CRS とその代数構造: 操作的意味論の統一的な解析へ向けて

CRS and Its Algebraic Structure: Towards Unified Analysis of Operational Semantics

浜名誠
Makoto Hamana

群馬大学 情報工学科
Department of Computer Science, Gunma University
hamana@cs.gunma-u.ac.jp

Klop によって提案された高階書換えの体系コンビナトリー簡約系 (Combinatory Reduction Systems, CRSs) が Fiore，Plotkin，Turi らによる構文の代数モデル Σ モノイドによって健全かつ完全にモデル化できることを示す．このモデルを推移的関係を持つものに制限すると，停止性を持つ CRS の完全な特徴付けができる．このことを用いて，与えられた CRS の停止性を代数的な解釈によって示すという便利な手法を与える．

1 はじめに

データをある形から別の形へ変換するという操作は，計算機を用いることの重要な利点の一つであり，したがってどのようなソフトウェアを構成する際にも普遍的に表れることになる事象である．このような変換は「書換え」と言いかえることができ，この書換えのための形式的体系として項書換え系 (Term Rewriting System, TRS) が知られている [2]．TRS では関数記号と変数という二つの構文要素のみからなる「一階の項」という構造が，普遍的なデータ構造を表すことになる．

その普遍性により項書換え系は多くの応用を持つが，対象となるデータが「プログラム」やそれに準ずるようなデータの場合，一階の項では取り扱いが難しくなる．なぜならプログラムは必然的に，変数宣言に代表される変数束縛の機構を持ち，これは一階の項から外れる構文要素であるからである．実際，プログラム変換には二階の高階項を使った書換えが有効であることが示されている [28]．このようなデータは数理論理学や自動定理証明での変換操作にも表れ，限量を含む論理式の変換には，やはり変数束縛を持ったデータを書換え対象とする必要がある．

変数束縛の取り扱いの困難さは，束縛を持ったデータはλ 計算でいうところのα 変換及び β 変換による等価性を反映しなければならない点にある．これは明らかに一階の項のみでは不十分である．このような目的のために，高階書換え系という種類の TRS を拡張した体系が知られている [1, 12, 16, 4, 3]．これらの体系は，変数束縛のある項を直接書換えの対象にできるという強力さを持つが，一方，TRS ほど理論が整備されていない．特に与えられた高階書換え系の停止性を証明する手段に乏しい．

本論文では，特にコンビナトリー簡約系 (Combinatory Reduction System, CRS)[12] という高階書換え系を対象とし，これに対して，健全かつ完全な代数的なモデルを与える．このモデルはまた，高階書換え系の停止性を代数的な解釈によって証明するという便利な手法を与える．

本論文の実質的な貢献　例として限量を持つ論理式の変換の例を見てみる．例えば次の CRS $\mathcal{R}$ は冠頭標準系への変換，すなわち限量を外へ押し出す操作を行う．これは変数束縛の機構が本質的に必要な高階書換え系の典型的な一例である [23, 24]:

$$\begin{array}{ll} \mathrm{P} \wedge \forall(x.\mathrm{Q}[x]) & \to \forall(x.\mathrm{P} \wedge \mathrm{Q}[x]) \\ \neg\forall(x.\mathrm{Q}[x]) & \to \exists(x.\neg(\mathrm{Q}[x])) \\ \forall(x.\mathrm{Q}[x]) \wedge \mathrm{P} & \to \forall(x.\mathrm{P} \wedge \mathrm{Q}[x]) \\ \neg\exists(x.\mathrm{Q}[x]) & \to \forall(x.\neg(\mathrm{Q}[x])) \end{array}$$

さらに $\vee$ と $\exists$ についても同様の書換え規則がある．現在知られている高階書換えに対する停止性証明法をこの $\mathcal{R}$ に適用することは難しいと知られている．それは，考察する書換えの定義の変更が必要であったり [11]，複雑な関数空間を束縛の解釈に必要としたりという困難さである [23, 22]．この論文は，$\mathcal{R}$ のような CRS の停止性を証明するためのよりシンプルな方法を与える (例 8.2 を参照).

論文の構成　本論文の構成は以下の通りである．まず始めに 2 節で本研究の背景を述べる．そして 3 節で CRS の定義を概説する．4 節では CRS のうち，性質のよい形のものを与えるクラスとして**構造的**CRS という概念を導入する．5 節では CRS の構文の代数モデルを与える．6 節では CRS の書換えの代数モデルを与える．7 節では CRS のメタ書換えの代数モデルを与える．最後に 8 節で，本論文の結果を使った停止性証明の例を与える．

2　背景

本節では，本論文で与える高階書換え系と代数モデルの歴史的背景を述べる．

2.1　高階書換え系と高階抽象構文

最初の本格的な高階書換え系の研究は Aczel による Contraction Scheme と呼ばれる体系である [1]．これは束縛の構造についての意味論的概念を背景に持つ．これに影響を受けた形で，Klop はより制限を緩めた Combinatory Reduction System (CRS) の体系を提案し，TRS と λ 計算の理論の双方を援用し拡張する形で，CRS の性質を詳しく調べた [12]．CRS は一階項に束縛を追加した独自のメタ言語を用いている．

その後，Nipkow はより直接的な高階書換えのフォーマットとして型付き λ 計算をメタ言語として使い，項書換えを β 同値性の法の元で行う形として定式化した Higher-order Rewrite System (HRS) という高階書換え系を提案した．この体系は型付き λ 計算を備えているため，証明検証系，定理証明システムへの応用を意識したものとなっている．

より証明検証系，特に Coq への応用を強く意識して提案されたのが，Blanqui，Jouannaud，岡田らの体系で [4, 3]，より強力な型理論とλ 計算を備え，これに TRS の書換えを加えたものになっている．λ 計算で知られる技法を使い，停止性の証明のための技法を考案している．

現在の束縛機構を持つ高階項書換え系のフォーマットの主流は，これら CRS, HRS, Blanqui-Jouannaud-岡田の体系のいずれかと言ってよい．CRS と HRS の関係は [18] で詳しく調べられており，互いにシミュレート可能であることが分かっている．

さてこの流れとは別に，高階抽象構文 [19, 7] という技法がある．これは証明検証系において，束縛のあるデータ構造，特にオブジェクトレベルのλ 式，限量のある論理式や証明をコーディングする際に，証明検証系自身が持つλ 計算をメタ言語として用いると，α 変換とβ 変換のコーディングが必要なくなり有用であるという技法である．このアイデアは直感的ではあるが，高階抽象構文に対する帰納法の適用が困難であるという本質的な問題があった [7].

高階書換えと高階抽象構文 (Higher-Order Abstract Syntax, HOAS) は，同一の文脈で議論されることはこれまでなかったが，基本的なアイデアは同じであると言ってよい．特に HRS と HOAS は双方とも型付きλ 計算をメタ言語に用いる点で同じアイデアといえる．

高階抽象構文に帰納法の適用が困難である問題は，高階書換え系の分野においても別の形で表れており，それは HRS に対して停止性の証明のためにこれまで与えられていたモデル [22, 23] が完全ではないことが本質的に関連している．というのも，双方ともある関数空間をメタ言語のモデル化に使っているため，帰納法の適用が困難であり，また完全性のための項モデルの構成が不可能になっていた．したがって，高階書換えのための完全なモデルを探すためには，高階抽象構文のためのよいモデルがまず必要ということになる．

このような高階抽象構文の問題にブレークスルーを与えたのが Plotkin らによる束縛代数の理論 [8] で，始代数意味論という明解な数学的構造で高階抽象構文，つまり束縛を持つ構文に対する帰納法を確立している．

2.2　操作的意味論の代数モデルと束縛の導入

この Plotkin らによる束縛代数 [8] の理論は，その前年に書換え系の国際会議 RTA'98 でそのアイデアが発表されている [21]．この意味で，束縛代数の理論と高階書換え系に接点がある．また，束縛代数は CRS の元となった Aczel による仕事 [1] にも起源があるという接点も持つ．実際，本論文の高階書換えの意味論として束縛代数を用いるという最も基本的なアイデアはここから来ている．

Plotkin によるこの束縛代数の一連の研究の流れは，さらに遡ると彼による著名な操作的意味論のフォーマット SOS(Structural Operational Semantics) についての代数的研究にたどり着く．SOS は，プログラミング言語や並行計算系のの動作ステップを形式的に規定するための体系で，プログラミング言語の理

論的研究には必須の道具になっている．

Plotkin と Turi は，SOS のフォーマットの代数的意味論を用い，SOS で与えられる言語の表示的意味論と操作的意味論の一致を自動的に導出する結果を得ていた [26]．この研究では SOS のうち GSOS というフォーマットを対象としているが，構文として使われる言語は一階の構文で与えられるもので束縛の構文構造は入っていない．したがってここで用いられた普遍代数の構造も通常の Σ 代数 (及び余代数) だった．この結果を束縛のある構文を対象とするものに拡張することは，次段階として必要なことで，計画として述べられている [26]．なぜなら SOS の主な使用対象のプログラミング言語や並行計算系において，束縛の機構は本質的に必要な言語要素であるからである．

