古典論理を含む) は次の図ようなクラス構造を持ち、(1)〜(6) の部分に分かれる。このとき、それぞれの部分クラスに属する命題はどのようなものがあるか。

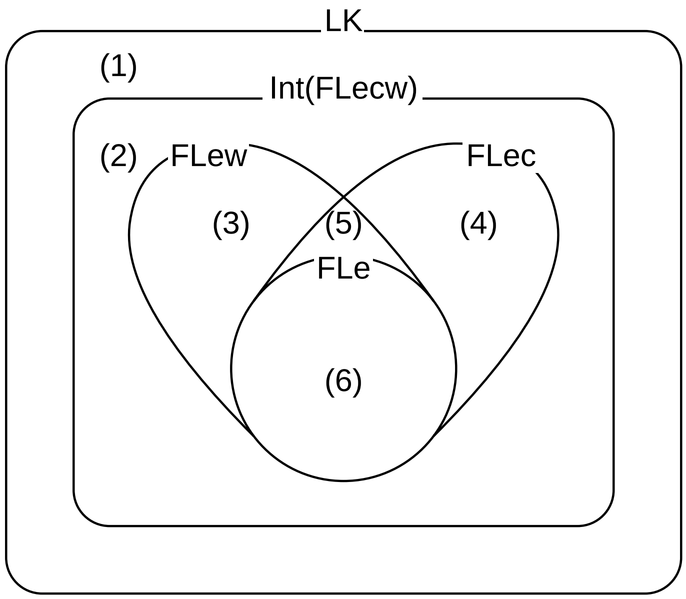


図 4.1: サマースクールの演習課題

ある論理式が証明不可能であることを紙の上で証明するためには、反例となるモデルを構成する必要がある。それは初学者に手に負えるものではない。しかしながら、定理自動証明システムを用いれば、単に論理式をプログラムに入力すればよく、簡単な作業となる。

サマースクールの受講者はこの演習を通して、部分構造論理への理解を深めることができた。

また、古典論理とラムダ項の型付けシステムではトレース実行ができるようになっている。これは与えられた論理式を分解していく課程を順を追って示すものである。例えば排中律 $A \vee \neg A$ では次のようになる。`next` ボタンを押す度に左上の状態からから右へ分解が進む様子が示される。トレース実行のような昨日は教育的な観点から非常に有用であると考えられる。

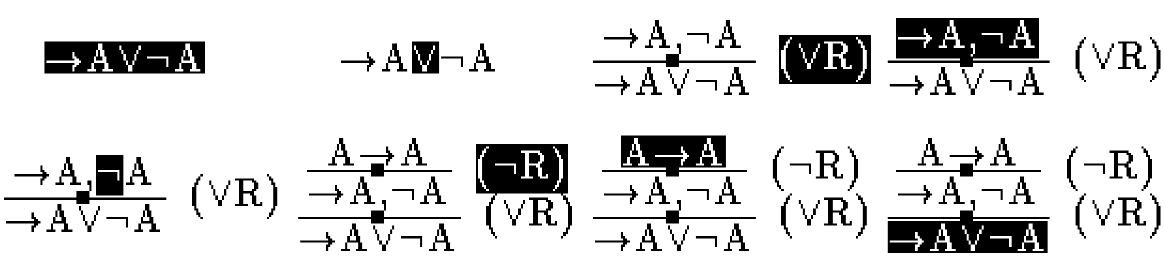


図 4.2: トレース実行の例

4.6 今後の課題

xpe の目指すところは、論理学の研究の際に役に立つようなツールである。証明図の作成ができるなど今のままでも充分に役に立つと考えられるが、今後はこの xpe をさらに発展させたいと考えている。それは次の 3 つの柱からなる。

- 証明反駁システムの実装
- モデルの妥当性の検証
- メタ定理自動証明システム

証明反駁システム　前章で示した通り、xpe では古典論理、様相論理 K,KT,S4 に対しては証明反駁システムが実装されている。

さらにまだこのようなアルゴリズムが知られていない体系、例えば S5 等においてアルゴリズムを見い出し、実装していきたいと考えている。

モデルの妥当性の検証　与えられた論理式が、特定のモデルで妥当であるか否かを検証することは機械的な作業である。モデルの妥当性は論理式に関して帰納的に定義されているので、分解した論理式をモデルに当てはめながら積み上げていけばよい。

しかしながら、非古典論理では大きな問題がある。それは、非古典論理ではモデルは一般には有向グラフとして定義されていることである。妥当性を調べるには、分解した論理式に対してグラフを一つ一つ丹念に追い掛けなければならず、もはや人間の手には負えなくなる。

これは当然計算機に実装すべきで課題である。有向グラフはノードのペアの集合で定義されるため、キャラクタベースでの実装も可能である。しかし、それではグラフの構造が隠れてしまい、使いにくいものになってしまう。グラフをグラフのまま扱え、その上で妥当性の検証ができる道具が必要である。そのようなシステムを構築したい。

メタ定理自動証明システム　いくつかの定理自動証明システムを実装した経験から分かってきたことがある。それは体系が異なっても共通化できる部分が多いということである。例えば入出力に関しては多少の差異はあるものの、一つのプログラムに吸収できる範囲である。では探索ルーチンではどうであろうか。これも共通化できる部分が多い。実際、様相論理の実装では全く同じプログラム用い、引数によって証明する体系を切り替えている。

では、さらに一般化できないであろうか。推論規則さえ与えれば、その体系の自動証明ができるようにできないであろうか。これを実現するのが、メタ定理自動証明システムである。ユーザーによって自由に拡張が行え、複雑な証明も肩代わりしてくれる道具となるであろう。

謝辞

浜野正浩助手には、セミナーを通して様々なアドバイスを頂きました。この場を借りて御礼を申し上げます。

Kowalski Tomasz 助手には、いつも英文の校正を行って頂き、本論文でも英文概要の作成に当たって、多大な助言を頂きました。ありがとうございます。

東条敏教授には論文題名と内容との繋がりを指摘して頂き、序章を大幅に書き直すことにより、全体としてまとまりがでてきました。ここに感謝の意を表します。

大堀淳教授には xpe の特に定理自動証明の部分に興味を持って頂き、web を通して公開することによりさらに発展してきました。ここに感謝の意を表します。

外山芳人教授には最短ラムダ戦略の研究をさせて頂き、また論文にまとめるにあたって関連研究と様々なアトバイスを頂きました。深く感謝いたします。

佐藤雅彦教授には、xpe の有用性、特に教育用としての有効性を指摘して頂きました。それでこのように xpe に関する記述を付録としてだけでなく、本文中に入れることができました。ありがとうございます。

石原哉助教授には、議論を通して論理学に対する考え方等を教わりました。また、xpe の学問としての有効性を指摘して頂き、リサーチレポートとしてまとめるに到りました。ここに感謝の意を表します。

そしてなにより、小野寛晰教授に心からの感謝を申し上げます。小野教授が北陸先端科学技術大学院大学に赴任してから 9 年間というもの、公私に渡ってお世話になりました。私の研究の全ては小野教授から授かったものであり、部分構造論理の自然演繹のテーマを与えて頂き、私がこの道に進むきっかけとなりました。無為な日々を過ごしたときもありましたが、そんなときでも私を励まし、そして導いて下さり、やっとここまで辿り着きました。研究者として一人前とはいかなくても、ここに半人前の姿を見せることができました。本当にありがとうございます。

そして、草稿の段階から校正を手伝ってくれ、問題点等を指摘して下さった高村博紀さんに感謝いたします。

関連図書

[1] H.P. Barendregt, *The Lambda Calculus, its Syntax and Semantics (second printing)*, North-Holland, 1985.

[2] Bayu Surarso and H. Ono, *Cut Elimination in Noncommutative Substructural Logics*, Reports on Mathematical Logic No.30 (1996), 13–29.

[3] Nick Benton, Gavin Bierman, Valeria de Paiva, and Martin Hyland, *A term calculus for intuitionistic linear logic*, Typed Lambda Calculi and Applications, M. Bezem and J.F. Groote eds., Lecture Notes in Computer Science 664 (1993), 75–90.

[4] D. van Dalen, *Logic and Structure (Second Edition)*, Springer-Verlag, 1989.

[5] R. Dyckhoff, *Contracton–free sequent calculi for intuitionistic logic*, The Journal of Symbolic Logic, vol. 57 (1992), no.3, 795–807.

[6] R. Dyckhoff and S. Negri, *Admissibility of structural rules contracton–free systems of intuitionistic logic*, The Journal of Symbolic Logic, vol. 65 (2000), no.4, 1499–1518.

[7] Jean-Yves Girard, *Linear Logic*, Theoretical Computer Science 50 (1987), 1–102.

[8] Jean-Yves Girard, Yves Lafont and Paul Taylor, *Proofs and Types*, Cambridge University Press, 1989.

[9] R. Goré, W. Heinle, A. Heuerding, G. Jäger and S. Schwendimanm, *An Introduction to Propositional Logics from a Computational Point of View*, to appear.

[10] Alain Heuerding, Michael Seyfried and Heinrich Zimmermann, *Efficient loop-check for backward proof search in some non-classical propositional logics*, In P. Miglioli, U. Moscato, D. Mundici, M. Ornaghi, editors, Tableaux 96, LNCS 1071, 210-225, 1996.

[11] Jörg Hudelmaier, *An $O(n\text{logn})$-Space Decision Procedure for Intuitionistic Propositional Logic* , J. Logic Computat., Vol 3 No.1, 1993, 63–75.

[12] J.Roger Hindley and Jonathan P.Seldin, *Introduction to Combinators and λ–Calculus*, Cambridge University Press, 1986.

[13] J.Roger Hindley, *BCK-combinators and linear λ-term have types*, Theoretical Computer Science 64 (1989), 97–105.

[14] E. Kiriyama and H. Ono, *The contraction rule and decision problems for logics without substractural rules*, Studia Logica 50 (1991), 299–319.

[15] J.J. Lévy, *Optimal Reductions in the Lambda Calculus*, To H. B. Curry, Essays in Combinatory Logic, Lambda Calculus and Formalism, J. P. Seldin and J. R. Hindley eds, Academic Press, 1980, 159-191.

[16] G. Mints, *Linear Lambda–Terms and Natural Deduction*, Studia Logica 60 (1998), 209–231.

[17] 毛利元彦, **二つの** *conjunction* **の概念に基づき拡張された** *linear* **λ** *-term* **の強正規性について**, 応用数学合同研究集会報告集, 30.1–30.2 (Dec 1994).

[18] Motohiko Mouri, *Natural Deduction Systems for Substructural Logics and Their Strong Normalizaion*, Second Workshop on Non–Standard Logics and Logical Aspects of Computer Science, Irkutsk, Russia, 53–54 (Jun 1995).

