

”Proofs and Types” ノート

Rei Tomori

2025 年 7 月 19 日

1 意義，表示と意味論

2 自然演繹

3 Curry-Howard 同型

3.1 表示的意義

型は現在議論されている対象の種類を表わす．たとえば型 $U \rightarrow V$ の対象は U から V への関数，型 $U \times V$ の対象は U の対象と V の対象の順序対といったように．

原子型の意味は重要ではない^{*1}．文脈による．

項は我々が Heyting の意味論と自然演繹で用いた 5 つの図式に極めて正確に従う．

1. 型 T の変数 x^T は，型 T の任意の項 t を表わす．ただし x^T は t で置換されるとする．
2. $\langle u, v \rangle$ は u と v の順序対である．
3. $\pi^1 t$ と $\pi^2 t$ はそれぞれ t の第一成分と第二成分への射影である．
4. $\lambda x^U. v$ は型 U の各項 u を $v[u/x]$ ，すなわち x^T を u の略記としたときの v に写す関数である．

表示的には，以下の secondary な等式

$$\begin{aligned}\langle \pi^1 t, \pi^2 t \rangle &= t \\ \lambda x^U. tx &= t \quad (x \notin \text{FV}(t))\end{aligned}$$

に加え，(primary な) 等式がある：

$$\begin{aligned}\pi^1 \langle u, v \rangle &= u \\ \pi^2 \langle u, v \rangle &= v \\ (\lambda x^U. v)u &= v[u/x]\end{aligned}\tag{1}$$

これらは適切な立ち位置を与えられてこなかった．

Thm 1. これらの等式によって定められる体系は健全かつ決定可能である．

^{*1} 命題変数と思うことにする

健全とは、相異なる変数 x, y に対して $x = y$ が証明不可能であることである。

この結果はいずれの等式においても成り立つが、最初の 3 つに対してのみ考えることにする。この定理は、Church-Rosser 性と正規化定理の帰結である (§4)。

3.2 操作的な意義

等式規則により、 λ 項の表示的意味が定められた。次に、 λ 項の操作的意味について考察しよう。

一般に、 λ 項はプログラムを表わす。プログラムの目的はその操作的意味を計算することである。(たとえば、 $(\lambda x. x + 1)2$ を計算することで、その表示的意味である 5 を得る)

対して、型はプログラムの仕様を定める。プログラムの型が与えられたとき、仕様について何が言えるだろうか。

仕様に関する言明として「このプログラムは 2 つの整数の和を計算する」を例に取り考える。この言明は十分に詳細だろうか。つまり、この計算がどう行なわれるか主張できるだろうか。^{*2}それとも、この言明は詳細すぎるだろうか。たとえば、プログラムが 2 つの整数値を取り、整数値を返すことはいえるだろうか。^{*3}

構文に限っていえば、この答は型システムに依存し、明確でない。たとえば、本書で導入する型システム (System F) でこのプログラムに型を付ける場合、 $\text{int} \rightarrow \text{int} \rightarrow \text{int}$ となり、最も明白な情報しか提供しない。対して、Martin-Löf の型システムなどの型体系はプログラムが何を計算するかに関する情報を与える。^{*4}つまり、型システムが表示的情報を与える。

より一般的なレベルにおいては、構文的差異を無視することになると、型は対象と一緒に繋げる命令と見做せる。たとえば、モジュール^{*5}を用いてプログラムを構成することを考えよう。モジュールは OOP におけるクラス同様、内部は外から隠蔽されているものとする。モジュールの型は (衝突しないように選ばれた) 全ての可能な引数の型によって決定される。したがって、特にモジュールは同じ型シグネチャを持つ他のモジュールに置換できる。しかし、この見方を数学的に定式化することは難しい。