したがって，Plotkin らによる束縛代数の理論 [8] は，この束縛を持つ式を使う SOS の代数モデルの研究の準備段階として，束縛を持つ抽象構文を考察したと見ることができる．本研究は，このアイデアを SOS ではなく，書換え系に適用したときに，それが高階書換え系の代数モデルとして適切なものになることを示したものと位置づけることができる．

2.3 高階書換え系と操作的意味論

SOS で与えられる操作的意味論と，TRS に代表される書換え系の理論は，双方とも「変換ステップ」の関係を与えるという点では非常に似ているが，研究分野としては独立に互いの仕事が関連することなく進んできた．これらが関連することがなかった理由の一つとして，SOS は多様な「式」(束縛や代入の構文要素を許す) を対象としていたのに対して，TRS が一階の項に限っていた点がある．

TRS による書換えの理論は，これまでの研究により多くの有用な概念と結果を蓄積しており，これを複雑なエンコーディングなしに直接に操作的意味論の研究に役立たせることができれば新しいソフトウェアの実現のための基礎理論として有用であることは言を俟たない．高階書換えの停止性のためのモデルを与えるという実質的な目的とともに，より概念的な本論文の目的は，このような TRS の理論と SOS の理論を接近させるための一ステップとして，Plotkin らによる SOS とその代数的モデルの研究の流れと平行するものとして高階書換え系を捉え，そのモデルを与えることにある．

3 コンビナトリー簡約系

本節では Combinatory Reduction System (CRS)，コンビナトリー簡約系の定義を概説する．この定義は，標準的な文献 [13] に従うが，一部 [5] で使われている構文を用いる．

3.1 CRS の定義

シグネチャΣ とはアリティを持つ**関数記号**(F^l と書かれる) とアリティを持つ**メタ変数**(z^l のように small caps で書かれる) での集合である (ここで，上付き文字 $l \in \mathbb{N}$ がアリティを示している．必要のないときは略される)．

(i) CRS の**項**は以下の構文から成る:

$$t ::= x \mid x.t \mid F^l(t_1,\ldots,t_l).$$

これらの構文要素はそれぞれ，**変数**，**抽象**，**関数項**と呼ばれる．

(ii) CRS の**メタ項**は項を以下の構文に拡張したものである:

$$s ::= x \mid x.s \mid F^l(s_1,\ldots,s_l) \mid \mathrm{z}^l\,[s_1,\ldots,s_l].$$

最後の構文要素は**メタ適用**と呼ばれる．

(iii) **付値**(valuation) とは各メタ変数 z^n を項 t からつくられる**代替**(substitute) (メタレベルのラムダ記法．詳しくは [13] を参照) に以下のようにに写す関数のことである:

$$\theta : \mathrm{z} \longmapsto \underline{\lambda}(x_1,\ldots,x_n).t \tag{1}$$

付値はメタ項を項へ写す関数に拡張できる:

$$\begin{aligned} \theta(x) &= x \\ \theta(x.t) &= x.\theta(t) \\ \theta(F(t_1,\ldots,t_l)) &= F(\theta(t_1),\ldots,\theta(t_l)) \\ \theta(\mathrm{z}[t_1,\ldots,t_l]) &= \theta(\mathrm{z})\,(\theta(t_1),\ldots,\theta(t_l)) \end{aligned}$$

ここで最後の等式の右辺はメタレベルでの適用を行っていることに注意．付値は，任意のメタ変数 z, z’ について $\theta(\mathrm{z})$ が自由変数 x を含むときに $\theta(\mathrm{z}’)$ において同じ x が束縛されて現れることがない場合に**安全 (safe)** と呼ばれる．

(iv) CRS の**書換え規則**は $l \to r$ と書かれる．ここで l と r はメタ項で，次の条件を満たすものである:

(iv-a) l と r は自由変数を持たないメタ項

(iv-b) l は「パターン」すなわち，部分項のすべてのメタ適用が $\mathrm{z}[x_1,\ldots,x_n]$(ここで各 x_i は異なる) となっている関数項である

(iv-c) r 中のメタ変数はすべて l 中のメタ変数に含まれる．

書換え規則 $l \to r$ は，次のようなメタ変数 z を含まないときθ **に対して安全**という: 代替 $\theta(\mathrm{z})$ が自由変数 x を抽象 $x.-$ の内部に代入する．

(v) CRS の**書換え関係 (rewrite relation)**$\to_{\mathcal{R}}$ は与えられた CRS $\mathcal{R}$ の文脈と安全な付値による閉包で与えられる：

$$\frac{l \to r \in \mathcal{R}}{\theta(l) \to_{\mathcal{R}} \theta(l)}\ \mathrm{safe}\ \theta \qquad \frac{s \to_{\mathcal{R}} t}{x.s \to_{\mathcal{R}} x.t}$$

$$\frac{s \to_{\mathcal{R}} t}{F(\ldots,s,\ldots) \to_{\mathcal{R}} F(\ldots,t,\ldots)}$$

ここで $l \to r$ は安全な付値θ に対して安全でなければならない．三番目の規則は i 番目の F の引数の書換えを表している．

3.2 メタ概念の役割

一見すると，一階の項書換え系 TRS や λ 計算に比べて，CRS の定義はかなり複雑であるように見える．特に構文的要素において，変数に加えてメタ変数，項に加えてメタ項，そしてアリティの概念を加味した代替の存在など，見慣れない要素が多くある．

しかしこの CRS の定義は，CRS の書換え関係のつくり方を見ると自然であることが分かる．CRS では書換え規則はスキーマであって，実際の書換えは項上の関係に具体化されており，メタ変数は出現しない．すなわち，CRS の書換え関係は項上であって，メタ項上ではない．より模式的には以下のようになる．

書換え規則	メタレベル
書換えステップ	オブジェクトレベル

付値はこのメタレベルからオブジェクトレベルへの具体化を行う写像である．一見複雑に見えるメタの名が付く構文要素はこのような区分けをすると分かりやすい．

さらに特筆すべきことは，この CRS が成す構造の数学的な自然さである．特に CRS のメタ項全体が，自由 Σ モノイドという構造を成すという明解な構造を持っている (定理 5.4)．本論文で与える CRS の代数的意味論はこの部分を出発点にしている．したがって，CRS は単に TRS へ束縛機構をアドホックに追加したものではない．

4 構造的 CRS

この節では，CRS のうち，性質のよい形のものを与えるクラスとして**構造的**CRS という概念を導入する．構造的 CRS は，束縛シグネチャと呼ばれる形のシグネチャからつくられるメタ項から成る CRS のことをいう．束縛シグネチャとは，各関数記号についていくつのバインダを取るかの情報のアリティがあるシグネチャである．

より形式的には，**束縛シグネチャ**(binding signature)Σ は関数記号の集合 Σ とアリティ関数 $a : \Sigma \to \mathbb{N}^*$ から成るものである．アリティ $\langle i_1,\ldots,i_l\rangle$ を持つ関数記号 は $F : \langle i_1,\ldots,i_l\rangle$ と書かれ，これは l 個の引数を持ち k 番目の引数において i_k 個の変数を束縛することを表す $(1 \le k \le l)$.

通常は，CRS の束縛変数には λ 計算で通常仮定する α 同値性を仮定するが，束縛変数の名前換えや項の α 同値類を考えることは議論が煩雑になるため，以降我々は，(束縛/自由) 変数の名前のつけ方に対して de Bruijn level [6, 14, 8](また [20] p.81 も参照) の方法を用いることにする．メタ変数に対しては通常の名前付きの方法を用いる．また，表記法として自然数 n を集合 $\{1,\ldots,n\}$ ($n = 0$ を許す) を表すものと約束する．この de Bruijn level の方法の元でこの n は 1 から n の変数の集合を表すことになる．

定義 4.1 (メタ) 項 t が束縛シグネチャ Σ からつくられ，また Σ によって規定されている各関数記号の束縛アリティに従っているとき，この t を**構造的**と呼ぶ．

したがって，構造的メタ項は以下の形を持つことになる:

$$t ::= x \mid F(x_1\cdots x_{i_1}.t_1,\ldots,x_1\cdots x_{i_l}.t_l) \\ \mid \mathrm{z}^l\,[t_1,\ldots,t_l]$$