[19] 毛利元彦, *Natural Deduction Systems for Substructural Logics and Their Strong Normalizaion*, 書き換えシステムの理論とその応用, 数理解析研究所講究録 918, 196–205 (Oct 1995).

[20] 毛利元彦, *Optimal reduction strategies in developments and BCK–lambda terms*, Proceedings of the 32nd MLG meeting, 17–19 (Nov 1998).

[21] 毛利元彦, *Theorem Provers with Counter Models from xpe*, Proceedings of the 34th MLG meeting, 43–47, (Jan 2001).

[22] 毛利元彦, *Xpe with Theorem Provers*, JAIST Research Report, IS-RR-2001-021, (Aug 2001).

[23] Motohiko Mouri, *Theorem Provers with Counter Models and xpe*, Bulletin of the Section of Logic, vol 30 No.2, 2001, 79–86.

[24] Motohiko Mouri, *An efficient construction of counter–models for modal logic S4*, The International Workshop on Rewriting in Proof and Computation, (Oct 2001).

[25] H. Ono, *Semantics for Substructural Logics*, Substructural Logics, 259–291, Peter Schroeder-Heister and Kosta Došen eds., Oxford University Press, 1994.

[26] 小野寛晰, 情報科学における論理, 日本評論社, 1994.

[27] L. Pinto and R. Dyckhoff, *Loop–free construction of counter–models for intuitionistic propositional logic*, Symposia Gaussiana, Conf. A (Behara, Fritsch, and Lintz, editors), Walter de Gruyter & Co., Berlin, 1995, 225–232.

[28] Dag Prawitz, *Natural Deduction*, Almqvist & Wiksell, 1965.

[29] W.W. Tait, *Intensional interpretations of functionals of finite type*, Journal of Symbolic Logic 32 (1967), 198–212.

[30] M. Takahashi, *Parallel reductions in λ-calculus*, Journal of Symbolic Comput. Vol.7(1989), 113–123.

[31] 高橋正子, 計算論, 計算可能性とラムダ計算, 近代科学社, 1991.

[32] M. Takano, *Subformula property as a substitute for cut-elimination in modal propositonal logics*, Mathematica Japonica, vol 37(1992), 1129–1145.

[33] A.S. Troelstra, *Lectures on Linear Logic*, Center for the Study of Language and Information, Stanford University, 1992.

[34] A.S. Troelstra, *Natural deduction for intuitionistic linear logic*, Annals of Pure and Applied Logic 73 (1995), 79–108.

[35] R.C. de Vrijer, *A direct proof of the finite developments theorem*, Journal of Symbolic Logic 50 (1985), 339–343.

[36] R.C. de Vrijer, *Exactly estimating functionals and strong normalization*, Ph.D thesis, Universiteit van Amsterdam, 1987.

[37] Lincoln A. Wallen, *Automated Proof Search in Non-Classical Logics*, Efficient Matrix Proof Methods for Modal and Intuitionistic Logics MIT Press, 1990.

[38] H. Wang, *A survey of mathematical logic*, Studies in Logic and the Foundations of Mathematics, North-Holland, Amsterdam, 1963.

研究業績リスト

学術誌

- Motohiko Mouri, *Theorem Provers with Counter Models and xpe*, Bulletin of the Section of Logic, vol 30 No.2, 2001, 79–86.

国際会議

- Motohiko Mouri, *Natural Deduction Systems for Substructural Logics and Their Strong Normalizaion*, Second Workshop on Non–Standard Logics and Logical Aspects of Computer Science, Irkutsk, Russia, 53–54 (Jun 1995).

- 毛利元彦, *Natural Deduction Systems for Substructural Logics and Their Strong Normalizaion*, 数理解析研究所講究録 918, 196–205 (Oct 1995).

- Motohiko Mouri, *An efficient construction of counter–models for modal logic S4*, The International Workshop on Rewriting in Proof and Computation, (Oct 2001).

国内研究集会

- 毛利元彦, **二つの** *conjunction* **の概念に基づき拡張された** *linear* **λ**-*term* **の強正規性について**, 応用数学合同研究集会報告集, 30.1–30.2 (Dec 1994).

- 毛利元彦, *Optimal reduction strategies in developmes and BCK–lambda terms*,Proceedings of the 32nd MLG meeting, 17–19 (Nov 1998).

- 毛利元彦, *XPE with Theorem Provers*, SLACS2000, 京都大学, (Oct 2000).

- 毛利元彦, *Theorem Provers with Counter Models from xpe*, Proceedings of the 34th MLG meeting, 43–47, (Jan 2001).

研究報告書

- 毛利元彦, *Xpe with Theorem Provers*, JAIST Research Report, IS-RR-2001-021, (Aug 2001).

付 録 A　xpe manual

3 章で紹介した xpe のマニュアルを付録として添付する。これは TEX 用の証明図を描くソフトウェアであり、また、証明反駁アルゴリズムが実装され、いくつかの体系で証明図または反例を出力するようになっている。

A.1 はじめに

本ソフトウエアは無保証である。本ソフトウエアの使用におけるいかなる損害に対しても作者は責任を負わないものとする。

一次配布元は `ftp://logic.jaist.ac.jp/pub/program/XPE/` である。

A.1.1 xpe の概要

xpe(X window system Proof Editor) は、龍田真さん (`http://research.nii.ac.jp/~tatsuta/`) 作成の TEX のマクロ proof.sty と高い互換性を持つ X Window System 上の証明図作成支援ソフトである。TEX で証明図を描くには、多くの場合 proof.sty を用いて行われる。このマクロにより証明図をたやすく作成できるようになったが、その構造上の問題、それは木構造を持った証明図を、proof.sty ではグルーピング ({ . . . }) を用いて一次元で表現することに起因する、その問題により、可読性や運用性にやや難がある。要するに、証明図が複雑になって 10 段にもなると、もう何を書いてあるのかさっぱり分からなくしまう。proof.sty を使ったことがある人なら覚えがあると思う。そこで、X Window System で出来合いの証明図を見ながら証明図が書けるソフトを作ってみた。結構評判がいいので、ここに公開する。どのぐらい評判がいいかと言うと、ある人曰く、

「proof.sty の使い方はよく分からないけど、TEX で証明図が書ける。」

そうです。

その上、最近の xpe は定理自動証明さえこなすようになり、論理学者必帯のツールとなるべく日々精進している。その成果は [8] 等に示されている。

A.1.2 必要なもの

1. C コンパイラ。(gcc など)
2. X Window System の開発環境。PC-UNIX を使っている人は X-devel もインストールする。
3. X Athena Widget のの開発環境。PC-UNIX の場合は Xaw など。
4. bison。GNU の yacc 処理系。他の yacc 処理系でも動くと思うが未確認。
5. flex。GNU の lex 処理系。これは flex 特有の関数 (yyrestart) を使用しているために必要。

動作確認済みの OS

- Sun OS 4.1.4
- Solaris 5.7 (Sparc) (但し、Open Window 環境のみ。CDE では font 周りがうまく表示されなかった。動作させる方法があるのなら作者まで連絡されたし。あなたの氏名と伴に本マニュアルに記載するので。)
- Linux for Intel (Turbolinux 3.0, Turbolinux 4.0,Redhat 6.1)
- FreeBSD
- OpenBSD 2.5

A.1.3 インストール

インストールの流れ

1. ソースを展開
2. コンパイル
3. インストール
4. .eamcs 等の設定

root になれる場合のインストール

管理者であるか、自分の計算機の場合には root になれるので定理自動証明のインストールが簡単になる。ただし、/usr/X11R6 というディレクトリが存在しなければならない。`make install` では/usr/X11R6/bin と/usr/X11R6/lib/X11 に xpe をインストールしようとするからである。

1. xpe のソースを展開する。

   (例)

   `% zcat xpe-`*version*`.tar.gz | tar xvf -`

2. コンパイルしてみる。

   (例)

   `% cd xpe-`*version*`;make`

   コンパイルが通らないときには、次節 [root になれない場合のインストール] の [コンパイルしてみる] を参照。

3. root になる。

   (例)

   `% su`

   `Password:` (root のパスワード)

4. インストールしてみる。

   (例)

   `% make install`

5. `xpe.el` を emacs–lisp の path の通ったところにおく。

   (例)

   `% cp ./xpe.el ~/lib/emacs`

   または

   `% cp ./xpe /usr/share/emacs/site-lisp`

6. /usr/X11R6/lib/X11/fonts/xpe.fonts を fontpath に通す。

   (例)

   `% xset fp+ /usr/X11R6/lib/X11/fonts/xpe.fonts`

   (.xinitrc 等に書き込んでおくとよい)

7. `~/.emacs` に次の二行を追加

   `( autoload 'xpe "xpe" nil t )`

   `( autoload 'xpe-current-proof "xpe" nil t )`

   次の方法でも可能である。

   `% cat dot-emacs-xpe >> ~/.emacs`

8. テスト。以上でインストールが完了したので、うまく xpe が 立ち上がるかテストしてみる。csh 系なら、

   `% rehash;xpe -f test`

   bsh 系なら、

   `% hash -r;xpe -f test`

   として xpe の画面が表れればインストールは成功である。エラーが出るときには、次節 [テスト] を参照。

root になれない場合のインストール

root 権限を持たないときは、`make install` が使えないので定理自動証明のインストールが複雑になる。Makefile の `PREFIX` さえ設定すれば [root になれる場合] と同様にインストールできると思うが、未確認である。そのため、この節では定理自動証明のインストールには行わない。また、`configure` script があれば定理自動証明のインストールの問題は解決する。

1. xpe のソースを展開する。

   (例)

   `% zcat xpe-`*version*`.tar.gz | tar xvf -`

2. コンパイルしてみる。

   (例)

   `% cd xpe-`*version*`;make`

   もしコンパイルが通らないときには、Makefile なりソースなりを弄る。そんなに凝ったことはしていないので、大抵の UNIX でコンパイルが通るはず。だめなときは、周りの分かる人にでも聞いてみて。作者に連絡してもらってもよいが、環境が違うのでなにもできないことが多い。逆に configure スクリプトなりを作って一連の作業を自動化したときは、作者まで連絡されたし。xpe に取り込ませてもらうので。

3. `xpe` を path の通ったところにおく。

   (例)

   `% cp ./xpe ~/bin`

   または

   `% cp ./xpe /usr/local/bin`

4. `xpe.el` を emacs–lisp の path の通ったところにおく。

   (例)

   `% cp ./xpe.el ~/lib/emacs`

   または

   `% cp ./xpe /usr/share/emacs/site-lisp`

5. xpe のリソースファイル `Xpe` を X の resource path の通ったところにおく。

   (例)