型 U_1, \dots, U_n の変数 x_1, \dots, x_n に依存する型 T の項 t を考えよう。この t は、§3.2 における関数型の解釈、つまり型 $U \rightarrow V$ を持つ項 $\lambda x^U. v$ を型 U の各項 u に対して $v[u/x]$ を返す関数と解釈するわけにはいかない。むしろ、パラメタ同士を繋ぎ合わせる命令と見做すべきである。モジュールを表わす項には適切な型の入力挿入する場所がある。たとえば、モジュールにおけるパラメタ x_i の各出現は項 $u_i : U_i$ が挿入される可能性を指す。 x_i に u_i が代入される各インスタンスにおいて、 u_i は x_i に同時代入される。さらに、項 t も他のモジュールのパラメタとして代入される可能性がある。つまり、モジュールのインスタンスを作るために入力として引数に値が代入され、さらにモジュール自身も他のモジュールの引数として取られうることが分かる。

変数と値を同じ現象の双対的な側面と見做すこの見方によれば、アルゴリズムの実行とは対称的な入出力のプロセスとして理解できる。アルゴリズムはいくつかの引数を取り、更にアルゴリズム自身も (高階関数である) 別のアルゴリズムの引数となりうる。

^{*2} 機械語レベルにおいて?

^{*3} いえない。2 整数が浮動小数点値にキャストされる可能性があるため。

^{*4} Agda でベクトルの末尾を取る関数 `tail` は、

```
1 tail : {A:Set}{n:Nat}Vec A (suc n) -> Vec A n
2 tail (_::xs) = xs
```

^{*5} ML における?

項の振る舞いを操作的に説明するために、表示的意味を定める等式規則から、左辺から右辺への書き換え規則を作ろう。この書き換え規則は項を計算するプログラム (i.e. インタプリタ) を定めるものと見做せる。プログラムの最終的な結果こそがプログラムの意味を与えるので、操作的意味の理解を深めるために以後正規化について考察する。

3.3 変換

これ以上評価規則を適用できない項を正規形と呼ぶ。

Definition 1. 項 t が正規形であるとは、 t の任意の部分項が以下の形でないことである：

$$\pi^1 \langle u, v \rangle, \pi^2 \langle u, v \rangle, (\lambda x^U. v)u$$

1 ステップの評価を変換という。

Definition 2 (変換). 項 t が t' に変換されるとは、 t, t' が以下の条件を満たすことである：

1. $\exists u, v. t = \pi^1 \langle u, v \rangle \wedge t' = u.$
2. $\exists u, v. t = \pi^2 \langle u, v \rangle \wedge t' = v.$
3. $\exists u, v. t = (\lambda x^U. v)u \wedge t' = v[u/x].$

t をレデックス (簡約基), t' をコントラクタムという。以下の進行補題により、redex と contractum の型は常に等しい。

Lemma 1 (進行). 任意の項 t, t' に対して、 $t : T$ かつ $t \rightarrow t'$ ならば $t' : T$ である。

Proof. t, t' を任意に取り、 $t : T$ とする。 $t \rightarrow t'$ を仮定し、 t の形で場合を分ける。

- t が組の射影の形で書けるとき。第一成分の場合を示す。 t が $\pi^1 \langle u, v \rangle$ のとき。 $v : T'$ とすると、 $\pi^1 : T \times T' \rightarrow T$ より、 $t : T$ である。評価規則より $t' = v : T'$ なので従う。
- t が β 簡約基の形で書けるとき。 $u : U, v : T$ および v に束縛出現を持たない $x^U : U$ により、 t は $(\lambda x^U. v)u$ の形で書ける。評価規則を適用することにより、 t のコントラクタム t' は $t' = v[u/x]$ であり、型は $(x^U$ の各出現を v で置き換えているので) T のままである。

□

多ステップの変換を簡約という。

Definition 3 (簡約). 項 u が項 v に簡約されるとは、 $n \in \mathbb{N}$ と列 $u = t_0, t_1, \dots, t_{n-1}, t_n = v$ が存在し、 $\forall i \in \mathbb{N}, i < n \Rightarrow t_i \rightarrow t_{i+1}$ なることであり、 $u \rightsquigarrow v$ と書く。