ここで F は束縛アリティ$\langle i_1,\ldots,i_l\rangle$ を持つ．

また，構造的メタ項は以下のように構成できる．まず $\mathbb{N}$-添字付き集合 Z を以下のように定義する:

$$Z(l) \triangleq \{\mathrm{z} \mid \mathrm{z}\ はアリティ\ l\ を持つ\ \}.$$

$$
\begin{array}{lll|lll}
\mathsf{CPS}(\mathrm{E}) & & \rightarrow \lambda k.(\![\mathrm{E}]\!)\,{}^{-}(\overline{\lambda}m.km) & \mathsf{CPS}(\mathrm{E}) & & \rightarrow \lambda 1.(\![\mathrm{E}]\!)\,{}^{-}(\overline{\lambda}2.12) \\
(\![\mathrm{V}]\!) & & \rightarrow \overline{\lambda}k.k\,{}^{-}\,\mathrm{V} & (\![\mathrm{V}]\!) & & \rightarrow \overline{\lambda}1.1\,{}^{-}\,\mathrm{V} \\
(\![\lambda x.\mathrm{E}[x]]\!) & & \rightarrow \overline{\lambda}k.k\,{}^{-}(\lambda x.\lambda k.(\![\mathrm{E}[x]]\!)\,{}^{-}(\overline{\lambda}m.km)) & (\![\lambda 1.\mathrm{E}[1]]\!) & & \rightarrow \overline{\lambda}1.1\,{}^{-}(\lambda 2.\lambda 3.(\![\mathrm{E}[2]]\!)\,{}^{-}(\overline{\lambda}4.34)) \\
(\![\mathrm{E}_0\mathrm{E}_1]\!) \rightarrow \overline{\lambda}k. & & (\![\mathrm{E}_0]\!)\,{}^{-}(\overline{\lambda}m.(\![\mathrm{E}_1]\!)\,{}^{-}(\overline{\lambda}n.mn(\lambda a.k\,{}^{-}a))) & (\![\mathrm{E}_0\mathrm{E}_1]\!) \rightarrow \overline{\lambda}1. & & (\![\mathrm{E}_0]\!)\,{}^{-}(\overline{\lambda}2.(\![\mathrm{E}_1]\!)\,{}^{-}(\overline{\lambda}3.23(\lambda 4.1\,{}^{-}4)))
\end{array}
$$

図 1: CRS の例: 値呼び CPS 変換 $\mathcal{S}$

このときメタ項 t は以下の規則を用いて $n \vdash t$ が導出できたとき構造的である．

$$
\frac{x \in n}{n \vdash x} \qquad \frac{\mathrm{Z} \in Z(l) \quad n \vdash t_1 \ \cdots \ n \vdash t_l}{n \vdash \mathrm{Z}[t_1, \ldots, t_l]}
$$

$$
\frac{F : \langle i_1, \ldots, i_l \rangle \in \Sigma \quad n{+}i_1 \vdash t_1 \ \cdots \ n{+}i_l \vdash t_l}{n \vdash F(\, n{+}1 \ldots n{+}i_1.t_1,\ \ldots,\ n{+}1 \ldots n{+}i_l.t_l \,)}
$$

これらの規則を用いて導出したメタ項は常に de Bruijn level の方法に従うものになる．またこのときの $n \vdash t$ を n の元での構造的メタ項と呼ぶ．最初の二つの規則のみ用いれば，n の元での**構造的項**を得る．

「構造的」という概念は書換え規則，CRS，付値に対しても明らかに拡張できる．具体的には以下の通りである．l と r を n の元での構造的メタ項とする．このとき書換え規則 $l \rightarrow r$ を構造的と呼ぶ．また，すべての書換え規則が構造的な CRS を，構造的 CRS と呼ぶ．

定義 4.2 付値 θ は，この写像による割り当て

$$\theta : \mathrm{Z} \mapsto \underline{\lambda}(x_1, \ldots, x_n).t$$

すべてに対して t がいつも構造的項で，かつ，t 中のすべての変数が $x_1, \ldots, x_n$ に含まれるとき，**構造的**と呼ばれる．

構造的 CRS は実際的な見地から見ても妥当な定義といえる．というのも，これまで高階書換え系の文献の中で議論されてきた具体的な CRS の例のほとんどすべては，構造的 CRS であるからである．例えば，Raamsdonk による高階書換え系の例のリスト [24] においてすべての CRS は構造的である．すなわち，与えられた「プレーンな」CRS に対して，適切な束縛シグネチャを簡単に見つけることができる．

例 4.3 (**CPS 変換**) 構造的 CRS のフォーマットは我々が計算機科学や数理論理学で用いるメタ言語と極めて類似している．構造的 CRS の一つの例は，導入で与えた冠頭標準系への変換 $\mathcal{R}$ である．ここではもう一つ別の例としてプログラム言語の理論と関係する例を挙げる．図 1 の CRS $\mathcal{S}$ は call-by-value CPS 変換を行うものである [5].

ここでメタ変数の集まり $Z = \{\mathrm{V}^0, \mathrm{E}^1, (\mathrm{E}_0)^0, (\mathrm{E}_1)^0\}$ と次の関数記号からなる束縛シグネチャを考える: $\lambda, \overline{\lambda} : \langle 1 \rangle,\ (-\ -), (-\,{}^{-}\,-) : \langle 0, 0 \rangle,\ \mathsf{CPS}, (\![-]\!) : \langle 0 \rangle$. 図 1 は CPS 変換の構造的 CRS $\mathcal{S}$ を二つの記法で与えたものである．左の列は通常の名前付き変数の記法で書かれた CRS，右の列は de Bruijn level の記法で書かれた CRS でこちらが我々が本論文で取り扱う記法である．

ポイントは，de Bruijn level の記法の方は単に変数名を (de Bruijn の) level に対応する「数で表した変数名」に置き換えたものに過ぎないということである．これはよりよく使われている de Bruijn *indexes* は異なるものであることに注意する必要がある．de Bruijn indexes においては，index は相対的に変数を表すものであって直接，変数名としてみなすことはできないが，de Bruijn level においては，level は項全体で大域的であるためそれを絶対的な変数名とみなせることができる点が異なる．これにより，de Bruijn level で書かれたメタ項はα 同値なメタ項の「正規形」として捉えることができる (例えば $\overline{\lambda}k.k\,{}^{-}\,\mathrm{V} =_\alpha \overline{\lambda}1.1\,{}^{-}\,\mathrm{V}$).

さて，構造的 CRS $\mathcal{S}$ は停止性を持つか，すなわち，$\mathcal{S}$ を用いた書換えが常に停止するかを考えたい[1]. 直感的には，この停止性は明らかであるように思える．なぜなら $(\![-]\!)$ は再帰的にλ 項を分解するからである．この論文では，与えられた CRS の停止性を証明するための意味論的手法を書換えの代数的特徴付けから導き出す．どのようにこの $\mathcal{S}$ が形式的に停止性を持つと示せるかは本論文の最後の例 8.3 で与えられる．

[1] この CRS はβ 簡約は含んでいない．単に変換をするだけである．

5 構文の代数的意味

この節と次の節では CRS の代数的意味論を展開する．CRS に対するモデルの必要性は，CRS の最も標準的な文献 [13] で，未解決問題として述べられていた．その後 CRS の研究は構文的な書換えの面は発展したが，モデルについては長く発見されていなかった．本論文はこれに対して，CRS に対する代数的モデルを与えるものである．基本的なアイデアは，Zantema [27] によって広く知られるようになった単調 Σ 代数による TRS の代数的意味論に沿う形になる．しかし明らかに通常の一階の Σ 代数では λ バインダなどの概念を解釈するには不十分である．我々は CRS の構文を Fiore, Plotkin and Turi による**束縛代数**の枠組み [8] で考察する．

5.1 準備: 圏 $\mathbf{Set}^{\mathbb{F}}$ とは

$\mathbb{F}$ を有限の順序数 $n = \{1, \ldots, n\}$ ($n = 0$ を許す) を対象，その間のすべての関数 $m \to n$ を射とする圏とする．この圏は n を de Bruijn level の方法で表した変数の集合と考えれば，すなわち変数の集合とその間の名前換え (代入) がつくる圏である．本論文で中心的に用いるのは圏 $\mathbf{Set}^{\mathbb{F}}$ である．これは関手 $\mathbb{F} \to \mathbf{Set}$ を対象，その間の自然変換を射とする関手圏である．この圏の対象 $A \in \mathbf{Set}^{\mathbb{F}}$ を**前層**(presheaf) と呼ぶ．前層 A は，荒く言えば $\mathbb{N}$-添字付き集合 $A(n)$ と考えてよい (正確には以下を参照).

CRS の代数的意味論ではこの圏 $\mathbf{Set}^{\mathbb{F}}$ の中で展開される．したがってこの副節では圏論に不馴れな読者に便利なようにこの圏と，これと関連する写像をより基本的な用語で説明する．本論文で与える CRS の意味論は，圏論的というよりも，TRS の停止性のための解釈で使われる通常の普遍代数を用いた代数的意味論 [27] に近い．したがってこのモデルを理解するためには圏論的知識はそれほど高度なものは必要ない．

圏 $\mathbf{Set}^{\mathbb{F}}$ は圏であるから，**対象**の集合と**射**の集合から成る．この圏の対象は関手 $\mathbb{F} \to \mathbf{Set}$ である．この圏の射はこれらの関手の間の自然変換である．これをより基本的な言葉で言い換えると，$\mathbf{Set}^{\mathbb{F}}$ の対象 A(しばしば $A \in \mathbf{Set}^{\mathbb{F}}$ と書かれる) は $\mathbb{N}$-添字付き集合 $\{A(n)\}_{n\in\mathbb{N}}$ でさらに，各関数 $\rho : m \to n$ に対して，いつも $A(\rho) : A(m) \longrightarrow A(n)$ が定義されている構造である．