   `% cp ./Xpe /usr/X11R6/lib/X11/app-defaults`

   `~/.Xdefaults` でリソース設定をしているなら、

   `% cat ./Xpe >> ~/.Xdefaults`

   `Xpe` をホームに置いてもよい。

   `% cp ./Xpe ~/.`

6. xpe.fonts を fontpath に通す。

   (例)

   `~/xpe-`*version* にソースを展開しているなら、

   `% xset fp+ ~/xpe-`*version*`/xpe.fonts`

   (.xinitrc 等に書き込んでおくとよい)

7. `~/.emacs` に次の二行を追加

   `( autoload 'xpe "xpe" nil t )`

   `( autoload 'xpe-current-proof "xpe" nil t )`

   次の方法でも可能である。

   `% cat dot-emacs-xpe >> ~/.emacs`

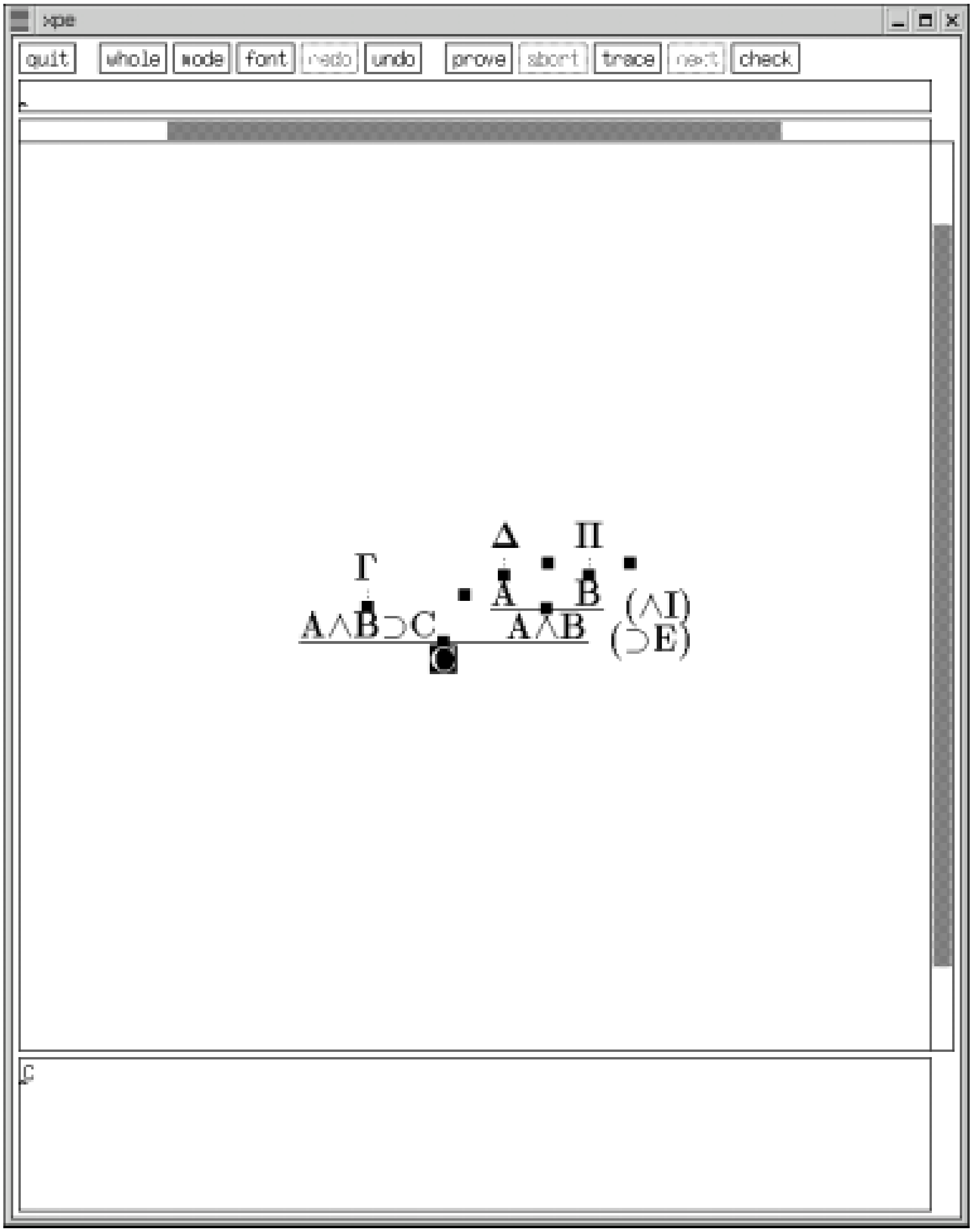
8. テスト。以上でインストールが完了したので、うまく xpe が 立ち上がるかテストしてみる。csh 系なら、

   `% rehash;xpe -f test`

   bsh 系なら、

```
% hash -r;xpe -f test
```

として次の画面が現れればインストールは成功である。




```
Cannot load font file 'hogehoge'
```

といわれるときは、font path の設定がうまくいってない。

```
% xset -q
```

として `xpe.fonts` に font path が通っているか確かめる。

```
Error:  Widget canvas has zero width and/or height
```

といわれるときは、リソースファイルの設定がうまくいっていない。

```
% cp ./Xpe  /.
```

としてリソースファイルをホームに置いて起動するか確かめる。

A.2 使ってみよう (チュートリアル)

A.2.1 シーケント (LK) の排中律 ($\to A \lor \neg A$) の証明図の作成

1. emacs(あるいは mule) を立ち上げる。

2. emacs で Control–SPACE としてリージョンの先頭を指定する。

3. `M-x xpe`

   新規に xpe を起動するときには、このように空のリージョンを指定して (つまりその場でおもむろに Control–SPACE として) `M-x xpe` とすればよい。もし既存の証明図を edit するときは、`\infer ...` をリージョンで囲んで `M-x xpe` とすれば、その証明図を取り込んで xpe が立ち上がる。

4. xpe の画面には  が表示されているであろう。ここに、`\to A \lor \lnot A` を入力し Return–key を押す。画面は  となったはずである。もし違う場合は入力が間違っている可能性がある。入力を確かめてみよ。

5.  の上部を左クリックする。

   画面: 

6. `\to A \lor \lnot A, A \lor \lnot A` を入力し Return を押す。

   画面: 

   このとき次のように入力の補完を行うこともできる。(A.3.3 参照)

   セルとは TEX のコマンドが解釈されて表示されている各部分であるが、このセルの上下左右をクリックして、空の入力で Return–key を押すと補完が行われる[1]。補完されるのは最も近いセルの内容である。今の場合、 の上部を左クリックして (画面: )、そのまま Return–key を押すと、テキスト部に`\to A \lor \lnot A` が補完される。

   また、テキストの編集では emacs ライクな編集を行える。今の場合、`A \lor \lnot A` を `C-k` で削除した後に `C-y` で 2 度ヤンクすると `A \lor \lnot A` がコピーされて編集が簡単になる。上部メニューの undo を使って試してみるとよいだろう。但し、`M-w` や `C-w` は Athena Widget の Text Widget の制限のため使えないので注意。

[1]もともとはバグであったが、便利なので手を加えて仕様にした。

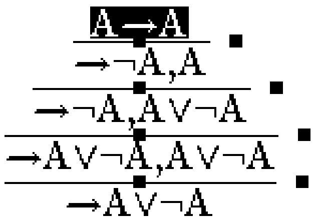
7. 以下同様に、

   `\to \lnot A, A \lor \lnot A`

   `\to \lnot A, A`

   `A \to A`

   を入力していくと、

   画面: 

   という LK の排中律の証明図が完成しているはずである。

8. 上部メニューの whole→write を選択

   これにより、emacs の ∗`xpe`∗ バッファーに

```
\infer{\to A \lor \lnot A}
  {\infer{\to A \lor \lnot A, A \lor \lnot A}
    {\infer{\to \lnot A, A \lor \lnot A}
      {\infer{\to \lnot A, A}
        {A \to A}
      }
    }
  }
```

   が出力される。あとはこれを適当な場所に移動させればよい。

A.2.2 自然演繹 (NK) の排中律 ($A \vee \neg A$) の証明図の作成

1. 上部メニューの whole→del を選択

   これにより、画面がクリアーされる。

2. `A` を入力

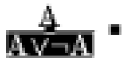
   画面: 

3. 

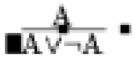
   の下方を左クリック。

   画面:

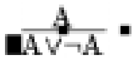
4. `A \lor \lnot A` を入力。

   画面: 

5. $A \lor \neg A$ の左側をクリック。

   画面: 

6. Return–key を押す。

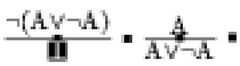
   画面: 

   テキスト部:`A \lor \lnot A`

7. `\lnot (A \lor \lnot A)` に変更。

   画面: 

8. $\neg(A \lor \neg A)$ の下方をクリックし、`\bot` を入力。

   画面: 

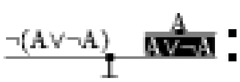
9. カット&ペーストを用いて、$\neg(A \lor \neg A)$ の右に $\dfrac{A}{A \lor \neg A}$ を 移動する。手順は以下のよう。

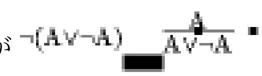
   (a) $A \lor \neg A$ を左クリック

   (b) 右クリックで `tree cut` を選択

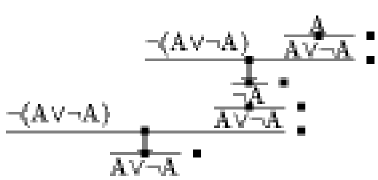
   (c) $\neg(A \lor \neg A)$ の右側を左クリック

   (d) 右クリックで `cell rep` (replace) を選択

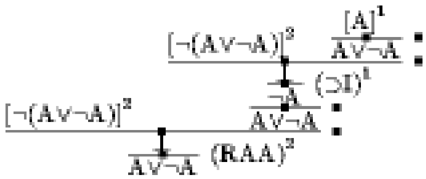
   画面: 

   また、次のように $\perp$ を挿入することも可能である。7. で画面が  となっているときに、$\neg(A \lor \neg A)$ の右下 (あるいは $A \lor \neg A$ の左下) を左クリックすると、画面が  となる。ここで`\bot` を入力すると、目的の証明図が得られる。

10. 以下カット&ペーストを駆使して証明図を完成させる。

    画面: 

11. 必要ならさらに規則等を書き加える。例えば次のよう。(A.3.3 節参照)

画面: 

