

ラムダ計算の簡約グラフについて

群馬大学理工学部 富岡 峻士*

Takashi Tomioka

Department of Computer Science,
Gunma University

群馬大学理工学部 藤田憲悦†

Ken-etsu Fujita

Department of Computer Science,
Gunma University

1 はじめに

簡約グラフとは、 λ 項の計算経路を有向グラフで表現したものである。簡約グラフは Venturini Zilli [3] によって包括的な研究が行われ、簡約グラフに関するいくつかの予想が示された。本研究では Venturini Zilli [3] が提示した *bottleneck* と *boundary* の定理と予想について検証し、反例を与えることを目的とする。

2 λ 項

本節では、 λ 計算の概要を Barendregt [1] の定義に従って述べる。

2.1 λ 項の定義

BNF 記法を用いると λ 項は以下のように定義できる。

定義 1 (λ 項)

$$M, N, T, U ::= x \mid (MN) \mid (\lambda x.M)$$

ここで x は変数を表し、 M, N, T, U は λ 項を表すメタ変数である。 (MN) を関数適用と呼び、 $(\lambda x.M)$ を関数抽象と呼ぶ。

関数抽象 $(\lambda x.M)$ の部分項 M の中に変数 x 現れたとき、 x は束縛されているという。束縛されている変数のことを束縛変数と呼び、束縛されていない変数を自由変数と呼ぶ。 λ 項 M に含まれる自由変数の集合を $FV(M)$ と表す。 λ 項 $\lambda x.M$ の束縛変数 x を新たな変数 y (ただし $y \notin FV(M)$) に置き換えた項 $\lambda y.N$ を、元の式と同一視し、これら二つは α 同値であるという。 $\lambda x.M$ と $\lambda y.N$ が α 同値であることを $\lambda x.M \equiv \lambda y.N$ と表す。

関数適用は左結合であり、 $(M_1 M_2) M_3$ という λ 項は $M_1 M_2 M_3$ のように括弧を省略できる。また関数抽象が複数連なる $\lambda x.(\lambda y.M)$ という λ 項は $\lambda xy.M$ と省略して表記できる。さらに以下の略記を用いる。 I は恒等関数の役割を果たす λ 項、 Ω は停止しない計算を表す代表的な λ 項である。 Ω_3 は計算をするごとに、内部にある Δ_3 の数が無限に増加していく項である。

$$I \equiv \lambda x.x, \Delta \equiv \lambda x.xx, \Delta_3 \equiv \lambda x.xxx, \Omega \equiv \Delta\Delta, \Omega_3 \equiv \Delta_3\Delta_3.$$

*本研究は京都大学数理解析研究所の助成を受けたものである。

†本研究は JSPS 科研費 JP17K05343 の助成を受けたものである。

$$M^n \equiv \begin{cases} M & (n = 1), \\ (M^{n-1})M & (n \geq 2). \end{cases}$$

2.2 β 簡約

$(\lambda x.M)N$ という形の項を β レデックスと言い、 M 中にあるすべての自由変数 x を N に置き換えた項 $M[x := N]$ に書き換えることを β 簡約と言う。 M が M' に β 簡約されるとき $M \rightarrow_\beta M'$ と書く。

推論規則を用いると β 簡約を以下のように定義できる。

定義 2 (β 簡約)

$$\frac{}{(\lambda x.M)N \rightarrow_\beta M[x := N]} \text{Beta}, \quad \frac{M \rightarrow_\beta M'}{\lambda x.M \rightarrow_\beta \lambda x.M'} \text{Lam},$$

$$\frac{M \rightarrow_\beta M'}{MN \rightarrow_\beta M'N} \text{AppL}, \quad \frac{N \rightarrow_\beta N'}{MN \rightarrow_\beta MN'} \text{AppR}.$$

β 簡約の反射推移閉包を $\overset{*}{\rightarrow}_\beta$ と表す。それ以上 β 簡約できない形をした λ 項を β 正規形という。

3 簡約グラフ

本節では Barendregt [1] に従って簡約グラフを定義する。そして Venturini Zilli [3] が導入した平面と *boundary* について述べる。

3.1 簡約グラフの定義

λ 項の簡約は一通りに決まるわけではなく、一般に何通りも存在する。例えば $(\lambda x.xx)((\lambda y.y)z)$ という λ 項の β 簡約は、図 1 に示すように複数通りの経路が存在する。