Proposition 1. 二項関係 (\rightsquigarrow) は (\rightarrow) の反射推移閉包である。

Proof. 反射性。各項 t に対して簡約列を $t_0 = t$ とすれば即座に従う。推移性。任意の t, u, v に対して $t \rightsquigarrow u, u \rightsquigarrow v$ を仮定する。定義より、リダクション列 $\{t_i\}_{i=0}^m, \{u_i\}_{i=0}^n$ を取れる。そこで、新たなリダクション列 $\{t'_i\}_{i=0}^{m+n}$ を、 $\forall i \leq m, t'_i = t_i, \forall i \leq n, t'_{m+i} = u_i$ と定めることができる。以上より、 $t \rightsquigarrow v$ が従う。 □

Definition 4 (正規形). t の正規形とは、 $t \rightsquigarrow u$ なる正規形 u である。

以後、STLC において正規形が一意的に存在することを示す。その準備として、頭部正規形と正規形の等価性を示そう。

Definition 5 (頭部正規形).

Lemma 2. 項 t が正規形であることの必要十分条件とは、 t が頭部正規形

$$\lambda x_1. \lambda x_2. \dots \lambda x_n. y \ u_1 \ u_2 \ \dots \ u_m \quad (2)$$

であることである。ただし y は x_i の何れかに一致しうる。また、各 j に対して u_j は正規形である。

4 正規化性定理

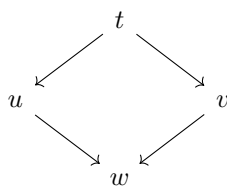
この章は、型付き λ 計算が計算論的に良い振る舞いをする事を保証する 2 つの結果について扱う; **正規化性定理**は正規形の存在性を、*Church-Rosser 性*はその一意性を保証する。^{*6}正規化性定理には 2 つの形がある。

- **弱正規化性定理** (正規化を行なうための評価戦略で、停止するものが存在する。) この主張はこの節で扱う。
- **強正規化性定理** (全ての評価戦略において正規化が停止する); これは §6 で示す。

4.1 Church-Rosser 性

この性質は、正規形の一意性を、その存在性とは独立して主張する。実際、それは型無し λ 計算のような正規化定理の成り立たない体系にとって意味がある。

Thm 2. $t \rightsquigarrow u$ ならば、 $u, v \rightsquigarrow w$ であるような w が存在する。



Corollary 1 (A). 項 t は高々 1 つの正規形を持つ。

Proof. $t \rightsquigarrow u, v$ なる正規形 u, v を任意に取る。このとき、定理より w が存在して $u, v \rightsquigarrow w$ 。

しかし u, v は正規形なのでそれら自身にのみ簡約される。つまり $u = v, v = w$ 。ゆえに $u = v$ 。□

Church-Rosser の証明には多少デリケートな部分がある (少なくとも力づくで示そうとする場合)。この性質は多種多様な体系で成り立ち、証明は常にはば一緒である。

Church-Rosser から即座に従う補題として、体系の健全性がある: (u と v が同じ型として) 任意の等式 $u = v$ が 1 の等式から演繹できるわけではない。実際、以下のことに注意を払おう:

^{*6} Church-Rosser は示さない。Barendregt などを見よ。

- $u \rightsquigarrow v$ ならば $u = v$ は 1 および等号の一般的公理から導出できる.
- 逆に, 1 と等号の公理から $u = v$ が演繹できるならば, 項 $u = t_0, t_1, \dots, t_{2n-1}, t_{2n} = v$ が存在し, 各 $i = 0, 1, \dots, n-1$ に対して $t_{2i}, t_{2i+2} \rightsquigarrow t_{2i+1}$. Church-Rosser の定理を繰返し適用することで, $u, v \rightsquigarrow w$ となる w の存在を得る.