12. 上部メニューの whole→write を選択。

    emacs に次のような内容が書き出される。

```
\infer[(RAA)^2]{A\lor \lnot A}
  {\infer{\bot}
    {[\lnot (A\lor \lnot A)]^2

    &\infer{A\lor \lnot A}
      {\infer[(\supset I)^1]{\lnot A}
        {\infer{\bot}
          {[\lnot (A\lor \lnot A)]^2

          &\infer{A\lor \lnot A}
            {[A]^1}
          }
        }
      }
    }
  }
```

A.2.3 ヒルベルト流の排中律 $(A \vee \neg A)$ の証明図

1. 上部メニューの whole→del を選択。

   これにより、画面がクリアーされる。

2. `A \lor \lnot A` と入力。以上。(だってそうなんだもの)

A.2.4 Typed-λ の $\lambda x.xy^{(a\to b)\to b}$ の証明図

1. 上部メニューの whole→del を選択。

   これにより、画面がクリアーされる。

2. `\lambda x.xy` と入力。

   画面: 

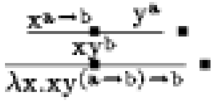
3. $\lambda x.xy$ の上部に `xy` を入力。

   画面: 

4. xy の上部に `x & y` を入力。

   画面: 

5. 各 λ 項に型を付ける。例えば次のよう。

   画面: 

6. 上部メニューの whole→write を選択。

   emacs に次のような内容が書き出される。

```
\infer{\lambda x.xy^{(a\to b) \to b}}
  {\infer{xy^b}
    {x^{a \to b}
    &y^a}
  }
```

A.2.5 古典命題論理での $(A \lor \neg A)$ の自動証明

1. 上部メニューの whole→del を選択。

   これにより、画面がクリアーされる。

2. `A \lor \lnot A` と入力。

3. 上部メニューの prover$rightarrow$classical を選択。

   画面: 

A.3 各機能について

A.3.1 起動

emacs からの起動

M–x xpe リージョンを読み込んで xpe を起動する。新規に証明図を書き始めるときは空のリージョン (すなわちその場でおもむろに Control–space を押す) を指定するとよい。

M–x xpe–current–proof emacs 上でカーソルが\infer または\deduce の上にあるとき、その証明図を一つ読み込んで xpe を起動する。[2]

[2]ただし、あまり検証されていないのでちゃんと動くかは不明。

コマンドラインからの起動

`% xpe` でコマンドラインから xpe を起動できる。xpe 特有のオプションは次のよう。

–f *filename* *filename* を読み込んで xpe を起動する。

–file *filename* 同上。

–emacs emacs から起動するときに用いるオプション。この場合、標準入力から証明図のデータを読み込み起動する。このように emacs から起動した場合、xpe は標準入力と標準出力を用いて emacs とやりとりしている。

A.3.2 上部メニュー

quit

xpe を終了させる。例え証明図に変更を加えていても強制的に終了する。但し、autosave ファイルが作られているので、ついうっかり終了させてしまったときには、autosave ファイル (A.3.7 節) を参照のこと。

whole

証明図が存在しているときに選択可能となる。(`whole`→`del` を行った後等) 証明図がないときには選択できない。

write 証明図を emacs に書き出す。コマンドラインから xpe を起動しているときには、標準出力に書き出される。

cut 証明図をセレクション (Windows でいうところのクリップボード) に貼り付けて消去する。消去した証明図をマウスを通して適当なところに貼り付けることができる。

copy 証明図をセレクションに貼り付ける。この証明図をマウスを通して適当なところに貼り付けることができる。

del 証明図を消去する。新しく証明図を書き始めるときなど。

rep(replace) セレクションの内容と証明図を入れ替える。別の証明図をエディットするときなど。

center 証明図を中央に移動する。(A.3.6 節参照)

mode

fix 普通の入力モード。入力を行った後、そのセル (A.3.3 節参照) に入力が留まる。

fill 入力を行った後、上にセルが無いときにはテキストを保持したまま入力が上部に移動する。シーケントの証明図を書くときに結構便利な入力モード。シーケントでは一般的に証明図の上部と下部の差異はわずかで、上部は下部のサブセットになっていることが多い。このモードでは下部をコピーして上部に持ち越すことで入力の労力を削減する。[3]

edit 黒点を左クッリックするとこで、ラベル (A.3.3 節参照) やセルの上下間に入力することができる。セルの上下間の黒点を左クリックしたときは、さらに右クリック (A.3.4 節参照) で証明図の属性 (`\infer`, `\infer*`, `\infer=`, `\deduce`) を変更できる。また、セルの左右を左クリックできるのも、このモードである。

view 黒点を消し左右を詰めることで、TEX の出力に近い表示をさせる。このモードでは、マウスでセルの左右を選択することはできない。A.3.5 節で説明されているように、‘`&`’ や ‘`&&`’ を用いてセルの挿入することは可能である。

[3] このモードは使っていると混乱してしまい、作者ですら使いこなせない。使いこなせれば便利そうなんだけど。

tex TEX のコマンドを解釈して表示する。解釈できない TEX のコマンドはそのまま表示する。どのようなコマンドが解釈できるかは、A.4.1 節を参照。

raw TEX のコマンドを解釈せずに表示する。実際にどのようなものが入力されているのか確かめたいときや、TEX では表示されないコマンド (空のグルーピング {} など) をエディットするときに用いる。公理を書くときには空のグルーピング {} を用いて、`\infer {A} {{}}` と書くが、この {} を消したいときなどに用いる。

font

フォントの大きさを変える。

small 小さいフォント

medium 中ぐらいのフォント

large 大きいフォント (そのまんまだなぁ:–)

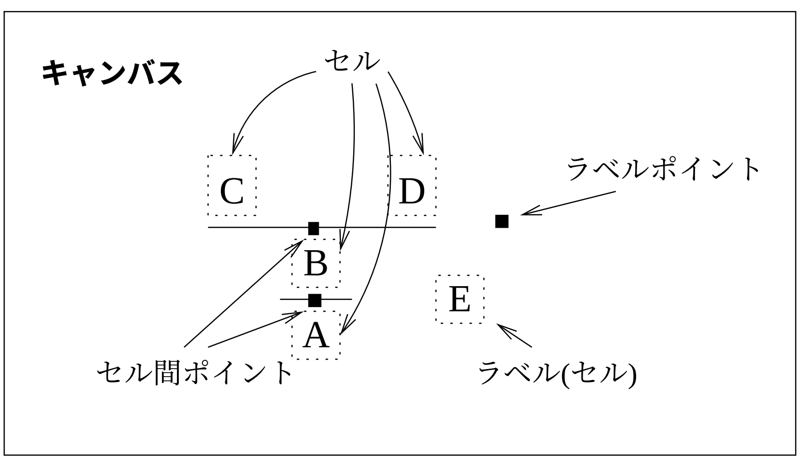
redo

いわゆるリドゥ。入力を行なったり、`cut` や `del` を行うとリドゥバッファはクリアされる。Netscape の next と back の関係に似ている。

undo

いわゆるアンドゥ。入力を行ったり、`cut` や `del` を行うとアンドゥスタックに証明図が積まれる。メモリーが確保される限り undo の回数は無制限である。(起こったことはないが) メモリーが足りなくなると、xpe は強制終了される。

A.3.3 入力



TEX のコマンドが解釈されて表示される矩形領域を**キャンバス**と呼ぶ。

セル ここを左クリックするとセルに変更を加えることができる。また、ここの上下左右を左クリックして色を反転させると、そこにセルを挿入することができる。ただし、左右に挿入できるのは、エディットモードのときのみである。また、ここを選択しているときに、右クリックメニューから証明図のタイプ (`infer` など) を変更できる。

セル間ポイント ここを左クリックすることで、セルの上下間にセルの挿入が可能になる。ここを選択しているときにも、右クリックメニューから証明図のタイプ (`infer` など) を変更できる。

ラベル (セル) ここを左クリックすると推論規則 (ラベル) に変更を加えることができる。ここを選択しているときにも、右クリックメニューから証明図のタイプ (`infer` など) を変更できる。

ラベルポイント ここを左クリックすることで、推論規則 (ラベル) を書き込める。

補完

セルの上下左右を左クリックして空の入力を行うと補完を行う。補完されるのは、最も近いセルの内容である。つまり、セルの上下左右を左クリックして二回 Return–key を押すと、クリックした場所にセルをコピーした証明図が作成される。

A.3.4 ポップアップメニュー (マウスによる証明図の編集)

キャンバスで右クリックで表れる。

paste いわゆるペースト。セレクションの内容をテキストのカーソル位置に貼り付ける。マウスの中ボタンと同様。

cell cut セルの内容をセレクションに貼り付け、そのセルを消去する。セルのテキストに変更を加えていても、Return–key を押さない限り修正以前の内容がセレクションにコピーされる。以下ポップアップメニューによるカット&ペーストでは、テキストへの変更は Return–key を押さない限りはカット&ペーストに反映されない。

cell copy セルのテキストをセレクションに貼り付ける。

cell del セルを削除する。

cell rep セルをセレクションの内容と置き換える。`cell del`⇒`paste`⇒Return–key と、ほぼ同じ動作である。

tree cut 左クリックによるセルを頂点とする証明木 (証明図) の内容をセレクションに貼り付け、その証明木を消去する。

tree copy そのセルを頂点とする証明木の内容をセレクションに貼り付ける。

tree del そのセルを頂点とする証明木を消去する。

tree rep そのセルを頂点とする証明木をセレクションの内容と置き換える。`tree del`⇒`paste`⇒Return–key と同じ動作である。

\infer セル間ポイントか上部にセルが存在するセルを左クリックすると選択可能となる。その証明木の属性を`\infer`に変更する。

\infer∗ その証明木の属性を`\infer∗`に変更する。

\infer= その証明木の属性を`\infer=`に変更する。

\deduce その証明木の属性を`\deduce`に変更する。

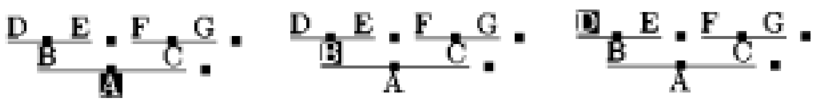
A.3.5 キーボードによる証明図の編集

セルカーソルの移動

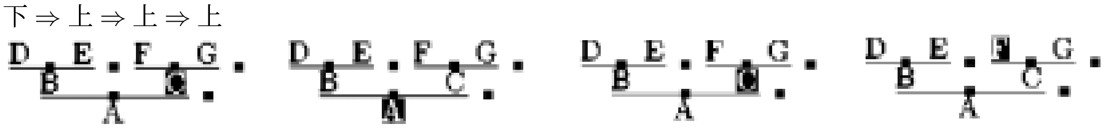
キャンバス上のどこかにセルカーソル (キャンバス上の反転している部分) があるとき、Shift+{↑,↓,←,→}–Key または Control+Shift+{p,n,l,r}–Key でそれぞれ上, 下, 左, 右に移動できる。上方向の移動では基本的に左上に移動する。また、emacs のようにある程度履歴を遡って移動する。