上記の式の簡約経路をグラフ化すると図 2 が得られる。各頂点が図 2 中の λ 項に対応し、有向辺が β 簡約を表していることが確認できる。ここで青い頂点が λ 項 $(\lambda x.xx)((\lambda y.y)z)$ 、赤い頂点が β 正規形を表している。このように λ 項を頂点とし β 簡約を有向辺で表したグラフを簡約グラフと言い、 λ 項 T の簡約グラフを $\mathcal{G}(T)$ と表記する。

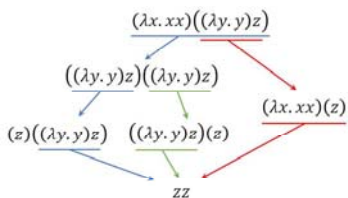


図 1: 簡約経路

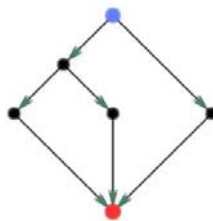


図 2: $\mathcal{G}((\lambda x.xx)((\lambda y.y)z))$

Barendregt [1] による簡約グラフ $\mathcal{G}(T)$ の定義を以下に示す。

定義 3 (簡約グラフ)

1. 頂点集合 $V(\mathcal{G}(T))$ は $\{N \mid T \overset{*}{\rightarrow}_\beta N\}$ である。
2. $M, N \in V(\mathcal{G}(T))$ とする。 $M \rightarrow_\beta N$ ならば、頂点 M から頂点 N に有効辺を引く。異なる位置に出現する β レデックスを β 簡約して $M \rightarrow_\beta N$ となる場合、 β レデックスの数だけ頂点 M から頂点 N に有向辺が引かれる。

β 簡約の結果が同じでも異なるレデックスの出現位置で簡約した場合、両者を区別するために簡約グラフでは多重辺とする。また α 同値の関係にある頂点は一つの頂点に統合する。このような例を図 3 に示す。

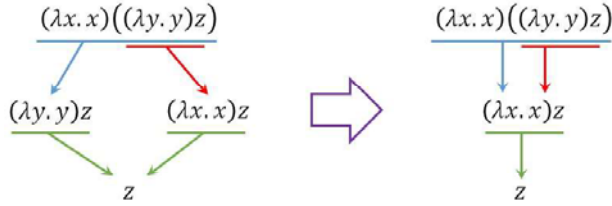


図 3: α 同値と多重辺の例

3.2 平面の定義

平面とは Venturini Zilli [3] が導入した概念で、簡約グラフの頂点集合を同値類で分割したものである。平面の定義と、平面同士の関係を次に述べる。

λ 項 T, U 間の同値関係 \sim を以下で定義する。

定義 4 ($T \sim U$)

$$T \sim U \Leftrightarrow T \xrightarrow{*}_{\beta} U \wedge U \xrightarrow{*}_{\beta} T$$

λ 項 T が属する同値類を T/\sim と表す。すなわち、 $T/\sim = \{U \mid T \sim U\}$ である。簡約グラフにおいて、 T/\sim は頂点 T を含む強連結な頂点の集合である。同値関係 \sim で分割された同値類たちを平面と呼び、平面の要素を点と呼ぶ。以降、平面を表す文字として \mathcal{A}, \mathcal{B} を用いる。

平面から平面への簡約 $\circ \rightarrow$ を以下に定義する。また、 $\circ \rightarrow$ の反射推移閉包を $\circ^* \rightarrow$ と表記する。

定義 5 (平面から平面への簡約)

$$\mathcal{A} \circ \rightarrow \mathcal{B} \Leftrightarrow \exists M \in \mathcal{A}, \exists N \in \mathcal{B}, M \rightarrow_{\beta} N \wedge M \neq N$$

$(\mathcal{G}(T)/\sim, \circ \rightarrow)$ によるグラフを $\mathcal{G}^0(T)$ と表し、凝縮簡約グラフと呼ぶ。通常の簡約グラフと区別するため、凝縮簡約グラフでは頂点を \circ で表す。例として $\mathcal{G}(\Delta(I\Delta))$ と $\mathcal{G}^0(\Delta(I\Delta))$ を図 4 に示す。