いま, u, v が同じ型の相異なる正規形とする. このとき, $u, v \rightsquigarrow w$ なる w は存在しない. そのため $u = v$ の証明はできない.

4.2 弱正規化定理

Thm 3 (弱正規化定理). 任意の項 u に対して, 一意な正規形 v に至るようなリダクション列 $u = t_0, t_1, \dots, t_n = v$ が存在する.

ここから, 表示的等価性の決定可能性が従う. つまり, $u = v$ は有限のステップで証明可能である (正規形に至るリダクション列をそれぞれに対して構成でき, その正規形は一意なため.) 実際, 我々は等式 $u = v$ が証明可能であることと, ある w に対して $u, v \rightsquigarrow w$ なることが同値であることをみた. そして w は正規形を持つので, この正規形は u, v に対する共通の正規形となる. したがって, u と v の表示的等価性を決定するためには, 次のように議論を進めればよいことが分かる:

- 最初のステップで, u と v の正規形を計算する.
- 2 つめのステップで, それらを比較する.

今に至るまで, 評価戦略を定めていない^{*7}ことに注意する. そのため, 簡約は決定的でない. つまり, 任意に t を取って固定すると, t の部分項に対して (有限個だが) 複数の簡約が可能である. 定理は”適切な”簡約を経ることで正規形を決められることを主張していると分かる. もちろん, 正規形に至らない評価も可能である.

とはいえ, 項のリダクション列たちを木で表わすと, これは有限分岐である. したがって, 全て書き下せば決定することはできる.

強正規化定理は, (正規形に至るという意味で) 全ての正規化戦略が良いことを保証し, 状況をより単純にする. ただし, 計算量が他より少ない簡約戦略も存在する.

4.3 弱正規化定理の証明

はじめに, 型もしくは項の大きさを degree と呼ぶことにしよう. 定義は次のようになる.

Definition 6 (型の大きさ). 型 T の大きさ $\partial(T)$ を, 型のつくる木の高さにより定める:

- T_i が原子型ならば $\partial(T_i) = 1$.
- $\partial(U \times V) = \partial(U \rightarrow V) = \max(\partial(U), \partial(V)) + 1$.

Definition 7 (簡約基の次数). 簡約基の大きさ $\delta(r)$ を次のように定める.

- $\partial(\pi^1 \langle u, v \rangle) = \partial(\pi^2 \langle u, v \rangle) = \partial(U \times V)$

^{*7} 強いていえば完全 β 簡約を戦略として採用している

- $\partial((\lambda x.v)u) = \partial(U \rightarrow V)$ ただし $(\lambda x.v) : U \rightarrow V$.

項の大きさ $d(t)$ を, t に含まれる簡約基の次数の上限により定める. ただし, 正規形の degree は 0 とする.

簡約基 r は簡約基としての大きさと項としての大きさの 2 つの degree を持つ. 定義より $\partial(r) \leq d(r)$.

4.3.1 Degree and Substitution

Lemma 3. $x : U$ とする. このとき, $d(t[u/x]) \leq \max(d(t), d(u), \partial(U))$.

Proof. 定義より, $d(t)$ は $t[u/x]$ に含まれる簡約基の degree の上限である. そこで, $t[u/x]$ に含まれる redex が何か考えよう.

- t の簡約基
- u の簡約基
- それ以外の簡約基, たとえば x が組もしくは関数適用の関数部に出現する場合で, u が組か抽象である場合. この degree は $\partial(U)$ である.

以上の最大値が $d(t)$ を与えるので, 主張が成り立つ. □

4.3.2 Degree and Conversion

r が型 T の簡約基ならば, $\partial(r) > \partial(T)$ である. 実際, r が組の射影の形で書けるとき, T は r の引数 $\langle u, v \rangle : T \times U$ の第一 (第二) 成分の型であり, 定義から従う. λ 項の正規形の場合も同様.