(例)

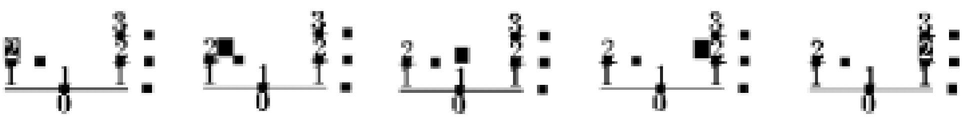
上 ⇒ 上 ⇒ 上



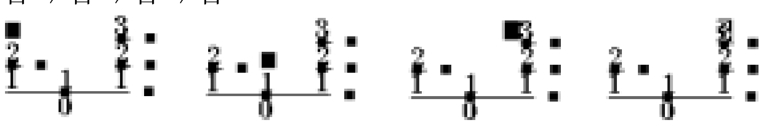
下 ⇒ 上 ⇒ 上 ⇒ 上



右 ⇒ 右 ⇒ 右 ⇒ 右 ⇒ 右

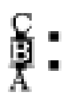


右 ⇒ 右 ⇒ 右 ⇒ 右



セルの挿入

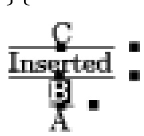
次の図のようにセルカーソルが `B` にあるときに、文字列 `Inserted` を上下左右へ挿入するときは、例えば次のようにする。



(例)

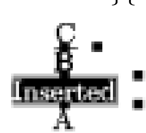
上へ挿入

`\infer{B}{Inserted}` を入力



下へ挿入

`\infer{Inserted}{B}` を入力

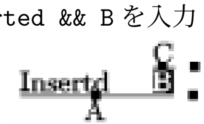


右へ挿入

`B & Inserted` を入力

左へ挿入

`Inserted && B` を入力



証明木の属性の変更

ファンクションキーの `F1`〜`F4` でセルカーソルの証明木の属性を変更できる。それぞれ、

F1 `\infer`

F2 `\infer*`

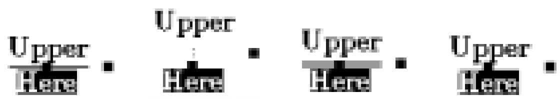
F3 `\infer=`

F4 `\deduce`

に対応している。

(例)

F1 **F2** **F3** **F4**



A.3.6 スクロール

スクロールバー

キャンバスの上と右に付随するスクロールバーで証明図の出力位置を変更できる。左クリックで画面サイズの 1/2 だけ左 (上) 方向に移動し、右クリックで 1/2 だけ右 (下) 方向に移動する。また、中クリックで任意の位置に移動できる。

センタリング

上部メニューの `whole→center` を選択すると、証明図を中央に移動する。また、セルカーソルがどこかにあるとき、Control–l でそのセルカーソルがキャンパスの中心になるようにセンタリングを行う。emacs の Control–l に似せてある。

A.3.7 オートセーブ

xpe はデフォルトで`~/.xpe_auto` にエディット中の証明図を書き出す。書き出すタイミングは、Return–key を押したり delete や cut を行うといった証明図への変更 5 回ごとである。xpe を不意に終了させてしまったときは、この file を参照のこと。環境変数 `XPEAUTOSAVEFILE` を設定するとその file に書き出す。なお、メモリーが足りなくなるなど xpe が異常終了した場合には、標準出力に編集中の証明図を書き出す[4]。

A.3.8 ユーザー定義マクロ

`~/.xperc` で設定できるようになっている。その構文は、

`\def マクロ 1{マクロ 2}`

である。TEX では { マクロ 2} に複数のマクロを記述できるが、xpe では一つだけであり、どちらかと言えば alias に近いものである。

(例)

[4]はずであるが、異常終了したときがないのでちゃんと機能するかは未確認。逆に考えると、xpe はかなり安定している。

```
\def\imp{\supset}
\def\fus{\ast}
\def\Dia{\Diamond}
```

A.3.9　定理自動証明の呼び出し

定理自動証明の詳しい説明は A.6 節を参照のこと。ここでは、簡単に呼び出し方を説明する。

定理自動証明

まずセルに論理式を入力する。そして上部メニューの `prove` を選択し、プルダウンメニューに表示される各定理自動証明を選択すると、論理式を定理自動証明にかけた結果が xpe に表示される。例は A.2.5 節を参照のこと。

定理自動証明のトレース実行

いくつかの定理自動証明は証明の課程をトレース (追跡) することができる。

定理自動証明と同様に、まずセルに論理式を入力する。そして上部メニューの `trace` を選択し、プルダウンメニューに表示される各定理自動証明を選択すると、トレースが始まる。`next` を押す度に次の課程が表示され、証明が終了するまで続く。

定理自動証明が内部でどのような処理を行っているのかを理解する手掛りになるであろう。教育的にも便利かも知れない。

A.4　TEX(proof.sty) との非互換な部分

xpe は TEX と高い互換性を持つが、いくつかの非互換性も存在する。

A.4.1　xpe が解釈可能なマクロ

xpe が解釈可能なマクロは、plain.tex(`/usr/share/texmf/tex/plain/base/plain.tex` など) に書かれた 0 引数のマクロである。解釈できないマクロはそのまま表示される。

フォントのマクロ

xpe では単に何も表示しないマクロである。一覧を次に示す。

`\rm`　`\em`　`\bf`　`\boldmath`　`\it`　`\sl`　`\sf`　`\sc`　`\tt`

フォントサイズのマクロ

これも xpe では何も表示しないマクロである。一覧を次に示す。

`\tiny`	`\scriptsize`	`\footnotesize`	`\small`
`\normalsize`	`\large`	`\Large`	`\LARGE`
`\huge`	`\Huge`		

一般的なマクロ

一覧は次のようになっている。

\%	%	\&	&	\#	#	\$	$\$$
\ss	ß	\ae	æ	\oe	œ	\o	ø
\AE	Æ	\OE	Œ	\O	Ø	\i	ı
\j	ȷ	\braceld	$\braceld$	\bracerd	$\bracerd$	\bracelu	$\bracelu$
\braceru	$\braceru$	\alpha	α	\beta	β	\gamma	γ
\delta	δ	\epsilon	ϵ	\zeta	ζ	\eta	η
\theta	θ	\iota	ι	\kappa	κ	\lambda	λ
\mu	μ	\nu	ν	\xi	ξ	\pi	π
\rho	ρ	\sigma	σ	\tau	τ	\upsilon	υ
\phi	ϕ	\chi	χ	\psi	ψ	\omega	ω
\varepsilon	ε	\vartheta	ϑ	\varpi	ϖ	\varrho	ϱ
\varsigma	ς	\varphi	φ	\Gamma	Γ	\Delta	Δ
\Theta	Θ	\Lambda	Λ	\Xi	Ξ	\Pi	Π
\Sigma	Σ	\Upsilon	Υ	\Phi	Φ	\Psi	Ψ
\Omega	Ω	\aleph	$\aleph$	\imath	$\imath$	\jmath	$\jmath$
\ell	ℓ	\wp	$\wp$	\Re	$\Re$	\Im	$\Im$
\partial	∂	\infty	∞	\prime	$\prime$	\emptyset	$\emptyset$
\nabla	∇	\top	$\top$	\bot	$\bot$	\triangle	$\triangle$
\forall	$\forall$	\exists	$\exists$	\neg	$\neg$	\lnot	$\lnot$
\flat	$\flat$	\natural	$\natural$	\sharp	$\sharp$	\clubsuit	$\clubsuit$
\diamondsuit	$\diamondsuit$	\heartsuit	$\heartsuit$	\spadesuit	$\spadesuit$	\coprod	$\coprod$
\bigvee	$\bigvee$	\bigwedge	$\bigwedge$	\biguplus	$\biguplus$	\bigcap	$\bigcap$
\bigcup	$\bigcup$	\intop	$\intop$	\prod	$\prod$	\sum	$\sum$
\bigotimes	$\bigotimes$	\bigoplus	$\bigoplus$	\bigodot	$\bigodot$	\ointop	$\ointop$
\bigsqcup	$\bigsqcup$	\smallint	$\smallint$	\triangleleft	$\triangleleft$	\triangleright	$\triangleright$
\bigtriangleup	$\bigtriangleup$	\bigtriangledown	$\bigtriangledown$	\wedge	$\wedge$	\land	$\land$
\vee	$\vee$	\lor	$\lor$	\cap	$\cap$	\cup	$\cup$
\ddagger	$\ddagger$	\dagger	$\dagger$	\sqcap	$\sqcap$	\sqcup	$\sqcup$
\uplus	$\uplus$	\amalg	$\amalg$	\diamond	$\diamond$	\bullet	$\bullet$
\wr	$\wr$	\div	$\div$	\odot	$\odot$	\oslash	$\oslash$
\otimes	$\otimes$	\ominus	$\ominus$	\oplus	$\oplus$	\mp	$\mp$
\pm	$\pm$	\circ	$\circ$	\bigcirc	$\bigcirc$	\setminus	$\setminus$
\cdot	$\cdot$	\ast	$\ast$	\times	$\times$	\star	$\star$
\propto	$\propto$	\sqsubseteq	$\sqsubseteq$	\sqsupseteq	$\sqsupseteq$	\parallel	$\parallel$
\mid	$\mid$	\dashv	$\dashv$	\vdash	$\vdash$	\nearrow	$\nearrow$
\searrow	$\searrow$	\nwarrow	$\nwarrow$	\swarrow	$\swarrow$	\Leftrightarrow	$\Leftrightarrow$
\Leftarrow	$\Leftarrow$	\Rightarrow	$\Rightarrow$	\leq	$\leq$	\le	$\le$
\geq	$\geq$	\ge	$\ge$	\succ	$\succ$	\prec	$\prec$
\approx	$\approx$	\succeq	$\succeq$	\preceq	$\preceq$	\supset	$\supset$
\subset	$\subset$	\supseteq	$\supseteq$	\subseteq	$\subseteq$	\in	$\in$
\ni	$\ni$	\owns	$\owns$	\gg	$\gg$	\ll	$\ll$
\not	$/$	\leftrightarrow	$\leftrightarrow$	\leftarrow	$\leftarrow$	\gets	$\gets$
\rightarrow	$\rightarrow$	\to	$\to$	\mapstochar	$\mapstochar$	\sim	$\sim$
\simeq	$\simeq$	\perp	$\perp$	\equiv	$\equiv$	\asymp	$\asymp$
\smile	$\smile$	\frown	$\frown$	\leftharpoonup	$\leftharpoonup$	\leftharpoondown	$\leftharpoondown$
\rightharpoonup	$\rightharpoonup$	\rightharpoondown	$\rightharpoondown$	\lhook	$\lhook$	\rhook	$\rhook$
\ldotp	$\ldotp$	\cdotp	$\cdotp$	\colon	$\colon$		