そして対象 $A, B \in \mathbf{Set}^{\mathbb{F}}$ 間の射 $f : A \to B$ とは，$n \in \mathbb{N}$ がパラメタとなっている以下の形の関数の族

$$\{f(n) : A(n) \longrightarrow B(n)\}_{n\in\mathbb{N}}$$

で，すべての関数 $\rho : m \to n$ について条件

$$\forall a \in A(m) \,.\, B(\rho)(f(m)(a)) = f(n)(A(\rho)(a))$$

を満たすものである．この条件は「自然性」と呼ばれるが，図式的に書くと以下の図式の可換性 (上側と下側を辿る二つの関数が等しい) を意味する．

$$\begin{array}{ccc}
m & & A(m) \xrightarrow{\quad f(m) \quad} B(m) \\
\rho \big\downarrow & & A(\rho) \big\downarrow \qquad\qquad \big\downarrow B(\rho) \\
n & & A(n) \xrightarrow[\quad f(n) \quad]{} B(n)
\end{array}$$

これらを非常に荒く言うと，$A \in \mathbf{Set}^{\mathbb{F}}$ は $\mathbb{N}$-添字付き集合で，加えて「ある構造」を持っている．また $\mathbf{Set}^{\mathbb{F}}$ の射 $f : A \longrightarrow B$ は $\mathbb{N}$-添字付き関数 $f(n) : A(n) \longrightarrow B(n)$ で「ある条件」を満たすものである．

これらの「ある〜」の部分は (不可欠であるが) 大まかな理解のためには無視してもよい．本論文で圏 $\mathbf{Set}^{\mathbb{F}}$ を用いる理由の最も大きなものは，$\mathbb{N}$ による indexing の必要性である．これは，indexing により自由変数の集合による構造のパラメタ化を意図し，さらにこれによって束縛変数の表現を可能にするためである．

5.2 束縛代数

Fiore,Plotkin,Turi による束縛を持つ構文の圏論的代数による意味論 [8] を簡単に概説する．

関手 $\delta : \mathbf{Set}^{\mathbb{F}} \to \mathbf{Set}^{\mathbb{F}}$ を

$$(\delta A)(n) = A(n + 1)$$

で定義する．与えられた束縛シグネチャΣ に対して，**シグネチャ関手**$\Sigma : \mathbf{Set}^{\mathbb{F}} \to \mathbf{Set}^{\mathbb{F}}$ とは

$$\Sigma A \triangleq \coprod_{F:\langle i_1,\ldots,i_l\rangle\in\Sigma} \prod_{1\le i\le l} \delta^{i_i} A$$

で定義されるものである．Σ **代数**は組 (A, α) で前層 A と**代数構造**と呼ばれる射 $\alpha = [F_A]_{F\in\Sigma} : \Sigma A \to A$ から成る．ここで F_A は 各 $f : \langle i_1, \ldots, i_l\rangle \in \Sigma$ に対して定義されてる**操作**$F_A : \delta^{i_1} A\times\ldots\times\delta^{i_l} A \to A$ である．特に前層 $\mathrm{V} \in \mathbf{Set}^{\mathbb{F}}$ を $\mathrm{V}(n) \triangleq n$ と定義し，**変数の前層**と呼ぶ．

命題 5.1 ([**8**]) 組 $(\mathbf{Set}^{\mathbb{F}}, \bullet, \mathrm{V})$ はモノイダル圏 [15] をなす．ここで「代入」のモノイダル積 $\bullet$ は以下のように与えられる．

$$(A \bullet B)(n) \triangleq (\coprod_{m\in\mathbb{N}} A(m)\times B(n)^m)/\sim \qquad (2)$$

$\sim$ は以下から生成される同値関係である．

$$(t; u_{\rho 1}, \ldots, u_{\rho m}) \sim (A(\rho)(t); u_1, \ldots, u_l)$$

$(\rho : m \to l \in \mathbb{F})$．ここで，$A(m)\times B(n)^m$ の要素を $(t; u_1, \ldots, u_m)$ と書いている．

定義 5.2 Σ をシグネチャ関手とする．Σ **モノイド**とはモノイダル圏 $(\mathbf{Set}^{\mathbb{F}}, \bullet, \mathrm{V})$ 中の**モノイドオブジェクト** [15] $M = (M, \eta : \mathrm{V} \to M, \mu : M \bullet M \to M)$ でかつ Σ 代数構造 $\alpha : \Sigma M \to M$ を持ち，以下を可換にするものである:

$$\begin{tikzcd}
\Sigma(M) \bullet M \arrow[r] \arrow[d, "\alpha \bullet M"'] & \Sigma(M \bullet M) \arrow[r, "\Sigma\mu"] & \Sigma M \arrow[d, "\alpha"] \\
M \bullet M \arrow[rr, "\mu"'] & & M
\end{tikzcd}$$

Σ モノイドの射を，Σ 代数の準同型射 $(M, \alpha) \to (M', \alpha')$ でモノイド射になっているものと定義する．

5.3 構造項の代数

構造的項全体とメタ項全体はそれぞれよい代数的構造を持つ．自由変数の集合 n の下でのすべての構造的項からなる前層を

$$T_\Sigma \mathrm{V}(n) = \{t \mid n \vdash t,\ t \text{ は項 }\}$$

と定義する．また，$\mathbf{Set}^{\mathbb{F}}$ の射 $\nu : \mathrm{V} \longrightarrow T_\Sigma \mathrm{V}$ を

$$\nu(n) : \mathrm{V}(n) \longrightarrow T_\Sigma \mathrm{V}(n),\ x \longmapsto x.$$

と定義する．以下 $n+1, \ldots, n+k.t$ を $n+\vec{k}.t$ と略記することにする．アリティ $\langle i_1, \ldots, i_l\rangle$ を持つ関数記号 $F \in \Sigma$ に対して $\mathbf{Set}^{\mathbb{F}}$ の射 $F_T : \delta^{i_1} T_\Sigma \mathrm{V} \times \cdots \times \delta^{i_l} T_\Sigma \mathrm{V} \longrightarrow T_\Sigma \mathrm{V}$ を

$$(t_1, \ldots, t_l) \longmapsto F(n+\vec{i_1}.t_1, \ldots, n+\vec{i_l}.t_l)$$

と定義する．

定理 5.3 構造的項の前層 $T_\Sigma \mathrm{V}$ は始 $\mathrm{V} + \Sigma$ 束縛代数を成す．

証明 [8] に負う．([8] Theorem 2.1) の構文的代数 (syntactic algebra) は，$\mathrm{V} + \Sigma$ 代数 $(T_\Sigma \mathrm{V}, [\nu, [F_{T_\Sigma}]_{F\in\Sigma}])$ に他ならない． □

さらに Z を任意の $\mathbb{N}$-添字付き集合とする (4 節を参照)．**メタ項の前層** $M_\Sigma Z$ を以下で定義する．

$$M_\Sigma Z(n) = \{t \mid n \vdash t\}.$$

このとき 変数の代入の操作を行う**乗算**と呼ばれる $\mathbf{Set}^{\mathbb{F}}$ 中の射 $\beta : M_\Sigma Z \bullet M_\Sigma Z \longrightarrow M_\Sigma Z$ がある [29].

定理 5.4 構造的メタ項 $M_\Sigma Z$ は $\hat{Z}$ 上の自由 Σ モノイドを成す．ここで $\hat{Z}(n) = \coprod_{k\in\mathbb{N}} \mathbb{F}(k, n)\times Z(k)$.

証明 [29] に負う．$\hat{Z} \in \mathbf{Set}^{\mathbb{F}}$ に対して，[29] で構成した自由 Σ モノイドは記法の相違を同一視すれば $(M_\Sigma Z, [F_{M_\Sigma}]_{F\in\Sigma}, \nu, \beta)$ に他ならない．すなわち [29] における各構文要素 $\mathsf{ovar}(x)$，$[n]t$，$\lceil \mathrm{z}\rceil\langle t_1, \ldots, t_l\rangle$ を本論文の x，$n.t$，$\mathrm{z}[t_1, \ldots, t_l]$ と同一視すればよい．ここで演算子 F_{M_Σ} は F_{T_Σ} と同様に定義する． □

5.4 付値の代数的特徴付け

定義 5.5 割り当て(assignment) $\phi : Z \longrightarrow A$ とは $\mathbf{Set}^{\mathbb{F}}$ の射で A が Σ モノイド構造 (A, ν, β) を持つもののことを言う．

ここで注意すべき事は，上の定義の Z は $\mathbf{Set}^{\mathbb{F}}$ の前層であるということである．よって単なるメタ変数の $\mathbb{N}$-添字付き集合 X ではこの表現での割り当ての定義域として用いることはできない．しかし幸運なことに，我々は常に $\mathbb{N}$-添字付き集合 X から前層を $\hat{X}(n) \triangleq \coprod_{k\in\mathbb{N}} \mathbb{F}(k, n)\times X(k)$ と定義することにより得ることができる ([29]5.2 節)．ゆえに以後は記法上の約束として，割り当て中に単に $\mathbb{N}$-添字付き集合 X と書いてその前層版 $\hat{X} \in \mathbf{Set}^{\mathbb{F}}$ を表すこともある．

割り当て ϕ は 図 2 のやり方により Σ モノイド射 $\phi^* : M_\Sigma Z \longrightarrow A$ に拡張できる．特に $A = T_\Sigma \mathrm{V}$ と置いたとき