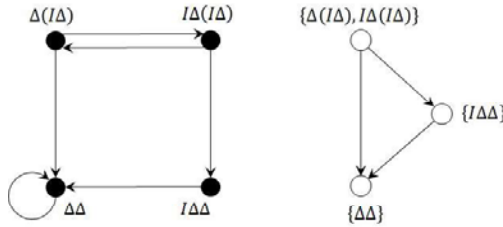


図 4: $\mathcal{G}(\Delta(I\Delta))$ と $\mathcal{G}^0(\Delta(I\Delta))$

3.3 bottleneck の定義

$\mathcal{G}^0(T)$ における平面 \mathcal{A}, \mathcal{B} について、 \mathcal{B} は \mathcal{A} の bottleneck であることを以下で定義し、 $\text{Bn}(\mathcal{B}, \mathcal{A})$ と書く。

定義 6 (bottleneck)

$\text{Bn}(\mathcal{B}, \mathcal{A}) \Leftrightarrow$ すべての $M \in \mathcal{A}, N \in \mathcal{B}, L \in V(\mathcal{G}(T))$ に対し、 $M \xrightarrow{*}_{\beta} L$ かつ $N \xrightarrow{*}_{\beta} L$ ならば、 $M \xrightarrow{*}_{\beta} P \xrightarrow{*}_{\beta} L$ となる $P \in \mathcal{B}$ が存在する。

特に $\text{Bn}(\mathcal{B}, T/\sim)$ であるとき、 \mathcal{B} は $\mathcal{G}(T)$ の bottleneck であるという。図 5 は bottleneck の定義を表したものである。

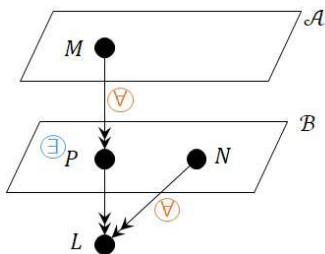


図 5: *bottleneck* の定義

3.4 boundary の定義

Venturini Zilli は無限個の頂点を持つ簡約グラフにおいて、無限回の簡約を繰り返した極限にあたる *boundary* を導入した [3]. 定義を以下に示す.

定義 7 (Ch(T) 上の順序関係)

1. 半順序集合 $(\mathcal{G}^0(T), \overset{*}{\rightsquigarrow})$ における無限簡約列 (ω -Chain) の集合を Ch(T) と書く.
2. $C, C' \in \text{Ch}(T)$, ここで

$$C : \mathcal{A}_0 \overset{*}{\rightsquigarrow} \mathcal{A}_1 \overset{*}{\rightsquigarrow} \mathcal{A}_2 \overset{*}{\rightsquigarrow} \dots$$

$$C' : \mathcal{B}_0 \overset{*}{\rightsquigarrow} \mathcal{B}_1 \overset{*}{\rightsquigarrow} \mathcal{B}_2 \overset{*}{\rightsquigarrow} \dots$$

ただし

$$\mathcal{A}_0 \equiv \mathcal{B}_0 \equiv T/\sim$$

である. このとき, 次の関係を定義する.

$$C \sqsubseteq C' \Leftrightarrow \forall n \exists m \mathcal{A}_n/\sim \overset{*}{\rightsquigarrow} \mathcal{B}_m/\sim.$$

$$C \approx C' \Leftrightarrow C \sqsubseteq C' \wedge C' \sqsubseteq C.$$

さらに, $C/\approx = \{C' \mid C' \approx C\}$.

3. $\overline{\mathcal{G}^0(T)} = \{C/\approx \mid C \in \text{Ch}(T)\}$.

$\overline{\mathcal{G}^0(T)}$ に現れる新しい要素, すなわち $\overline{\mathcal{G}^0(T)} \setminus \mathcal{G}^0(T)$ を *boundary* と呼ぶ.

4 定理と予想の反例

Venturini Zilli は凝縮簡約グラフが無限個の平面を含む場合, グラフは狭まる (無限個の *bottleneck* があるとき) か, 広がる (比較不能な平面を含むとき) かのいずれかだと考えた. そこでグラフの広がり方を検討するため, *bottleneck* と *boundary* の個数に関する以下の定理を示した [3].