Lemma 4. $t \rightsquigarrow u$ ならば $d(u) \leq d(t)$.

多ステップの場合は 1 ステップの場合から従う. 実際評価列の長さに関する帰納法から示せる. そこで, より弱めた, 1 ステップに関する主張を示せば十分である.

Lemma 5. $t \rightarrow u$ ならば $d(u) \leq d(t)$.

Proof. u は t に対し, t のある簡約基 r をコントラクトム c で置き換えることで得られる. 以下, u に含まれる redex の形について場合を分ける.

- r を c で置換することで変更される, t に出現したが r に含まれない簡約基. このとき, $d(r) = \partial(r)$ なので $d(u) = d(t)$
- c の簡約基. c は r の簡約によって得られる, もしくは r の内部の簡約によって得られる. つまり, $(\lambda x.s)s'$ は $s[s'/x]$. また, §4.3.1. の補題より $d(s[s'/x]) \leq \max(d(s), d(s'), \partial(T))$. 最初に確認したように $\partial(T) < d(r)$ なので, $d(c) \leq d(r)$.
- r を c によって置き換えることで得られる簡約基. この場合の次数は §4.3.1 により $r : T$ として $\partial(T)$ に等しい. また, $\partial(T) < \partial(r)$ が成り立つ.

□

4.3.3 最大次数の項の変換

Lemma 6. r を t における最大の次数 n を持つ簡約基とする. また r 自身を除いた r に含まれる簡約基の次数は n 未満とする. u が t において r をコントラクト c で置き換えることで得られる項ならば, u の次数 n の簡約基の個数は t より真に小さい.

Proof. t において, 最大の次数 n を持つ簡約基が r に限られると仮定する. β 変換を施したとき, 以下が成り立つ.

- r 外部の簡約基は変わらないので次数には影響しない.
- r の内側の簡約基は一般に簡約されているが, 増加している場合もある. たとえば, $(\lambda x. \langle x, x \rangle)s \rightarrow \langle s, s \rangle$ を考えればよい. ただしこの場合, 仮定より s の次数は n 未満である (そのため, t の次数は n 未満).
- r は破壊され, 補題 5 より r 未満の項に書き変わる. r は redex であり, c と次数は等しくなりえない.

したがって補題 5 より, $d(t) = \max(t_1, t_2, \dots, r, t_{k+1} \dots)$ に対し, $d(u) = \max(t_1, t_2, \dots, c, t_{k+1} \dots) < n$. □

4.3.4 定理の証明

§4.3.3. から, 任意の項 t に対し, 各ステップごとに項の最大の次数を持つ簡約基を選んで簡約すれば, リダクション列に現れる項の次数は狭義減少列をなす. \mathbb{N} 上の順序は整礎なので, かならずリダクション列は有限の長さとなることが保証される. これをより正確に述べよう.

Thm 4 (弱正規化定理). 任意の項 u に対して, 一意な正規形 v に至るようなリダクション列 $u = t_0, t_1, \dots, t_n = v$ が存在する.

Proof. t を項とする. 項上の関数 $\mu(t) = (n, m)$ を,

$$\begin{aligned} n &= d(t) \\ m &= t \text{ の次数 } n \text{ の簡約基の個数} \end{aligned}$$

として定める. t を任意に取り, $\mu(t) = (n, m)$ とする. 整礎帰納法で示す.

補題 6 により, t の簡約基 r であって, r そのものでない r 内の簡約基の次数が n 未満のものを取る. このとき, t の r の出現を r のコントラクト c で置き換えた項を t' とすると, t' の redex の次数 n の項は t より真に少ない.