命題 5.6 割り当て $\theta : Z \longrightarrow T_\Sigma V$ は構造的付値であり，また，$\theta^* : M_\Sigma Z \longrightarrow T_\Sigma \mathrm{V}$ はそのメタ項上への拡張である．

$$\begin{aligned}
\phi^* : M_\Sigma Z(n) &\longrightarrow A(n) \\
x &\longmapsto \nu(n)(x) \qquad (x \in n) \\
F(n{+}\vec{i_1}.t_1, \ldots, n{+}\vec{i_l}.t_l) &\longmapsto F_A(n{+}\vec{i_1}.\phi^*(n{+}i_1)(t_1), \ldots, n{+}\vec{i_l}.\phi^*(n{+}i_l)(t_l)) \\
\mathrm{z}[t_1, \ldots, t_l] &\longmapsto \beta(n)(\,\phi(l)(\mathrm{z}); \phi^*(n)(t_1), \ldots \phi^*(n)(t_l)\,)
\end{aligned}$$

図 2: ϕ を拡張した Σ モノイド射 ϕ^*

これがなぜかを見るために，初めに割り当て θ は写像の族 $\theta(n) : Z(n) \longrightarrow T_\Sigma\mathrm{V}(n)$ で次のように値を与えるものであることに注意する．

$$\theta(n) : \mathrm{z} \longmapsto t \in T_\Sigma\mathrm{V}(n).$$

つまり，これは n 引数メタ変数 z をある n の元での構造項 t に写すものである．構造的付値の定義とこの定義を見比べ，また代替 $\underline{\lambda}(x_1, \ldots, x_n).t$ を $t \in T_\Sigma\mathrm{V}(n)$ である (なぜなら θ は構造的であるから) とみなすと，両者の定義は一致することがわかる．ゆえに，以後は「付値」の語をこの意味で用いる:

定義 5.7　付値(valuation) は項の Σ モノイドへの割り当て $\theta : Z \longrightarrow T_\Sigma\mathrm{V}$ である．また同様に，**メタ付値**とは，メタ項の Σ モノイドへの割り当て $\theta : Z \longrightarrow M_\Sigma X$ である．

θ^* は付値 θ をある意味で「準同型」的に拡張したものある．同様の主張は，CRS を扱う文献 [13, 18, 5, 17] でよく述べられるが，どのような正確な意味で「準同型」的なのかは，通常は詳しく説明されていない．我々の枠組みではこれを正確に表すことができる．すなわち，θ^* は Σ **モノイド射**である．つまり Σ 代数構造を保存し (したがって Σ 準同型射) とさらにモノイド構造も保存する写像である．

5.5　構造的付値の十分性

元来の CRS の付値の定義 (3 節) では写像 $\theta : \mathrm{z} \mapsto \underline{\lambda}(x_1, \ldots, x_n).t$ で，ここで t は**任意の**項であった．したがって当然 t は $x_1, \ldots, x_n$ 以外の自由変数を含む可能性がある．一方構造的付値の場合は，これと異なり t 中の変数は必ず $x_1, \ldots, x_n$ からのみ取られる．しかし，書換え規則中のメタ変数のアリティを拡大する操作を行うと構造的付値のみを使って CRS の項上の書換え関係を生成しても，元の (構造的でない) 付値を使って生成した書換え関係と同じになる．これを以下に示す．

二つの自然数 $m \leq m'$ について関数 $\rho : \mathbb{N} \to \mathbb{N}$ を $\rho(m{+}i) \triangleq m'{+}i$ $(i \in \mathbb{Z})$ と定義する．$\mathbb{N}$-添字付きのメタ変数の集合 $Z' = Y \cup \{\mathrm{z}^m\}$, $Z = Y \cup \{\mathrm{z}^{m'}\}$, $\mathrm{z} \notin Y$ を仮定する．**メタ変数 z のアリティの ρ による m から m' への拡大** とは以下のように定義されるメタ項上の関数である．

$$\begin{aligned}
\rho_\mathrm{z}(x) &= \rho(x) \qquad (x \in \mathbb{N}) \\
\rho_\mathrm{z}(n.t) &= \rho(n).\rho_\mathrm{z}(t) \\
\rho_\mathrm{z}(F(\vec{t})) &= F(\rho_\mathrm{z}(\vec{t})) \\
\rho_\mathrm{z}(\mathrm{z}[t_1, \ldots, t_m]) &= \mathrm{z}[1, \ldots, m' - m, \rho_\mathrm{z}(t_1), \ldots, \rho_\mathrm{z}(t_m)].
\end{aligned}$$

記法 5.8　以後，しばしば記法 $Z|n \vdash s \to t$ を書換え規則や書換えステップに用いる．これは $s \to t$ に表れるメタ変数と変数がそれぞれ Z と n に含まれることを表す．また，一方の情報が重要でないときは，単純に $Z \vdash s \to t$ か $n \vdash s \to t$ と書くこともある．

$\mathcal{R}$ を構造的 CRS とする．このとき $\mathcal{R}$ の**拡大閉包**とは $\mathcal{R}^\circ$ と書かれ，次の規則で生成される書換え規則の集合である:

$$\frac{l \to r \in \mathcal{R}}{l \to r \in \mathcal{R}^\circ} \qquad \frac{Y \cup \{\mathrm{z}^m\} \vdash l \to r \in \mathcal{R}^\circ}{Y \cup \{\mathrm{z}^{m+j}\} \vdash \vec{j}.\rho_\mathrm{z} l \to \vec{j}.\rho_\mathrm{z} r \in \mathcal{R}^\circ}$$

ここで，$z \notin Y$ で ρ_z はメタ変数 z のアリティの m から m' への拡大 である ($j \in \mathbb{N}$ は任意)．この意図は，元のメタ変数 z^m は m 個までの変数を含む項で置き換えられるものだったが，拡大を行うと $m + j$ 個までの変数を含む項で置き換えられるようになることを表している．

このとき，書換え規則の生成は以下のように再定

式化できる:

$$\frac{Z \vdash \vec{n}.l \to \vec{n}.r \in \mathcal{R}}{n \vdash \theta^*(n)(l) \Rightarrow_{\mathcal{R}} \theta^*(n)(r)}$$

$$\frac{n+i \vdash s \Rightarrow_{\mathcal{R}} t}{n \vdash F(\ldots, n+\vec{i}.s, \ldots) \Rightarrow_{\mathcal{R}} F(\ldots, n+\vec{i}.t, \ldots)}$$

ここで $\theta : Z : \longrightarrow T_\Sigma \mathrm{V}$ は構造的付値.

命題 5.9 de Bruijn level の方法で書かれた構造的 CRS $\mathcal{R}$ に対して，元の定義 (3節参照) と $\mathcal{R}^\circ$ を用いた上記の定義は，構造的項上の同じ書換え関係を生成する．すなわち，構造的項 s, t について，$s \Rightarrow_{\mathcal{R}^\circ} t$ と $s \to_{\mathcal{R}} t$ は同値.

証明 書換えステップの導出木についての帰納法による. □

6 書換えの代数的意味

この節では，構造的 CRS の書換え規則を Σ 代数で解釈し，この枠組みで完全な特徴付けを行う．**以後，この論文では構造的 CRS のみを取り扱う．**よって，構造的 CRS を特に断らずに単に「CRS」と述べる．

前層 A に対して，$>_A$ を推移的関係 (transitive relation) の族 $\{>_{A(n)}\}_{n\in\mathbb{N}}$ として表す．ここで，$>_{A(n)}$ は集合 $A(n)$ 上の推移的関係である．本論文では，以下の意味での単調性の概念を用いる [27].

定義 6.1 $(A_1, >_{A_1}), \ldots, (A_l, >_{A_l}), (B, >_B)$ を推移的関係を持つ前層とする．$\mathbf{Set}^{\mathbb{F}}$ の射 $f : A_1 \times \cdots \times A_l \longrightarrow B$ は，すべての $a_1, b_1 \in A_1(n), \ldots, a_l, b_l \in A_l(n)$ について，ある k について $a_k >_{A(n)} b_k$ で他の $j \neq k$ については $a_j = b_j$ であるなら $f(n)(a_1, \ldots, a_l) >_{B(n)} f(n)(b_1, \ldots, b_l)$ であるとき，**単調**と呼ばれる.

以下，書換え規則 を V+Σ 代数で解釈する．ここで，V+Σ 代数とは，節 5.3 の束縛代数の定義に従い，シグネチャ関手 $\mathrm{V}{+}\Sigma : \mathbf{Set}^{\mathbb{F}} \to \mathbf{Set}^{\mathbb{F}}$ (これは正確に書くと V を与える定数関数 $K_\mathrm{V}(K_\mathrm{V}(X) \triangleq \mathrm{V})$ 関手との和である関手 $K_\mathrm{V} + \Sigma$ のことだが，慣習にしたがって簡単に V+Σ と書く) に関する代数のことを意味する．より詳しくいえば，Σ 代数 (A, α) に $\mathbf{Set}^{\mathbb{F}}$ の射

$$\nu : \mathrm{V} \longrightarrow A$$

が付随したものが，V+Σ 代数 $(A, [\nu, \alpha])$ である．Σ 中の関数記号は α で解釈し，加えて V 中の変数を ν を解釈することを意図した構造である.