定理 8 [3] $\mathcal{G}^0(T)$ が無限に多くの *bottleneck* を持つならば, $\mathcal{G}^0(T)$ の *boundary* は単一である.

以下に Venturini Zilli による定理 8 の証明 [3] を示す.

証明 9 [3] $\mathcal{G}^0(T)$ が無限個の *bottleneck* を持つとする. このとき, 任意の無限簡約列は全ての *bottleneck* を通過する. したがって任意の *boundary* の要素 $C_1, C_2 \in \text{Ch}(T)$ は無限回合流し, $C_1 \approx C_2$ である. よって *boundary* は単一である.

しかし、この定理は一般には成り立たない。証明9において、 $\mathcal{G}^0(T)$ 上の全ての無限簡約列は $\mathcal{G}^0(T)$ の全ての *bottleneck* を通過すると説明していた。しかしながら、2つの *bottleneck* 間に無限個の平面がある場合、全ての *bottleneck* を通過しない無限簡約列が存在する。

実際、 $T \equiv (\lambda x.I)\Omega_3\Delta_3\Delta_3$ とすると反例が得られる。図6は $\mathcal{G}^0(T)$ を表し、白頂点に*で *bottleneck* を示しており、無限に多くの *bottleneck* が含まれている。一方、図7は $\overline{\mathcal{G}^0(T)}$ を表している。Ch(T)の要素を以下に列挙する。

- $C_0 : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_1 : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_2 : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_3 : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3\Delta_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3\Delta_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} \dots$
- \vdots
- $C_\omega : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} (\lambda x.I)(\Omega_3\Delta_3\Delta_3)\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_{\omega+1} : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_{\omega+2} : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3\Delta_3 \xrightarrow{*} \dots$
- $C_{\omega+3} : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3\Delta_3\Delta_3 \xrightarrow{*} \dots$
- \vdots
- $C_{\omega-2} : (\lambda x.I)\Omega_3\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3 \xrightarrow{*} I\Delta_3\Delta_3\Delta_3 \xrightarrow{*} \dots$

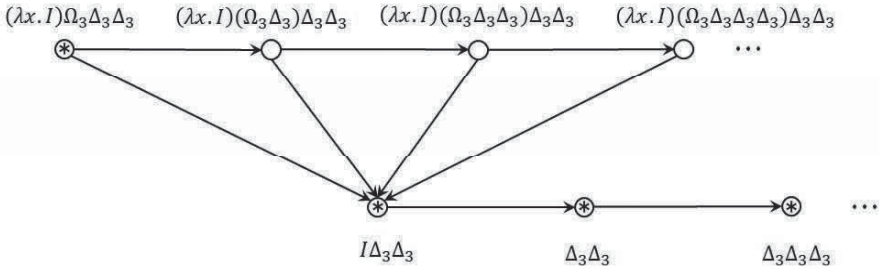


図 6: $\mathcal{G}^0((\lambda x.I)\Omega_3\Delta_3\Delta_3)$



図 7: $\overline{\mathcal{G}^0((\lambda x.I)\Omega_3\Delta_3\Delta_3)}$

$C_\omega, C_{\omega-2}$ は無限に次の平面へと向かう簡約列であり、他の要素は一つの平面に留まり続ける簡約列である。ここで $C_\omega, C_{\omega-2}$ が *boundary* に対応しており、図7中では黒頂点に*で *boundary* を示している。以上より *bottleneck* は無限個存在するが、*boundary* は2個現れるため、定理8の反例となる。

また、以下の予想もされている。

予想 10 [3]
 定理 8 の逆が成り立つ。

この予想に対しては $T \equiv (\lambda x.I)\Omega_3$ とすることで反例が得られる． $\text{Ch}(T)$ の要素を以下に示す．

$$\begin{aligned}
 C_0 &: (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)\Omega_3 \overset{*}{\circ} \dots \\
 C_1 &: (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3) \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3) \overset{*}{\circ} \dots \\
 C_2 &: (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3\Delta_3) \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3\Delta_3) \overset{*}{\circ} \dots \\
 C_3 &: (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3\Delta_3\Delta_3) \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3\Delta_3\Delta_3) \overset{*}{\circ} \dots \\
 &\vdots \\
 C_\omega &: (\lambda x.I)\Omega_3 \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3) \overset{*}{\circ} (\lambda x.I)(\Omega_3\Delta_3\Delta_3) \overset{*}{\circ} \dots \\
 C_{\omega+1} &: (\lambda x.I)\Omega_3 \overset{*}{\circ} I \overset{*}{\circ} I \overset{*}{\circ} \dots
 \end{aligned}$$