$\mu(t') < \mu(t)$. t' に対して同じ操作を施し, その後同様の操作を繰替えして得られたリダクション列を構成できる. これを $\{t_i\}_{i=0}^{\infty}$ とする. 同様の議論により $\dots d(t_1) < d(t)$ である. これが無限列をなすとしよう. すると $n \geq 0$ より m が無限降下列をなすことになるが, これは \mathbb{N} が整礎集合であることに反し矛盾する. ゆえにリダクション列は有限列であり, 主張が示された. □

4.4 強正規化定理

正規形に至る redex を各簡約ステップで選ぶアルゴリズムは補題 6 によって定められた。そのため、弱正規化定理は (より構成的な意味合いを持つため) その主張よりも若干優れているといえる。それでもやはり、全ての正規化戦略が停止するかという問いは興味深い。

Definition 8. 項 t が強正規化可能とは、 t から始まる無限長の簡約列が存在しないことである。

Lemma 7. t が強正規化可能である必要十分条件とは、 t から始まる全ての簡約列の長さを上から抑える $\nu(t) \in \mathbb{N}$ が存在することである。

Proof. (\Rightarrow) t から始まる全ての簡約列が $\nu(t)$ で抑えられたとする。このとき、 t の任意の簡約列は有限長なので即座に従う。

(\Leftarrow) t が強正規化可能であるとする。このとき、 t の任意の簡約列は有限長である。以下の定理を思いだそう：

Lemma 8 (König の補題)。有限分岐の無限木には無限長の枝が存在する。

この対偶を取れば、有限分岐で任意の枝が有限長の木は有限木であることが即座に従う。^{*8}いま、 t から始まる (有限の) 簡約列 $\{t_i\}_{i=1}^N$ が与えられたとき、各変換ステップにおける変換列は t_i の簡約対象の簡約基 r_i により特徴付けると約束する。

すると、可能な簡約列は木の形で表わすことができる。 t_i が r_i を変換して t_{i+1} を得るとき、ノード t_i からノード t_{i+1} に至る辺を r_i でラベル付けすればよい。各ステップにおいて、項は有限個の部分項を含むので特に分岐は有限となる。仮定より全ての枝は有限長なので、König の補題からこの木は有限木である。

よって、特に $\nu(t)$ をこの木の大きさとして定めれば、 $\nu(t)$ は簡約列の長さの上界となるので示される。□

型付き λ 計算の任意の項が強正規化可能であることの証明には、いくつかの方法がある：

- 内部化。弱正規化性の場合に証明を帰着できるように計算体系を (回りくどい形で) 翻訳する方法。
- 簡約可能性。複雑な組合せ的信息を操作するための方法である”遺伝的計算可能性”の性質を導入する。この方法は非常に複雑な状況に一般化できる唯一の方法なので、以後この手法を使うことにする。この手法は第 6 章の主題となるだろう。

^{*8} 橙色 Gödel 本でやっていたように各枝にラベルを振らない場合は選択公理が必要。

5 シーケント計算

Gentzen によるシーケント計算は、論理の対称性を最も美しく示している。自然演繹と数多くの点で類似しているが、直観主義的な場合に制限されない。^{*9}

シーケント計算では同じ証明を書く複数の方法が存在するため、Curry-Howard 同型を持たない。^{*10} このため、シーケント計算を型付き λ 計算として用いることはできないが、並列性と (おそらく) 関連付けられる、深い部分にあるこの種の構造を見て取ることができる。^{*11} しかしながら、その対応を書き表すには、自然演繹で複数の結論を認めるなどの構文への新たなアプローチが必要となる。

5.1 計算体系

5.1.1 シーケント

Definition 9 (シーケント). シーケントとは、 \underline{A} , \underline{B} をそれぞれ論理式の有限列 $A_1, \dots, A_n, B_1, \dots, B_m$ としたときの式 $\underline{A} \vdash \underline{B}$ である。