amssym より

$\boxdot$	\boxdot	$\boxplus$	\boxplus	$\boxtimes$	\boxtimes	$\square$	\square
$\blacksquare$	\blacksquare	$\centerdot$	\centerdot	$\lozenge$	\lozenge	$\blacklozenge$	\blacklozenge
$\circlearrowright$	\circlearrowright	$\circlearrowleft$	\circlearrowleft	$\rightleftharpoons$	\rightleftharpoons	$\leftrightharpoons$	\leftrightharpoons
$\boxminus$	\boxminus	$\Vdash$	\Vdash	$\Vvdash$	\Vvdash	$\vDash$	\vDash
$\twoheadrightarrow$	\twoheadrightarrow	$\twoheadleftarrow$	\twoheadleftarrow	$\leftleftarrows$	\leftleftarrows	$\rightrightarrows$	\rightrightarrows
$\upuparrows$	\upuparrows	$\downdownarrows$	\downdownarrows	$\upharpoonright$	\upharpoonright	$\restriction$	\restriction
$\downharpoonright$	\downharpoonright	$\upharpoonleft$	\upharpoonleft	$\downharpoonleft$	\downharpoonleft	$\rightarrowtail$	\rightarrowtail
$\leftarrowtail$	\leftarrowtail	$\leftrightarrows$	\leftrightarrows	$\rightleftarrows$	\rightleftarrows	$\Lsh$	\Lsh
$\Rsh$	\Rsh	$\rightsquigarrow$	\rightsquigarrow	$\leftrightsquigarrow$	\leftrightsquigarrow	$\looparrowleft$	\looparrowleft
$\looparrowright$	\looparrowright	$\circeq$	\circeq	$\succsim$	\succsim	$\gtrsim$	\gtrsim
$\gtrapprox$	\gtrapprox	$\multimap$	\multimap	$\therefore$	\therefore	$\because$	\because
$\doteqdot$	\doteqdot	$\Doteq$	\Doteq	$\triangleq$	\triangleq	$\precsim$	\precsim
$\lesssim$	\lesssim	$\lessapprox$	\lessapprox	$\eqslantless$	\eqslantless	$\eqslantgtr$	\eqslantgtr
$\curlyeqprec$	\curlyeqprec	$\curlyeqsucc$	\curlyeqsucc	$\preccurlyeq$	\preccurlyeq	$\leqq$	\leqq
$\leqslant$	\leqslant	$\lessgtr$	\lessgtr	$\backprime$	\backprime	$\risingdotseq$	\risingdotseq
$\fallingdotseq$	\fallingdotseq	$\succcurlyeq$	\succcurlyeq	$\geqq$	\geqq	$\geqslant$	\geqslant
$\gtrless$	\gtrless	$\sqsubset$	\sqsubset	$\sqsupset$	\sqsupset	$\vartriangleright$	\vartriangleright
$\vartriangleleft$	\vartriangleleft	$\trianglerighteq$	\trianglerighteq	$\trianglelefteq$	\trianglelefteq	$\bigstar$	\bigstar
$\between$	\between	$\blacktriangledown$	\blacktriangledown	$\blacktriangleright$	\blacktriangleright	$\blacktriangleleft$	\blacktriangleleft
$\vartriangle$	\vartriangle	$\blacktriangle$	\blacktriangle	$\triangledown$	\triangledown	$\eqcirc$	\eqcirc
$\lesseqgtr$	\lesseqgtr	$\gtreqless$	\gtreqless	$\lesseqqgtr$	\lesseqqgtr	$\gtreqqless$	\gtreqqless
$\Rrightarrow$	\Rrightarrow	$\Lleftarrow$	\Lleftarrow	$\veebar$	\veebar	$\barwedge$	\barwedge
$\doublebarwedge$	\doublebarwedge	$\angle$	\angle	$\measuredangle$	\measuredangle	$\sphericalangle$	\sphericalangle
$\varpropto$	\varpropto	$\smallsmile$	\smallsmile	$\smallfrown$	\smallfrown	$\Subset$	\Subset
$\Supset$	\Supset	$\Cup$	\Cup	$\doublecup$	\doublecup	$\Cap$	\Cap
$\doublecap$	\doublecap	$\curlywedge$	\curlywedge	$\curlyvee$	\curlyvee	$\leftthreetimes$	\leftthreetimes
$\rightthreetimes$	\rightthreetimes	$\subseteqq$	\subseteqq	$\supseteqq$	\supseteqq	$\bumpeq$	\bumpeq
$\Bumpeq$	\Bumpeq	$\lll$	\lll	$\llless$	\llless	$\ggg$	\ggg
$\gggtr$	\gggtr	$\circledS$	\circledS	$\pitchfork$	\pitchfork	$\dotplus$	\dotplus
$\backsim$	\backsim	$\backsimeq$	\backsimeq	$\complement$	\complement	$\intercal$	\intercal
$\circledcirc$	\circledcirc	$\circledast$	\circledast	$\circleddash$	\circleddash	$\lvertneqq$	\lvertneqq
$\gvertneqq$	\gvertneqq	$\nleq$	\nleq	$\ngeq$	\ngeq	$\nless$	\nless
$\ngtr$	\ngtr	$\nprec$	\nprec	$\nsucc$	\nsucc	$\lneqq$	\lneqq
$\gneqq$	\gneqq	$\nleqslant$	\nleqslant	$\ngeqslant$	\ngeqslant	$\lneq$	\lneq
$\gneq$	\gneq	$\npreceq$	\npreceq	$\nsucceq$	\nsucceq	$\precnsim$	\precnsim
$\succnsim$	\succnsim	$\lnsim$	\lnsim	$\gnsim$	\gnsim	$\nleqq$	\nleqq
$\ngeqq$	\ngeqq	$\precneqq$	\precneqq	$\succneqq$	\succneqq	$\precnapprox$	\precnapprox
$\succnapprox$	\succnapprox	$\lnapprox$	\lnapprox	$\gnapprox$	\gnapprox	$\nsim$	\nsim
$\ncong$	\ncong	$\diagup$	\diagup	$\diagdown$	\diagdown	$\varsubsetneq$	\varsubsetneq
$\varsupsetneq$	\varsupsetneq	$\nsubseteqq$	\nsubseteqq	$\nsupseteqq$	\nsupseteqq	$\subsetneqq$	\subsetneqq
$\supsetneqq$	\supsetneqq	$\varsubsetneqq$	\varsubsetneqq	$\varsupsetneqq$	\varsupsetneqq	$\subsetneq$	\subsetneq
$\supsetneq$	\supsetneq	$\nsubseteq$	\nsubseteq	$\nsupseteq$	\nsupseteq	$\nparallel$	\nparallel
$\nmid$	\nmid	$\nshortmid$	\nshortmid	$\nshortparallel$	\nshortparallel	$\nvdash$	\nvdash
$\nVdash$	\nVdash	$\nvDash$	\nvDash	$\nVDash$	\nVDash	$\ntrianglerighteq$	\ntrianglerighteq
$\ntrianglelefteq$	\ntrianglelefteq	$\ntriangleleft$	\ntriangleleft	$\ntriangleright$	\ntriangleright	$\nleftarrow$	\nleftarrow
$\nrightarrow$	\nrightarrow	$\nLeftarrow$	\nLeftarrow	$\nRightarrow$	\nRightarrow	$\nLeftrightarrow$	\nLeftrightarrow
$\nleftrightarrow$	\nleftrightarrow	$\divideontimes$	\divideontimes	$\varnothing$	\varnothing	$\nexists$	\nexists
$\Finv$	\Finv	$\Game$	\Game	$\mho$	\mho	$\eth$	\eth
$\eqsim$	\eqsim	$\beth$	\beth	$\gimel$	\gimel	$\daleth$	\daleth
$\lessdot$	\lessdot	$\gtrdot$	\gtrdot	$\ltimes$	\ltimes	$\rtimes$	\rtimes
$\shortmid$	\shortmid	$\shortparallel$	\shortparallel	$\smallsetminus$	\smallsetminus	$\thicksim$	\thicksim
$\thickapprox$	\thickapprox	$\approxeq$	\approxeq	$\succapprox$	\succapprox	$\precapprox$	\precapprox
$\curvearrowleft$	\curvearrowleft	$\curvearrowright$	\curvearrowright	$\digamma$	\digamma	$\varkappa$	\varkappa
$\Bbbk$	\Bbbk	$\hslash$	\hslash	$\hbar$	\hbar	$\backepsilon$	\backepsilon

latexsym より

$\mho$	`\mho`	$\Join$	`\Join`	$\Box$	`\Box`	$\Diamond$	`\Diamond`
$\leadsto$	`\leadsto`	$\sqsubset$	`\sqsubset`	$\sqsupset$	`\sqsupset`	$\lhd$	`\lhd`
$\unlhd$	`\unlhd`	$\rhd$	`\rhd`	$\unrhd$	`\unrhd`		

long 系

$\hookrightarrow$	`\hookrightarrow`	$\hookleftarrow$	`\hookleftarrow`	$\bowtie$	`\bowtie`	$\models$	`\models`
$\Longrightarrow$	`\Longrightarrow`	$\longrightarrow$	`\longrightarrow`	$\longleftarrow$	`\longleftarrow`	$\Longleftarrow$	`\Longleftarrow`
$\mapsto$	`\mapsto`	$\longleftrightarrow$	`\longleftrightarrow`	$\Longleftrightarrow$	`\Longleftrightarrow`	$\iff$	`\iff`
$\longmapsto$	`\longmapsto`						

A.4.2　スペースの取り扱い

xpe の字句解析器ではトークンにスペースを付けて切り出している。よって先頭のスペースはどんなトークンにも付随していなため自動的に落とされる。(A.5.1 参照)

A.4.3　上付き文字 ‘^’ と下付き文字 ‘_’

TEX では `A^B^C` はエラーとなるが、xpe では ‘^’ や ‘_’ は左結合と解釈されるため、`{A^B}^C` と同値になる。これは単に手抜きである。将来的には TEX の解釈に従う可能性がある。

A.4.4　`\infer` の引数

xpe に証明図を読み込ませるとき`\infer` や`\deduce` の引数はグルーピング {} で囲まなければならない。例えば `\infer A B` はエラーとなる。これは、{} を引数の識別子として扱っているからである。この場合は、`\infer {A} {B}` としてから読み込ませるか、読み込ませてから`\infer {A} {B}` と変更すればよい。

尚、xpe では引数を全てグルーピングしてから出力するので、xpe で書いたものは xpe で読み込める。

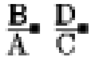
A.4.5　デリミター ‘&’