C_ω が *boundary* に対応しており，単一となっている（図 9）．しかし *bottleneck* は 2 個であるため（図 8），予想 10 に対する反例となる．

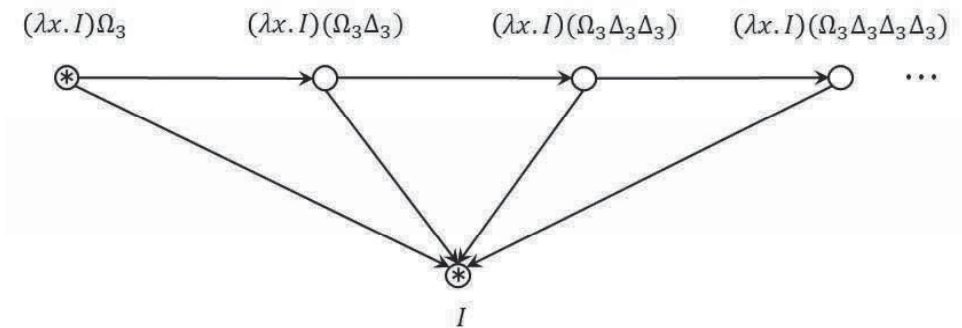


図 8: $G^0((\lambda x.I)\Omega_3)$



図 9: $\overline{G^0((\lambda x.I)\Omega_3)}$

5 順序型

Venturini Zilli [3] は次に示す *spectrum* を用いて λ 項の順序型を定義した．どの順序数が λ 項の順序型で表現できるか，という問題は未解決であり Intrigila ら [4] は ϵ_0 未満の任意の順序数が表現可能であることを示した．この節では Intrigila ら [4] に従って，順序型の構成方法について調査したことをまとめる．

5.1 spectrum の定義

定義 11 ($\text{Red}(T)$ 上の順序関係)

1. λ 項 T から始まるすべての有限及び無限の簡約列の集合を $\text{Red}(T)$ で表す．
2. $\mathcal{D}_1, \mathcal{D}_2 \in \text{Red}(T)$ ，ここで

$$\mathcal{D}_1 = T_1^{(0)} \rightarrow_\beta T_1^{(1)} \rightarrow_\beta T_1^{(2)} \rightarrow_\beta \dots$$

$$\mathcal{D}_2 = T_2^{(0)} \rightarrow_{\beta} T_2^{(1)} \rightarrow_{\beta} T_2^{(2)} \rightarrow_{\beta} \cdots$$

ただし

$$T_1^{(0)} \equiv T \equiv T_2^{(0)}$$

である。このとき、次の関係を定義する。

$$\mathcal{D}_1 \leq \mathcal{D}_2 \Leftrightarrow \forall n \exists m T_1^{(n)} \xrightarrow{*}_{\beta} T_2^{(m)}.$$

$$\mathcal{D}_1 \simeq \mathcal{D}_2 \Leftrightarrow \mathcal{D}_1 \leq \mathcal{D}_2 \wedge \mathcal{D}_2 \leq \mathcal{D}_1.$$

3. $\widetilde{\mathcal{D}} = \{\mathcal{D}' | \mathcal{D}' \simeq \mathcal{D}\}$, $\widetilde{\text{Red}}(T) = \{\widetilde{\mathcal{D}} | \mathcal{D} \in \text{Red}(T)\}$ とおく。

$$\widetilde{\mathcal{D}}_1 \leq \widetilde{\mathcal{D}}_2 \Leftrightarrow \mathcal{D}_1 \leq \mathcal{D}_2.$$

定義 12 ($\text{Spec}(T)$)