定義 6.2 A を $\mathrm{V}+\Sigma$ 代数とする．**項で生成された割り当て**(term-generated assignment)$\phi : Z \longrightarrow A$ とは $\mathbf{Set}^{\mathbb{F}}$ の射で以下のような合成で表せるもののことである:

$$Z \xrightarrow{\theta} T_\Sigma \mathrm{V} \xrightarrow{!_A} A$$

ここで θ はある付値，$!_A$ は始 $\mathrm{V}+\Sigma$ 代数 $T_\Sigma \mathrm{V}$ からの 唯一の準同型写像である．本論文を通して，$!_A$ と書いたとき，始代数からの唯一の $\mathrm{V}+\Sigma$ 準同型写像を表す.

これはより具体的は，メタ変数 z の解釈の方法を表している．つまり，始めに項で生成された割り当て θ で z にある項 t を割り当てて，次にこの項を V+Σ 代数 A で解釈するわけである．なぜこれが必要かというと，CRS の書換え関係は，項上に (メタ項上でないことに注意) 生成されるためである．つまり，CRS の書換え関係を解釈するのに，任意の割り当てを用いてはならなく，項で生成された割り当てのみを使わなければならないのである.

定義 6.3 **単調** V+Σ **代数**$(A, >_A)$ とは V+Σ 代数 $A = (A, [\nu, [F_A]_{F\in\Sigma}])$(ここで $\nu : \mathrm{V} \longrightarrow A$) でかつ，各 $n \in \mathbb{N}$ について $A(n)$ 上の推移的関係 $>_{A(n)}$ を持つものであり，また，各操作 F_A が単調なものであるものである．さらに，もしすべての $n \in \mathbb{N}$ について，$>_{A(n)}$ が整礎なとき，A を**整礎**(well-founded) と呼ぶ.

定義 6.4 $\mathcal{R}$ を CRS とする．以下が成り立つとき，単調 V+Σ 代数 $(A, >_A)$ は書換え規則 $Z \vdash \vec{n}.l \to \vec{n}.r$ を**満たす**という: すべての 項で生成された割り当て $\phi : Z \longrightarrow A$ について

$$\phi^*(n)(l) >_{A(n)} \phi^*(n)(r).$$

(V+$\Sigma, \mathcal{R}$) **代数** A とは，単調 V+Σ 代数で拡大閉包 $\mathcal{R}^\circ$ 中のすべての書換え規則を満たすものをいう.

$\mathbb{N}$ 添字付き推移関係 $\to^{+}_{\mathcal{R}(n)} \triangleq \{(s,t) \mid n \vdash s \Rightarrow^{+}_{\mathcal{R}} \circ t\}$ を定義する．ここで右辺の $(-)^{+}$ は 推移閉包を表している．

定理 6.5 CRS $\mathcal{R}$ について $(T_\Sigma \mathrm{V}, \to^{+}_{\mathcal{R}})$ は始 $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数，すなわち任意の $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数 A について，唯一の単調準同型写像 $T_\Sigma \mathrm{V} \longrightarrow A$ が存在する．

証明 $(A, >_A)$ を $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数とする．$T_\Sigma \mathrm{V}$ は始 $\mathrm{V}{+}\Sigma$ 代数であるから (定理 5.3)，$!_A : T_\Sigma \mathrm{V} \longrightarrow A$ は唯一の $\mathrm{V}{+}\Sigma$ 代数準同型写像である．よって残りは，$!_A$ が単調であることを示せばよい．これは $\Rightarrow_{\mathcal{R}}$ の導出木についての帰納法と $\Rightarrow^{+}$ の長さについての帰納法で示せばよい．ここで，$!_A \circ \theta$ が項で生成された割り当てであることと，すべての操作 F_A は A 上で単調であることを用いる． □

次は $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数が多ステップの書換え関係について健全かつ完全であることを表している．

系 6.6 $\mathcal{R}$ を CRS とすると次の二つは同値．

(i) $n \vdash s \to^{+}_{\mathcal{R}} t$ が成立する．

(ii) すべての $(\mathrm{V} + \Sigma, \mathcal{R})$ 代数 $(A, >_A)$ について $!_A(n)(s) >_{A(n)} !_A(n)(t)$.

証明 (i)⇒(ii): 定理 6.5 による．
(ii)⇒(i): $(A, >_A) = (T_\Sigma \mathrm{V}, \to^{+}_{\mathcal{R}})$ と取る． □

上の系を整礎な単調代数の場合に限ると，停止性を持つ CRS の完全な特徴付けを得る:

定理 6.7 CRS $\mathcal{R}$ が停止的である ⇔ 整礎な $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数が存在する．

証明 (⇐): A を整礎な $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数とする．$\mathcal{R}$ が停止性を持たない，すなわち無限の書換えの列 $n \vdash t_1 \to_{\mathcal{R}} t_2 \to_{\mathcal{R}} \cdots$. が存在すると仮定する．系 6.6 より，$!_A(n)(t_1) >_{A(n)} !_A(n)(t_2) >_{A(n)} \cdots$. を得る．これは $>_A$ が整礎であることと矛盾する．
(⇒): $\mathcal{R}$ が停止性を持つなら，推移的関係 $\to^{+}_{\mathcal{R}}$ が整礎であるから $(T_\Sigma \mathrm{V}, \to^{+}_{\mathcal{R}})$ が期待する整礎な $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数である． □

6.1 関数による解釈の不完全性

例として，メタ変数の集合 $Z = \{\mathrm{F}^1, \mathrm{X}^1\}$ と束縛シグネチャ$\Sigma = \{c : \langle 0 \rangle\}$ を仮定する．以下の一つの書換え規則のみからなる CRS $\mathcal{R}$ を考える:

$$c(\mathrm{F}[\mathrm{F}[\mathrm{X}[1]]]) \to \mathrm{F}[\mathrm{X}[1]].$$

実際にはメタ項を closed にするためにバインダ “1.” を両辺につける必要があるが，ここでは省略している．さて，$\mathcal{R}$ の停止性を示したい．直感的な議論では，この書換え規則を使った書換えは，必ず c の記号が減ることにより停止すると考えることができる．にもかかわらず，これまで知られている高階書換えを遺伝的な単調関数類 (hereditary monotone functionals) で解釈する方法 [22, 23] では，$\mathcal{R}$ **の停止性は証明できなかった**．これはこのモデルが高階書換えに対して不完全であったためである．これとは対照的に，我々は定理 6.7 を使って $\mathcal{R}$ の停止性を示すことができる．単調 $\mathrm{V}{+}\Sigma$ 代数 $(T_\Sigma \mathrm{V}, \succ_{T_\Sigma \mathrm{V}})$ を考える．ここで推移的関係 $\succ_{T_\Sigma \mathrm{V}}$ は以下で定義されるものである．

$$s \succ_{T_\Sigma \mathrm{V}(n)} t \quad \Leftrightarrow \quad \text{記号 } c \text{ の数は } s \text{ から } t \text{ で減っている}$$

ここで，$T_\Sigma \mathrm{V}(n)$ 中のすべての項は c と n 中の変数からのみできていることに注意する．ゆえにすべての $T_\Sigma \mathrm{V}$ への割り当ては $\mathrm{F} \mapsto c^k(1)$, $\mathrm{X} \mapsto c^m(1)$ (それぞれ k 回と m 回の c のネストを表す) の形になることがわかる．これは整礎な $(\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数 $(T_\Sigma \mathrm{V}, \succ_{T_\Sigma \mathrm{V}})$ を与え，よって定理 6.7 より $\mathcal{R}$ の停止性が結論できる．

7 メタ書換えの代数的意味論

本節では，CRS の標準的な定義からさらに進んで，新しい書換えの概念を導入する．これは，我々が**メタ書換え**と呼ぶ，**メタ項**の上での書換えである．これまでの CRS を扱う文献では，メタ書換えは形式的には定義されていなかったが，Oostrom はメタ項の上の書き換えの性質として，メタ CR とメタ SN を考察し，これらがそれぞれは，CRS の CR と SN からは導き出せないことを指摘した ([17] Sect. 3.4)．すなわちこれは，メタ項上の書換えが，CRS の通常の項上の書換えより真に広い範囲のものであることを表している．類似のものとして，このようなメタレベルの書換えは，佐藤らの体系 [25] においてより深く調べられている．

我々は特に**メタ停止性**，すなわち，メタ書換えの停止性を議論する．メタ停止性は停止性を導くという

利点も持つからである (上記のように逆は言えない). 本節では，メタ書換えの代数的意味論を与える．基本的には 6 節の意味論を繰り返すことになるが，Σ 代数の代りに Σ モノイドを用いる点が異なる．

始めにメタ書換えの厳密な定義を行う．Z をメタ変数の $\mathbb{N}$-添字付き集合とする．CRS $\mathcal{R}$ を任意の二つの書換え規則がそれぞれ互いに素なメタ変数を持つようなっており (もしそうでなければ，適切に書換え規則を名前換えする) さらにこれらすべてのメタ変数は Z から取られるとき，この CRS を $(\mathcal{R}, Z)$ と書くことにする．メタ書換え関係 $\rightsquigarrow_{\mathcal{R}}$ を次のように定義する．

$$\frac{\vec{n}.l \to \vec{n}.r \in \mathcal{R}}{n \vdash \theta^*(n)(l) \rightsquigarrow_{\mathcal{R}} \theta^*(n)(l)}$$

$$\frac{n{+}i \vdash s \rightsquigarrow_{\mathcal{R}} t}{n \vdash F(\ldots, n{+}\vec{i}.s, \ldots) \rightsquigarrow_{\mathcal{R}} F(\ldots, n{+}\vec{i}.t, \ldots)}$$

$$\frac{\mathrm{z} \in Z(l) \quad n \vdash s \rightsquigarrow_{\mathcal{R}} t}{n \vdash \mathrm{z}[\ldots, s, \ldots] \rightsquigarrow_{\mathcal{R}} \mathrm{z}[\ldots, t, \ldots]}$$