TEX では、
`\infer{A}{B \infer{C}{D}}` と
`\infer{A}{B & \infer{C}{D}}` では

$\dfrac{B\dfrac{D}{C}}{A} \qquad \dfrac{B \quad \dfrac{D}{C}}{A}$

のように若干出力が異なるが、xpe では全て後者のように間に ‘&’ が入っているとして扱われる。もし、どうしても ‘&’ を取り除きたいときは、出力後にエディタで ‘&’ を取り除く必要がある。

また、



のように複数の証明図をエディットすると出力時に

```
\infer{A}
  {B}
&\infer{C}
  {D}
```

'&' が入ってしまう[5] 。TEX にかけるときはこれも取り除く必要がある。

A.4.6 ラベル付きの \infer∗ と \deduce

proof.sty では \infer ∗ [Label]{Lower}{Upper} と \deduce[Label]{Lower}{Upper} はそれぞれ

$$\begin{array}{c} Upper \\ \vdots\ Label \\ Lower \end{array} \qquad \begin{array}{c} Upper \\ Label \\ Lower \end{array}$$

と Label が内側に入りこむように表示されるが、xpe ではそれぞれ

$$\begin{array}{c} Upper \\ \vdots \\ Lower \end{array}\quad Label \qquad \begin{array}{c} Upper \\ Lower \end{array} Label$$

と外側に押し出されるように表示される。これも手抜きである (自爆)。

A.5 xpe の内部構造

A.5.1 字句解析器

flex(GNU の lex 処理系) を用いて書かれている。{sp}≡[\t\n\r]∗ としておよそ次のようにスペース付きで字句を切り出している。

```
"\\infer"{sp}"*"{sp}    :INFERS
"\\infer"{sp}"="{sp}    :INFERE
"\\infer"{sp}           :INFER
"\\deduce"{sp}          :DEDUCE
"\\"[a-zA-Z]+{sp}       :MACRO
"\\".{sp}               :MACRO
"["{sp}                 :LBRACE
"]"{sp}                 :RBRACE
"{"{sp}                 :LGROUP
"}"{sp}                 :RGROUP
"&&"{sp}                :DAND
"&"{sp}                 :AND
"^"{sp}                 :HAT
"_"{sp}                 :BAR
[ \t\n\r]+              :
<<EOF>>                 :END
```

[5]不必要な '&' は出力されない仕様にもできるのだが、そうすると必要な '&' が出力されたときに、ユーザーがなぜエラーとなるのか分からなくなりそうなので、必ず '&' を出力するようにしてある。

```
\000                    :END
.{sp}                   :LETTER
```

A.5.2 構文解析器

bison(GNU の yacc 処理系) を用いて書かれている。およそ次のような BNF になっている。

```
start : upper END
      | END

tree    : inf
        | infs
        | infe
        | ded
        | LGROUP tree RGROUP

inf     : INFER   LBRACE label RBRACE  LGROUP lower RGROUP  LGROUP upper RGROUP
        | INFER                        LGROUP lower RGROUP  LGROUP upper RGROUP

infs    : INFERS  LBRACE label RBRACE  LGROUP lower RGROUP  LGROUP upper RGROUP
        | INFERS                       LGROUP lower RGROUP  LGROUP upper RGROUP

infe    : INFERE  LBRACE label RBRACE  LGROUP lower RGROUP  LGROUP upper RGROUP
        | INFERE                       LGROUP lower RGROUP  LGROUP upper RGROUP

ded     : DEDUCE  LBRACE label RBRACE  LGROUP lower RGROUP  LGROUP upper RGROUP
        | DEDUCE                       LGROUP lower RGROUP  LGROUP upper RGROUP

label   : laterm
        | laterm label

laterm  : laword
        | LGROUP RGROUP
        | LGROUP terms RGROUP
        | laterm HAT laterm
        | laterm BAR laterm

laword  : LETTER
        | MACRO
        | LBRACE

lower   : terms

terms   : term
```

```
        | terms term

term    : word
        | LGROUP RGROUP
        | LGROUP terms RGROUP
        | term HAT term
        | term BAR term

word    : LETTER
        | MACRO
        | LBRACE
        | RBRACE

upper   : tree
        | tree upper
        | tree AND upper
        | tree DAND upper
        | terms
        | terms AND upper
        | terms DAND upper
        | terms tree
        | terms tree upper
        | terms tree AND upper
        | terms tree DAND upper
```

特徴的なのは、`terms` では左解析木、`upper` では右解析木を用いて構文解析を行っていることである。これによりグルーピングの中が tree なのか、あるいは terms なのかを yacc が判断できるようになる。

A.6 定理自動証明

本章では xpe 上で動く定理自動証明を紹介する。ここではその推論規則の紹介のみに留める。詳細は、[7], [10], [11] 等を参照のこと。

A.6.1 論理記号

- 命題変数 $\cdots$ アルファベット, \ アルファベット列
- $\neg \cdots$ `\neg`, `\lnot`
- $\supset \cdots$ `\supset`, `\imp`
- $\wedge \cdots$ `\land`
- $\vee \cdots$ `\lor`
- $\ast \cdots$ `\ast`, `\fus`

- $\forall \cdots$ `\forall`
- $\exists \cdots$ `\exists`
- $\rightarrow \cdots$ `\to`, `\vdash`, `\rightarrow`

(例)

$A \supset \neg A \cdots$ `A \supset \lnot A`

$\Gamma \rightarrow A \supset B \cdots$ `\Gamma \to A \supset B`

$A \wedge (B \vee C) \rightarrow (A \wedge B) \vee (A \wedge C) \cdots$ `A \land(B \lor C)\to(A \land B) \lor (A \land C)`

$\forall x.A(x) \rightarrow \exists x.A(x) \cdots$ `\forall x.A(x)\hto\exists x.A(x)`

A.6.2 論理体系

classical,classical-trace(古典命題論理)

シーケントの両辺は multi set(多重集合) と考える。set(集合) と考えてもよいが、multi set の方が出力された証明図の可読性が高いため、そのようにしてある。classical-trace ではこの体系のトレース実行をする。

$$\Gamma, A \rightarrow \Delta, A$$

$$\frac{\Gamma \rightarrow \Delta, A \quad \Gamma, B \rightarrow \Delta}{\Gamma, A \supset B \rightarrow \Delta}\ (\supset L) \qquad \frac{\Gamma, A \rightarrow \Delta, B}{\Gamma \rightarrow \Delta, A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \rightarrow \Delta}{\Gamma, A \wedge B \rightarrow \Delta}\ (\wedge L) \qquad \frac{\Gamma \rightarrow \Delta, A \quad \Gamma \rightarrow \Delta, B}{\Gamma \rightarrow \Delta, A \wedge B}\ (\wedge R)$$

$$\frac{\Gamma, A \rightarrow \Delta \quad \Gamma, B \rightarrow \Delta}{\Gamma, A \vee B \rightarrow \Delta}\ (\vee L) \qquad \frac{\Gamma \rightarrow \Delta, A, B}{\Gamma \rightarrow \Delta, A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \rightarrow \Delta}{\Gamma \rightarrow \Delta, \neg A}\ (\neg R) \qquad \frac{\Gamma, A \rightarrow \Delta}{\Gamma \rightarrow \Delta, \neg A}\ (\neg R)$$

fle(直観主義部分構造論理)

シーケントの左辺は multi set と考える。

$$A \to A$$

$$\frac{\Gamma \to A \quad \Delta, B \to C}{\Gamma, \Delta, A \supset B \to C}\ (\supset L) \qquad \frac{\Gamma, A \to B}{\Gamma \to A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \to C}{\Gamma, A * B \to C}\ (*L) \qquad \frac{\Gamma \to A \quad \Delta \to B}{\Gamma, \Delta \to A * B}\ (*R)$$

$$\frac{\Gamma, A \to C}{\Gamma, A \wedge B \to C}\ (\wedge L) \qquad \frac{\Gamma, B \to C}{\Gamma, A \wedge B \to C}\ (\wedge L)$$

$$\frac{\Gamma \to A \quad \Gamma \to B}{\Gamma \to A \wedge B}\ (\wedge R) \qquad \frac{\Gamma, A \to C \quad \Gamma, B \to C}{\Gamma, A \vee B \to C}\ (\vee L)$$

$$\frac{\Gamma \to A}{\Gamma \to A \vee B}\ (\vee R) \qquad \frac{\Gamma \to B}{\Gamma \to A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \to}{\Gamma \to \neg A}\ (\neg R) \qquad \frac{\Gamma, A \to}{\Gamma \to \neg A}\ (\neg R)$$

flew(直観主義部分構造論理)

fle の始式を次のように変更し、さらに Weakening 規則を加える。

$$\Gamma, A \to A$$

$$\frac{\Gamma \to}{\Gamma \to A}\ (W)$$

flec(直観主義部分構造論理)

シーケントの左辺は multi set と考える。

$$A \to A$$

$$\frac{\Gamma \to A \quad \Delta, B \to C}{\Gamma, \Delta, A \supset B \to C}\ (\supset L) \qquad \frac{\Gamma, A \supset B \to A \quad \Delta, A \supset B, B \to C}{\Gamma, \Delta, A \supset B \to C}\ (\supset L)$$

$$\frac{\Gamma, A \supset B \to A \quad \Delta, B \to C}{\Gamma, \Delta, A \supset B \to C}\ (\supset L) \qquad \frac{\Gamma \to A \quad \Delta, A \supset B, B \to C}{\Gamma, \Delta, A \supset B \to C}\ (\supset L)$$

$$\frac{\Gamma, A \to B}{\Gamma \to A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \to C}{\Gamma, A * B \to C}\ (*L) \qquad \frac{\Gamma, A * B, A, B \to C}{\Gamma, A * B \to C}\ (*L)$$

$$\frac{\Gamma \to A \quad \Delta \to B}{\Gamma, \Delta \to A * B}\ (*R)$$

$$\frac{\Gamma, A \to C}{\Gamma, A \wedge B \to C}\ (\wedge L) \qquad \frac{\Gamma, A \wedge B, A \to C}{\Gamma, A \wedge B \to C}\ (\wedge L)$$

$$\frac{\Gamma, B \to C}{\Gamma, A \wedge B \to C}\ (\wedge L) \qquad \frac{\Gamma, A \wedge B, B \to C}{\Gamma, A \wedge B \to C}\ (\wedge L)$$

$$\frac{\Gamma \to A \quad \Gamma \to B}{\Gamma \to A \wedge B}\ (\wedge R)$$

$$\frac{\Gamma, A \to C \quad \Gamma, B \to C}{\Gamma, A \vee B \to C}\ (\vee L) \qquad \frac{\Gamma, A \vee B, A \to C \quad \Gamma, A \vee B, B \to C}{\Gamma, A \vee B \to C}\ (\vee L)$$

$$\frac{\Gamma \to A}{\Gamma \to A \vee B}\ (\vee R) \qquad \frac{\Gamma \to B}{\Gamma \to A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \to}{\Gamma \to \neg A}\ (\neg R) \qquad \frac{\Gamma, A \to}{\Gamma \to \neg A}\ (\neg R)$$

il(直観主義命題論理)