半順序集合 $(\text{Red}(T), \leq)$ を T の *spectrum* と言い、 $\text{Spec}(T)$ と表す。

例として Ω_3 の *spectrum* を考える。 Ω_3 の簡約経路を列挙すると、以下のようになる。

$$\mathcal{D}_n = \begin{cases} \Omega_3 & (n=0) \\ \mathcal{D}_{n-1} \rightarrow_{\beta} \Omega_3 \Delta_3^n & (n \geq 1) \end{cases}$$

$$\mathcal{D}_{\omega} = \Omega_3 \rightarrow_{\beta} \Omega_3 \Delta_3 \rightarrow_{\beta} \Omega_3 \Delta_3^2 \rightarrow_{\beta} \cdots \rightarrow_{\beta} \Omega_3 \Delta_3^n \rightarrow_{\beta} \cdots$$

すると $\widetilde{\text{Red}}(T) = \{\widetilde{\mathcal{D}}_0, \widetilde{\mathcal{D}}_1, \dots, \widetilde{\mathcal{D}}_{\omega}\}$ となり、 $\text{Spec}(T)$ を図 10 に示す。

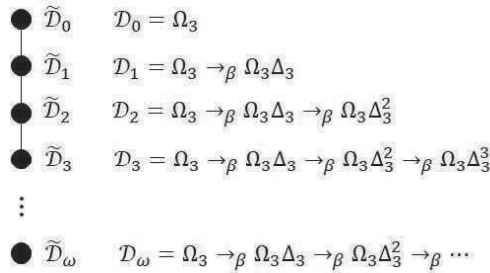


図 10: $\text{Spec}(\Omega_3)$

5.2 順序数

順序数とは自然数を拡張した概念であり、全順序集合の比較に用いられる。順序数は自分自身より小さい順序数の集合として定義される。以下に例を示す。

$$0 = \emptyset$$

$$1 = \{0\}$$

$$2 = \{0, 1\}$$

$$\omega = \{0, 1, 2, \dots\}$$

$$\omega + 1 = \{0, 1, 2, \dots, \omega\}$$

$\text{Spec}(T)$ が直線状になるとき、 $(\widetilde{\text{Red}}(T), \leq)$ は全順序集合となる。任意の全順序集合はある順序数と関係づけられており、その順序数のことを順序型という。例えば $\text{Spec}(\Omega_3)$ の要素は $\{\widetilde{\mathcal{D}}_0, \widetilde{\mathcal{D}}_1, \dots, \widetilde{\mathcal{D}}_{\omega}\}$ であり、順序型は $\omega + 1$ である。

5.3 順序関数

λ 項 T に複数の β レデックスが独立して現れる場合、 $\text{Spec}(T)$ に比較不能な要素が発生し、順序型を定義できない。そのため、複数の β レデックスが存在する場合、一方の β レデックスの内部に他の β レデックスが存在する必要がある。このような構造を持つ λ 項を得るため、以下の形式を用いる [4]。

定義 13 (*nf-representable*)

α を ω 以上の順序数、 \mathcal{N} を以下の式で与えられる β 正規形 N の集合とする。

$$N \equiv \lambda y.yN_1 \cdots N_k \quad (k \geq 0)$$

ただし N は自由変数を含まず、 $N_1 \cdots N_k$ は β 正規形である。 y は $N_1 \cdots N_k$ に含まれる場合もある。ある $N_\alpha \in \mathcal{N}$ が存在して $\text{Spec}(N_\alpha O)$ の順序型が α であるとき、 α は *nf-representable* であるという。ただし $O \equiv \lambda xy.y$ である。

例えば $N_{\omega+1} \equiv \lambda y.yI\Delta_3\Delta_3$ である。そして順序関数を以下で定義する。

定義 14 (順序関数)

F を順序数から順序数への関数とする。 F が *nf-representable* であるとは、以下を満たす λ 項 \underline{F} が存在することをいう。

1. \underline{F} は自由変数を持たない β 正規形である。
2. 全ての *nf-representable* な順序数 α および λ 項 N_α に対して、 $\underline{F}N_\alpha \rightarrow_\beta N_{F(\alpha)}$ と β 簡約される。

図 11 に、順序関数と λ 項の対応を示す。

$$\begin{array}{ccc} \lambda\text{項} & & \text{順序数} \\ N_\alpha & & \alpha \\ & & \downarrow \\ \underline{F}N_\alpha \rightarrow_\beta N_{F(\alpha)} & & F(\alpha) \end{array}$$