ここでθ は**メタ付値**$Z \longrightarrow M_\Sigma X$ (定義 5.7) である．$\rightsquigarrow_{\mathcal{R}}$ が整礎な関係のとき，$\mathcal{R}$ はメタ停止性を持つという．

定義 7.1 単調 Σ モノイド $(A, >_A)$ とは，Σ モノイド A で各 $n \in \mathbb{N}$ に対して $A(n)$ 上の推移的関係 $>_{A(n)}$ を持つち，かつ各操作 F_A が単調なものである．さらにもしすべての $n \in \mathbb{N}$ に対して $>_{A(n)}$ が整礎ならば，A を**整礎**という．

定義 7.2 (A, ν, β) を単調 Σ モノイド，$\phi : Z \longrightarrow A$ を割り当てとする．射 $\sigma : Z \bullet A \longrightarrow A$ を合成

$$Z \bullet A \xrightarrow{\phi \bullet \mathrm{id}_A} A \bullet A \xrightarrow{\beta} A$$

として定義する．割り当て ϕ は，σ が単調[2]のとき**許容的**(admissible) という．

ここで乗法 β は単調である必要はないことに注意する．実際，乗法が単調であるような Σ モノイドの例を見つけることは困難である．単位 $\nu : \mathrm{V} \longrightarrow A$ は V が推移的関係を持たないので自動的に単調である．

定義 7.3 $(\mathcal{R}, Z)$ を CRS とする．以下が成り立つとき単調 Σ モノイド $A = (A, >_A)$ は書換え規則 $Z' \vdash \vec{n}.l \to \vec{n}.r \in \mathcal{R}$ を**満たす**という: すべての割り当て $\phi : Z' \longrightarrow A$ に対して

$$\phi^*(n)(l) >_{A(n)} \phi^*(n)(r)$$

A を，許容的な付値 $Z \longrightarrow A$ が少なくとも一つある Σ モノイドとする．もし A がすべての拡大閉包 $\mathcal{R}^\circ$ 中のすべての書換え規則を満たすならば，これを $(\Sigma, \mathcal{R})$ **モノイド**と呼ぶ.

重要な $(\Sigma, \mathcal{R})$ モノイドの例として，$(M_\Sigma Z, \rightsquigarrow_{\mathcal{R}}^+)$ がある．

命題 7.4 $(M_\Sigma Z, \rightsquigarrow_{\mathcal{R}}^+)$ は任意の書換え規則 $Z \vdash \vec{n}.l \to \vec{n}.r \in \mathcal{R}$ を満たす.

証明 書換え関係 $\rightsquigarrow_{\mathcal{R}}$ の定義より任意の割り当て $\theta : Z \longrightarrow M_\Sigma Z$ に対して $\theta^*(n)(l) \rightsquigarrow_{\mathcal{R}} \theta^*(n)(r)$ である． □

許容的な割り当ての概念はメタ書換えの解釈のために重要な要素である．すべての割り当てを用いるのは，メタ書換えを解釈するには不適切である．なぜなら，ある場合には書換えの関係の解釈に重要な単調性が崩れるからである．これは例えば次の例を考えるとよく分かる．定数 $\Sigma = \{a, b, c\}$ とメタ変数 $Z = \{\mathrm{z}^1\}$ を仮定し，CRS $\mathcal{R} = \{a \to b\}$ を考える．このときメタ書換え $\mathrm{z}[a] \rightsquigarrow_{\mathcal{R}} \mathrm{z}[b]$ がある．この書換えステップを $(\Sigma, \mathcal{R})$ モノイド $(M_\Sigma Z, \rightsquigarrow_{\mathcal{R}}^+)$ で解釈してみる．割り当て $\phi : \mathrm{z} \longmapsto c$ を取ると，これは順序を保存しない:

$$\phi^*(1)(\mathrm{z}[a]) = c \not\rightsquigarrow_{\mathcal{R}} c = \phi^*(1)(\mathrm{z}[b]).$$

我々がメタ書換えの解釈のために必要なのは「単調」なメタ書換えの解釈である．したがって，この種の「非単調」な書換えステップの解釈を引き起こすような割り当てを禁止するということが,「許容的割り当て」の概念のアイデアである．

さて，ここで定理 6.5 の拡張として $(M_\Sigma Z, \rightsquigarrow_{\mathcal{R}}^+)$ がメタ書換えの始モデル (initial model) であることを主張する次の定理を示す．

[2] より詳しくは，$\sigma(n) : \coprod_{m \in \mathbb{N}} Z(m) \times A(n)^m / \sim \longrightarrow A(n)$ が単調，すなわち $\mathrm{z} \in Z(m)$ で，すべての $a_1, b_1 \in A(n), \ldots, a_m, b_m \in A(n)$ に対してある k については $a_k >_{A(n)} b_k$ で，他の $j \neq k$ については $a_j = b_j$ であるとき $\sigma(n)(\mathrm{z}; a_1, \ldots, a_m) >_{A(n)} \sigma(n)(\mathrm{z}; b_1, \ldots, b_m)$

定理 7.5 $(\mathcal{R}, Z)$ を CRS とする．Z から $(\Sigma, \mathcal{R})$ モノイド $(A, >_A)$ への任意の許容的な割り当て ϕ に対して，圏 $\mathbf{Set}^{\mathbb{F}}$ 中で以下の図式を可換にする単調な Σ モノイド射 ϕ^* が唯一に存在する．ここで $\eta_Z : \mathrm{z}^l \longmapsto \mathrm{z}[1, \ldots, l]$ である．

$$\begin{tikzcd} Z \arrow[r, "\eta_Z"] \arrow[dr, "\phi"'] & M_\Sigma Z \arrow[d, "\phi^*"] \\ & A \end{tikzcd}$$

証明 $\phi : Z \longrightarrow A$ を $(\Sigma, \mathcal{R})$ モノイド $(A, >_A)$ への許容的な割り当てとする．$M_\Sigma Z$ は自由 Σ モノイドであるから [10]，ϕ^* は上の図式を可換にする唯一の Σ モノイド射である．よって残りは ϕ^* が単調なことを示せばよい．これは $\Rightarrow_{\mathcal{R}}$ の導出木についての帰納法と $\Rightarrow^+$ の長さについての帰納法で示せばよい．ここで，書換え規則の具体化の場合は，$(\phi^* \circ \theta)^* = \phi^* \circ \theta^*$ を使いメタ項の構造についての帰納法で示せる．メタ適用の場合は，ϕ^* が $\mathrm{z}[\ldots, s, \ldots] \rightsquigarrow_{\mathcal{R}} \mathrm{z}[\ldots, t, \ldots]$ の関係を保存することを示す必要があるが，これは ϕ を許容的と仮定していたことを使う． □

定理 7.6 $\mathrm{CRS}(\mathcal{R}, Z)$ はメタ停止性を持つ $\Leftrightarrow$ 整礎な $(\Sigma, \mathcal{R})$ モノイドが存在する．

証明 $(\Leftarrow)$: A は整礎な $(\Sigma, \mathcal{R})$ モノイドであると仮定．$\mathcal{R}$ がメタ停止性を持たないと仮定，すなわち無限のメタ書換え列

$$Z|n \vdash t_1 \rightsquigarrow_{\mathcal{R}} t_2 \rightsquigarrow_{\mathcal{R}} \cdots.$$

が存在すると仮定すると定理 7.5 より，任意の許容的な割り当て $\phi : Z \longrightarrow A$ に対して

$$\phi^*(n)(t_1) >_{A(n)} \phi^*(n)(t_2) >_{A(n)} \cdots.$$

を得る．これは $>_A$ が整礎であることと矛盾する．
$(\Rightarrow)$: $\mathcal{R}$ が停止性を持つなら，推移的関係 $\rightsquigarrow^+_{\mathcal{R}}$ が整礎であるから $(M_\Sigma Z, \rightsquigarrow^+_{\mathcal{R}})$ が期待する整礎な $(\mathrm{V}{+}\Sigma, \mathcal{R})$ モノイドである． □

8 束縛 CRS の停止性

$(\mathcal{R}, X)$ を書換え規則中のメタ適用はすべて $\mathrm{z}^l[1, \ldots, l]$ の形になっている CRS であるとする．我々は，このような CRS を**束縛 CRS** と呼ぶことにする．なぜならこのような CRS は，本質的にはメタ適用がないものと同様であるからである (束縛 TRS の概念も参照のこと [9])．なぜなら，この形のメタ適用は z が代入によって具体化されたときに，適用としての代入は変数 $1, \ldots, l$ を $1, \ldots, l$ に置き換えるということになり，何もしないことになるためである．