シーケントの表示的な解釈を考えよう。 $\underline{A} \vdash \underline{B}$ を、 $\underline{A} = \bigwedge_{i=1}^n A_i \vdash \bigvee_{j=1}^m B_j = \underline{B}$ と解釈する方法である。つまり、 A_i の連言が B_j の選言を含意するというものである。ただし、 $\underline{A} = \emptyset$ のとき $\underline{A} = \top$, $\underline{B} = \emptyset$ のとき $\underline{B} = \perp$ と解釈するものとする。とりわけ、

- \underline{A} が空ならば、シーケントは B_j の選言を主張する。
- \underline{A} が空かつ \underline{B} が B_1 のみならば、 B_1 を主張する。
- \underline{B} が空ならば、 A_i の連言の否定を主張する。
- どちらも空ならば、矛盾を主張する。

5.1.2 構造規則

以下の規則は、式を挿入できる位置に制約を加える。

1. exchange 則: turnstile の右辺または左辺の式の交換を許す。論理の可換性に対応する。

$$\frac{\underline{A}, C, D, \underline{A'} \vdash \underline{B}}{\underline{A}, D, C, \underline{A'} \vdash \underline{B}} \mathcal{LX} \qquad \frac{\underline{A} \vdash \underline{B}, C, D, \underline{B'}}{\underline{A} \vdash \underline{B}, D, C, \underline{B'}} \mathcal{RX}$$

2. weakening 則: シーケントをより弱いものに置き換える。

$$\frac{\underline{A} \vdash \underline{B}}{\underline{A}, C \vdash \underline{B}} \mathcal{LW} \qquad \frac{\underline{A} \vdash \underline{B}}{\underline{A} \vdash C, \underline{B}} \mathcal{RW}$$

3. contraction 則: 選言・連言の中等律に対応する。

^{*9} 自然演繹の場合の制限とは、証明の対称性についての事だった。

^{*10} Curry-Howard 同型があったとすると、正規形の一意性に反するため

^{*11} ということか?

$$\frac{\underline{A}, C, C \vdash B}{\underline{A}, C \vdash B} \mathcal{LC}$$

$$\frac{\underline{A} \vdash C, C, B}{\underline{A} \vdash C, B} \mathcal{LC}$$

これらの規則は計算規則のうち最も重要なものである。論理記号を使うことなしに論理演算の振る舞い、とくに選言と連言の振る舞いを決定できるためである。とはいえ、表示的な観点からは明らかだったとしても、構造規則は操作的観点から検討する必要がある。

これらの構造規則を弱めたシーケント計算の体系を考えることができる。たとえば線形論理がそれである。

5.1.3 直観主義的なケース

直観主義的なシーケント計算はシーケントの形に制約を加えることで実現できる。

Definition 10. 直観主義的シーケントとは、 \underline{B} が高々 1 つの式からなる列であるようなシーケント $\underline{A} \vdash \underline{B}$ である。

したがって、右辺に対する exchange 則 \mathcal{RX} 、右辺に対する contraction 則 \mathcal{RC} は適用できない。 \underline{B} が空列であるときにのみ \mathcal{RW} が適用できる。

\vdash の位置は、contraction 則を適用できない位置を示しているといえる。その意味で、直観主義的な制約は式の管理方法を変更しているといえる。他方、適用できる規則に対する斯様な制約は、右辺と左辺の対称性を失わせていることに注意しよう。そして、対称性を回復させるためには、exchange 則と contraction 則のない体系とすればよい。

他の方法としては、古典的シーケント計算の構造規則を 1 つを除き残し、シーケントを直観主義的シーケントに制限することで体系を得る。

5.1.4 "恒等"群

1. 任意の式 C に対して、恒等公理 $C \vdash C$ が成り立つ。 C は原子式に制限してもよい。
2. それと等価な規則が cut 則である。

$$\frac{\underline{A} \vdash C, \underline{B} \quad \underline{A'}, C \vdash \underline{B'}}{\underline{A}, \underline{A'} \vdash \underline{B}, \underline{B'}} \text{cut}$$