シーケントの左辺は set と考える。

$$\Gamma, A \to A$$

$$\frac{\Gamma \to}{\Gamma \to A}\ (W)$$

$$\frac{\Gamma, A \supset B \to A \quad \Gamma, B \to C}{\Gamma, A \supset B \to C}\ (\supset L) \qquad \frac{\Gamma, A \to B}{\Gamma \to A \supset B}\ (\supset R)$$

$$\frac{\Gamma, A, B \to C}{\Gamma, A \wedge B \to C}\ (\wedge L) \qquad \frac{\Gamma \to A \quad \Gamma \to B}{\Gamma \to A \wedge B}\ (\wedge R)$$

$$\frac{\Gamma, A \to C \quad \Gamma, B \to C}{\Gamma, A \vee B \to C}\ (\vee L)$$

$$\frac{\Gamma \to A}{\Gamma \to A \vee B}\ (\vee R) \qquad \frac{\Gamma \to B}{\Gamma \to A \vee B}\ (\vee R)$$

$$\frac{\Gamma, A \to}{\Gamma \to \neg A}\ (\neg R) \qquad \frac{\Gamma, \neg A \to A}{\Gamma, \neg A \to}\ (\neg L)$$

k, kt, s4, s5, s5−like, k+(様相論理)

シーケントの両辺は集合と考える。これは (T) 規則等が multi set と考えると停止しないからである。集合と考えた場合は、同じシーケントが下方に出現しないとう条件を加えて停止するようになる。

- k $\cdots$ cl に ($\Box$) 規則を加えたもの。
- kt $\cdots$ cl に $(\Box), (t)$ 規則を加えたもの。
- s4 $\cdots$ cl に $(\Box), (t), (4)$ 規則を加えたもの。
- s5 $\cdots$ cl に $(\Box), (t), (5), (Cut)$ 規則を加えたもの。
- s5-like $\cdots$ cl に $(\Box), (t), (5)$ 規則を加えたもの。

$$\frac{\Gamma \to A}{\Box\Gamma, \Delta \to \Box A, \Pi}\ (\Box) \qquad \frac{A, \Box A, \Gamma \to \Delta}{\Box A, \Gamma \to \Delta}\ (T)$$

$$\frac{\Box\Gamma \to A}{\Box\Gamma, \Delta \to \Box A, \Pi}\ (4) \qquad \frac{\Box\Gamma \to \Box\Pi, A}{\Box\Gamma, \Delta \to \Box\Pi, \Box A, \Sigma}\ (5)$$

$$\frac{\Gamma \to \Delta, A \quad A, \Gamma \to \Delta}{\Gamma \to \Delta}\ (Cut)$$

k+(反例出力付き様相論理)

k の機能を強めて証明不可能のときに couter kripke model を出力するようにしたもの。cl に次の (□) 規則を加えている。

$$\frac{\Gamma \to A_1 \quad \cdots \quad \Gamma \to A_n}{\Box\Gamma, \Delta \to \Box A_1, \cdots, \Box A_n, \Pi} \ (\Box)$$

clprover(古典述語論理)

インターネットに接続している場合、神戸大の田村直之さんの古典述語論理の prover が利用できる。ここでは、述語変数はアルファベットの大文字、変数はアルファベットの小文字である。$(R\forall), (L\exists)$ には y が下式に現れないという適用条件がある。

$$\frac{A(y), \forall x.A(x), \Gamma \to \Delta}{\forall x.A(x), \Gamma \to \Delta} \ (L\forall) \qquad \frac{\Gamma \to A(y), \forall x.A(x), \Delta}{\Gamma \to \forall x.A(x), \Delta} \ (R\forall)$$

$$\frac{A(y), \exists x.A(x), \Gamma \to \Delta}{\exists x.A(x), \Gamma \to \Delta} \ (L\exists) \qquad \frac{\Gamma \to A(y), \exists x.A(x), \Delta}{\Gamma \to \exists x.A(x), \Delta} \ (R\exists)$$

llprover(線型論理)

clprover と同様に、llprover も利用できる。詳細は http://bach.seg.kobe-u.ac.jp/llprover/ に譲る。が、parser を書く気力がなかったので、入力と出力が異なることに注意。

A.6.3 型付けシステム

type-assign, type-assign-trace(type assignment system TA$_\lambda$)

ラムダ項を入力すると p.t.s. を証明図の形で返す。type-assign-trace ではこのトレース実行をする。

$$\frac{\begin{matrix}[x] : A \\ \vdots \\ M : B\end{matrix}}{\lambda x.M : A \to B} \ (\to I) \qquad \frac{M : B \to A \quad N : B}{MN : A} \ (\to E)$$

A.7 FAQ 集

A.7.1 TeX の出力と xpe の表示が違うのですが

仕様です[6]。 まぁ、xpe はあくまでエディターなので、実際の出力と異なるのはしょうがないですね。これ以上 TeX に近付けるとなると、TeX そのものをインプリメントすることになってしまうので、そんなことはやりません。

A.7.2 日本語を扱いたいのですが

私も扱いたいです。 誰かインプリメントしてくれないかなぁ。Athena Widget を使っているので割と簡単にできるらしいのですが、私は知りません。「右」と「左」しか扱わないのなら、TeX に

[6] 出たな伝家の宝刀

```
\def\R{\mbox{右}}
\def\L{\mbox{左}}
```

と書いておいて、xpe では「右」,「左」の代わりにそれぞれ '\R','\L' を埋め込んでおくという手があります。

A.7.3 xpe はどう読むのですか

「エックスピーイー」と読みます。「エックスペッ」という呼び方は「カトちゃんぺ」と混同しやすい[7]ので呼んではいけません。

A.7.4 Windows や Macintosh には移植されないのですか

いっそのこと UNIX を覚えたら？Windows なら Cygwin (http://sources.redhat.com/cygwin/) をインスールすれば動作することを確認しました。ただし、その設定を書くと長くなるので、必要な人は自分あるいは周りの人と挑戦してみて下さい。

慶應の白旗優さん (`http://www.fbc.keio.ac.jp/~sirahata/`) のように Java で書けばどんなプラットホームでも動いたのでしょうが、C で書いてしまったので移植するのは大変ですね。でも、始めから Java で書けば、Java には lex や yacc があるし、ユニコードで数式や日本語が表現できるので、かえって簡単だったかも知れません。まぁ、C で書くことによって軽いアプリケーションになったのでよしとしよう。

もし PC–X Server がインストールされていて近くに UNIX マシンがあるのなら、PC–X Server 経由で起動するという手もあります。検証はしていませんが、カット&ペーストを用いてエディターとやりとりもできるでしょう。

A.7.5 複数の証明図をエディットしたいのですが

今のところはサポートしていませんが、証明図の一番下のセルの左右に証明図を挿入することにより、擬似的に複数の証明図がエディットできます (A.4.5 参照)。また、

```
% xpe &
```

などとしてコマンドラインから複数の xpe を立ち上げれば、その 2 つの xpe 間でカット&ペーストができます。ただ、auto–save file(A.3.7 節参照) は共通になるので、気を付けて使って下さい。

A.8 既知のバグ

いまのところはない。

A.9 xpe メーリングリスト

xpe に関する情報交換の場として、xpe メーリングリストが運営されている。参加するには

```
majorodmo-j@logic.jaist.ac.jp
```

にあてて、本文に

```
subscribe xpe
```

と書いたメールを送ると送り主のアドレスがメーリングリストに登録される。

[7] はて!?誰が間違えるのだろう？

A.10 謝辞

東北大の鈴木太朗さんには、yacc の使い方をいろいろと教わりました。おかげでバグバグしてた xpe がこんなにまともになりました。ありがとうございますです、はい。

また、日本大学の志村立矢さんには FreeBSD に対応して頂きました。この場を借りて御礼を申し上げます。

関連図書

[1] Charles Donnelly, Richard M. Stallman 著, 石川直太 訳, ***Bison* 入門**, アスキー出版局, 1999.

[2] Les Hancock 他著, 倉骨彰 他訳, **改定第 *3* 版 *C* 言語入門**, アスキー出版局, 1999.

[3] Donald E.Knuth 著, 鷺谷好輝 訳, ***TEX* ブック**, アスキー, 1992.

[4] John R. Levine, Tony Mason, Doug Brown 著, 村上列 訳, ***lex&yacc* プログラミング**, アスキー, 1992.

[5] G. T.Nicol 著, 市川和久 訳, ***Flex* 入門**, アスキー出版局, 1999.

[6] Adrian Nye 他著, 今泉貴史 監訳, ***X* ツールキット · イントリンシクス · プログラミング · マニュアル**, ソフトバンク, 1992.

[7] E Kiriyama, H Ono, ***The contraction rule and decision problems for logics without substractural rules***, Studia Logica 50 (1991), 299–319.

[8] Motohiko Mouri ***Theorem Provers with Counter Models and xpe***, Bulletin of the Section of Logic, to apear.

[9] Chris D.Peterson, ***Athena Widget Set – C Language Interface***, MIT X Consortiumm, 1992.

[10] Lincoln A. Wallen, ***Automated Proof Search in Non-Classical Lotjgics***, Efficient Matrix Proof Methods for Modal and Intuitionistic Logics MIT Press, 1990.

[11] H Wang, ***A survey of mathematical logic***, Studies in Logic and the Foundations of Mathematics, North-Holland, Amsterdam, 1963.

[12] B.W. カーニハン 他著, 石田晴久 訳, **プログラミング言語 *C* 第 *2* 版**, 共立出版株式会社, 1989.

[13] J. ホップクロフト 他著, 野崎昭弘 他訳, **オートマトン 言語理論 計算論 *I***, サイエンス社, 1984.

[14] 安居院猛 他著, ***X* アプリケーションプログラミング *2 Athena* ウィジェット編**, 新紀元社, 1992.

[15] 小野寛晰 著, **情報科学における論理**, 日本評論社, 1994.

[16] 木下凌一 他著, ***X-Window ver.11* プログラミング (第 *2* 版)**, 日刊工業新聞社, 1993.

[17] 五月女健治 著, ***yacc/lecc* プログラムジェネレータ *on UNIX***, 啓学出版, 1992.

[18] 森本寛 著, ***X Toolkit* プログラミングテクニック**, 日刊工業新聞社, 1992.