図 11: 順序関数と λ 項の対応

表 1 に基本的な順序関数をまとめる [4]。

順序関数	λ 項
$F(\alpha) = \alpha + 1$	$\underline{F} \equiv \lambda zy.y(yIzy)$
$F(\alpha) = \beta + \alpha + 1$	$\underline{F} \equiv \lambda zy.y(yIN_\beta y)(\lambda uvx.x(vu))yz$
$F(\alpha) = \alpha \cdot \omega + 1$	$\underline{F} \equiv \lambda wy.yIHHy, H \equiv \lambda xyz.z(y(Iwy)xy)$

順序関数 $F(\alpha) = \alpha \cdot \omega + 1$ を表す \underline{F} に対し $\text{Spec}(\underline{F}N_\alpha O)$ を示したものを図 12 に示す。ここで $H_\alpha \equiv H[w := N_\alpha]$ である。計算中、 β レデックスが 2 か所現れる箇所が存在する。内側の β レデックスを β 簡約し続けると順序数 α の構造が現れ、外側を β 簡約すると再度 β レデックスが 2 か所現れる。この構造が無限に繰り返されるため、 \underline{F} は順序関数 $\alpha \cdot \omega + 1$ を表現している。

$N_{\omega+1}$ に順序関数 $F(\alpha) = \alpha \cdot \omega + 1$ を適用すると N_{ω^2+1} が得られ、複数回繰り返すことで $N_{\omega^3+1}, N_{\omega^4+1} \cdots N_{\omega^n+1}$ が得られる。さらに、任意の順序関数 F に対して F^ω を

$$F^\omega(\alpha) := \sup_{n < \omega} \left(\sum_{i=0}^n F^n(\alpha) \right)$$

$$\begin{aligned}
& FN_\alpha O \rightarrow_\beta (\lambda y. yIH_\alpha H_\alpha y)O \rightarrow_\beta OIH_\alpha H_\alpha O \rightarrow_\beta IH_\alpha H_\alpha O \\
& \rightarrow_\beta H_\alpha H_\alpha O \rightarrow_\beta (\lambda yz. z(y(IN_\alpha y)H_\alpha H_\alpha y))O \\
& \rightarrow_\beta \lambda z. z(O(OIN_\alpha O)H_\alpha H_\alpha O) \xrightarrow{\alpha} \lambda z. z(O(OIN_\alpha O)H_\alpha H_\alpha O) \\
& \quad \downarrow \beta \quad \xrightarrow{\beta} \underline{(IN_\alpha O)} \rightarrow_\beta \underline{(N_\alpha O)} \rightarrow_\beta \dots \\
& \lambda z. z(IH_\alpha H_\alpha O) \rightarrow_\beta \lambda z. z(H_\alpha H_\alpha O) \\
& \rightarrow_\beta \lambda z. z((\lambda yz_1. z_1(y(IN_\alpha y)H_\alpha H_\alpha y))O) \\
& \rightarrow_\beta \lambda z. z(\lambda z_1. z_1(O(OIN_\alpha O)H_\alpha H_\alpha O)) \xrightarrow{\alpha} \lambda z. z(\lambda z_1. z_1(O(OIN_\alpha O)H_\alpha H_\alpha O)) \\
& \quad \downarrow \beta \quad \xrightarrow{\beta} \underline{(IN_\alpha O)} \rightarrow_\beta \underline{(N_\alpha O)} \rightarrow_\beta \dots \\
& \lambda z. z(\lambda z_1. z_1(IH_\alpha H_\alpha O)) \\
& \quad \vdots
\end{aligned}$$

図 12: $\text{Spec}(FN_\alpha O)$

と定義する。 F^ω は nf -representable であり [4]、これを用いることにより $N_{\omega^{\omega+1}}$ が得られる。

Intrigila らは $\omega^{\omega^{\dots}}$ までの順序数の構成方法を示しており [4]、以下の定理を提示した。

定理 15 [4]

ϵ_0 未満の任意の順序数は nf -representable である。

ここで ϵ_0 とは、 $\alpha = \omega^\alpha$ を満たす最小の α のことである。さらに以下の予想が述べられている。

予想 16 [4]

ϵ_0 は nf -representable である。

5.4 $\overline{\mathcal{G}^0(T)}$ と $\text{Spec}(T)$ の対応

3.2 節で述べた $\overline{\mathcal{G}^0(T)}$ と 5 節で述べた $\text{Spec}(T)$ は順序同型であることが知られている [3]。実際、写像 $\varphi: \text{Spec}(T) \rightarrow \overline{\mathcal{G}^0(T)}$ を以下のように定義すれば同型の順序が得られる。

$D \in \text{Spec}(T)$ とおく。

1. D が無限列の場合: $D = M_0 \rightarrow_\beta M_1 \rightarrow_\beta M_2 \rightarrow_\beta \dots, M_0 \equiv T$ において

$$\varphi(D) = C/\approx, \text{ ただし } C: M_0/\sim \xrightarrow{\circ^*} M_1/\sim \xrightarrow{\circ^*} M_2/\sim \xrightarrow{\circ^*} \dots$$

2. D が有限列の場合: $D = M_0 \rightarrow_\beta M_1 \rightarrow_\beta \dots \rightarrow_\beta M_n, M_0 \equiv T$ において

$$\varphi(D) = C/\approx, \text{ ただし } C: M_0/\sim \xrightarrow{\circ^*} M_1/\sim \xrightarrow{\circ^*} \dots \xrightarrow{\circ^*} M_n/\sim \xrightarrow{\circ^*} M_n/\sim \xrightarrow{\circ^*} M_n/\sim \dots$$

そして $\overline{\mathcal{G}^0(T)}$ と $\text{Spec}(T)$ に関して、次の考察が与えられる。

$\overline{\mathcal{G}^0(T)}$ において、boundary は無限簡約列 $M_0/\sim \xrightarrow{\circ^*} M_1/\sim \xrightarrow{\circ^*} M_2/\sim \xrightarrow{\circ^*} \dots$ である。従って、boundary に対応する $\text{Spec}(T)$ の要素も無限簡約列となる。特に $\text{Spec}(T)$ が全順序集合であるとき、 $\text{Spec}(T)$ の要素を小さい順に $\tilde{D}_0, \tilde{D}_1, \tilde{D}_2, \dots$ と順序数を割り当てると、 α が極限順序数のとき \tilde{D}_α は boundary に対応する。

以上より、 $\text{Spec}(T)$ の順序型が α ならば、 $\overline{\mathcal{G}^0(T)}$ の boundary は α 以下の極限順序数の数だけ存在すると推察される。

6 まとめと今後の課題

本研究では, Venturini Zilli[3] が提示した *bottleneck* と *boundary* の定理について, 反例を与えた. 定理の逆が成り立つという予想についても反例を与えた. さらに, $\overline{\mathcal{G}^0(T)}$ と $\text{Spec}(T)$ が同型であることに着目し, *boundary* の個数と順序型の対応について考察を与えた.

Venturini Zilli[3] が残した他の問題について, $\mathcal{G}^0(T)$ は局所有限であるという予想については Intrigila ら [5] が否定的に解決している. λ 項で表現できる順序型についての問題は, Intrigila ら [4] が ϵ_0 未満の順序数の構成方法を示した. ϵ_0 以上の順序数も表現可能であると予想されており, 今後の課題として λ 項による ϵ_0 の構成が挙げられる.

参考文献

- [1] H. P. Barendregt, “The Lambda Calculus. Its Syntax and Semantics”, North-Holland, revised edition, 1984.
- [2] N. B. B. Grathwohl, J. Ketema, J. D. Pallesen, and J. G. Simonsen, “Anagopos: A Reduction Graph Visualizer for Term Rewriting and Lambda Calculus”, in *Proc. 22nd International Conference on Rewriting Techniques and Applications*, pp. 61-70, 2011.
- [3] M. Venturini Zilli, “Reduction graphs in the lambda calculus”, *Theoretical Computer Science*, Volume 29, pp. 251-275, 1984.
- [4] B. Intrigila and M. Venturini Zilli, “Orders, reduction graphs and spectra”, *Theoretical Computer Science*, Volume 212, pp. 211-231, 1999.
- [5] B. Intrigila, A. R. Laurenzi, “Two Problems on Reduction Graphs in Lambda Calculus”, *Fundamenta Informaticae*, vol. 44, no. 1-2, pp. 133-144, 2000.