これは，理論的にも次のように説明できる．束縛 CRS において特徴的なことは，書換え規則とメタ書換えを解釈するには，Σ モノイドの構造のうち，モノイド構造を使う必要がない点である．すなわち，乗法 β は解釈に使われることがない．なぜなら例えば，メタ変数 z^2 を仮定して書換え規則中にメタ項 $\mathrm{z}[1, 2]$ があったとする．これを Σ モノイド (A, ν, β) への割り当て $\phi : X \longrightarrow A$ で解釈すると

$$\begin{aligned} \phi^*(2)(\mathrm{z}[1, 2]) &= \beta(2)(\phi(2)(\mathrm{z}); \nu(2)(1), \nu(2)(2)) \\ &= \phi(2)(\mathrm{z}) \end{aligned}$$

となる．これは $A \bullet \mathrm{V} \cong A$ すなわち，V がモノイダル圏 $\mathbf{Set}^{\mathbb{F}}$ の単位 (unit) であることに起因する．よって $\mathrm{z}[1, 2]$ のようなメタ項を解釈するには，単に ϕ を用いればよいことになる．

ゆえに束縛 CRS の解釈のためには，任意の $X{+}\mathrm{V}{+}\Sigma$ 代数 A を用いる．解釈のやり方は，メタ書換えの解釈の議論を繰り返せばよい．A が束縛 CRS $\mathcal{R}$ の書換え規則 $\vec{n}.l \to \vec{n}.r \in \mathcal{R}$ を**満たす**とはすべての $X{+}\mathrm{V}{+}\Sigma$ 代数への割り当て $\phi : X \longrightarrow A$ に対して $\phi^*(n)(l) >_A \phi^*(n)(r)$ が成り立つことをいう．そして，始 $X{+}\mathrm{V}{+}\Sigma$ 代数を $B_\Sigma X$ 書くことにして，これの要素を**束縛メタ項**と呼ぶ．このとき束縛 CRS は，束縛メタ項からなる CRS といえることに注意する．束縛メタ項上のメタ書換えを $\to_{\mathcal{R}} \triangleq \rightsquigarrow_{\mathcal{R}} \cap \bigcup_{n \in \mathbb{N}} (B_\Sigma X \times B_\Sigma X)(n)$. と定義する．すると $(B_\Sigma X, \to^+_{\mathcal{R}})$ は始 $(X{+}\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数である．よって次を得る．

命題 8.1 束縛 $\mathrm{CRS}(\mathcal{R}, X)$ は束縛メタ項上に限ったメタ停止性を持つ $\Leftrightarrow$ 整礎 $(X{+}\mathrm{V}{+}\Sigma, \mathcal{R})$ 代数が存在する．

束縛 CRS $\mathcal{R}$ に対して，$\mathcal{R}$ の束縛メタ項上に限ったメタ停止性は，明らかに $\mathcal{R}$ の項の上の停止性を導く．なぜなら (メタ適用がないため) すべての項は束縛メタ項であるからである．ゆえに束縛 CRS の場合，これは解釈による興味深い停止性証明方法になる．

例 8.2 1 節の冠頭標準系への変換の CRS $\mathcal{R}$ の停止性を示してみる．形式的には，$\mathcal{R}$ は束縛シグネチャ

$\Sigma = \{\forall, \exists : \langle 1\rangle, \wedge, \vee : \langle 0,0\rangle, \neg : \langle 0\rangle\}$ とメタ変数の集合 $X = \{\mathrm{P}^0, \mathrm{Q}^1\}$ からつくられている．$\mathcal{R}$ を de Bruijn level の方法で書くには，単に変数を x を 1 と書けばよい．

$$\begin{array}{ll} \mathrm{P} \wedge \forall(1.\mathrm{Q}[1]) & \to \forall(1.\mathrm{P} \wedge \mathrm{Q}[1]) \\ \neg\forall(1.\mathrm{Q}[1]) & \to \exists(1.\neg(\mathrm{Q}[1])) \\ \forall(1.\mathrm{Q}[1]) \wedge \mathrm{P} & \to \forall(1.\mathrm{P} \wedge \mathrm{Q}[1]) \\ \neg\exists(1.\mathrm{Q}[1]) & \to \forall(1.\neg(\mathrm{Q}[1])). \end{array}$$

命題 8.1 を停止性を示すのに用いる．X+V+Σ 代数 K を各 $n \in \mathbb{N}$ に対して，$K(n) = \mathbb{N}$ で $>_{K(n)}$ を $\mathbb{N}$ 上の通常の順序 $>$ であるように取る．操作は

$$\wedge_{K(n)}(x,y) = \vee_{K(n)}(x,y) = 2x + 2y$$
$$\neg_{K(n)}(x) = 2x \qquad \forall_{K(n)}(x) = \exists_{K(n)}(x) = x + 1.$$

で与える．するとすべての操作は単調である．K は書換え規則を満たすことを示す．割り当て $\phi : X \longrightarrow K$ を $\mathrm{P} \mapsto x \in \mathbb{N}$ で $\mathrm{Q} \mapsto y \in \mathbb{N}$ とすると

$$\begin{aligned} \phi^*(0)(\mathrm{P} \wedge \forall(1.\mathrm{Q}[1])) &= 2x + 2(y+1) \\ &>_{K(0)} (2x+2y) + 1 \\ &= \phi^*(0)(\forall(1.\mathrm{P} \wedge \mathrm{Q}[1])) \\ \phi^*(0)(\neg\exists(1.\mathrm{Q}[1])) &= 2(y+1) \\ &>_{K(0)} 2y + 1 \\ &= \phi^*(0)(\forall(1.\neg(\mathrm{Q}[1]))) \end{aligned}$$

となる．他の書換え規則についても同様である．$>_{K(0)}$ は整礎であるから，これはϕ とともに K は整礎 $(X+\mathrm{V}+\Sigma, \mathcal{R})$ 代数であることを示している．よって，束縛 CRS $\mathcal{R}$ は命題 8.1 より束縛メタ項上に限ったメタ停止性を持つ．ゆえに $\mathcal{R}$ はすべての項の上で停止性を持つ．この解釈は，遺伝的な単調関数類モデル [23] と比べて (多項式しか必要としていないため) より単純化されているという利点がある．

例 8.3 例 4.3 の CPS 変換のための CRS $\mathcal{S}$ の停止性もまた次の多項式による解釈で証明できる．X+V+Σ 代数 K を $K(n) = \mathbb{N}$ で，単位が $\nu : \mathrm{V} \to K, i \mapsto 0$ であるように取る．

$$\mathsf{CPS}_{K(n)}(e) = 5e + 5 \quad (\![e]\!)_{K(n)} = 5e + 1$$
$$\overline{\lambda}_{K(n)}(e) = e \quad \lambda_{K(n)}(e) = e + 1$$
$$(e_0 \,\overline{\ }\, e_1)_{K(n)} = e_0 + e_1 \quad (e_0\ e_1)_{K(n)} = e_0 + e_1 + 1.$$

これが $\mathcal{S}$ を満たすことは書換え規則の解釈を計算するだけでわかる．ゆえに $\mathcal{S}$ は停止性を持つ．

つまり，もし CRS が束縛 CRS ならば，$\forall, \exists, \overline{\lambda}, \lambda$ のような抽象を引数に取る高階の関数記号を「関数」として解釈する必要がないわけである．この利点は，特に多項式を用いた証明法による束縛 CRS の停止性証明の自動化にも有用である．

謝辞

本研究を遂行するにあたって，契機となるアイデア，議論，助言と励ましをいただいた Gordon Plotkin, John Power, Neil Ghani, Aart Middeldorp 各氏に感謝致します．本研究は科学研究費補助金若手 (B)16700005 の援助を受けています．

参考文献

[1] Aczel, P.: A general Church-Rosser theorem, Technical report, University of Manchester, 1978.

[2] Baader, F. and Nipkow, T.: *Term Rewriting and All That*, Cambridge University Press, 1998.

[3] Blanqui, F., Jouannaud, J.-P., and Okada, M.: The Calculus of Algebraic Constructions, *Rewriting Techniques and Applications (RTA 1999)*, LNCS 1631, Springer, 1999, pp. 301–316.

[4] Blanqui, F., Jouannaud, J.-P., and Okada, M.: Inductive data type Systems, *Theoretical Computer Science*, Vol. 272(2002), pp. 41–68.

[5] Danvy, O. and Rose, K.: Higher-Order Rewriting and Partial Evaluation, *Rewriting Techniques and Applications, 9th International Conference, (RTA'98)*, LNCS 1379, 1998.

[6] de Bruijn, N.: Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem, *Indagationes Mathematicae*, Vol. 34(1972), pp. 381–391.

[7] Despeyroux, J., Felty, A., and Hirschowitz, A.: Higher-order abstract syntax in Coq, *Typed Lambda Calculi and Applications, LNCS 902*, 1995, pp. 124–138.

[8] Fiore, M., Plotkin, G., and Turi, D.: Abstract syntax and variable binding, *Proc. 14th Annual Symposium on Logic in Computer Science*, 1999, pp. 193–202.

[9] Hamana, M.: Term Rewriting with Variable Binding: An Initial Algebra Approach, *Fifth ACM-SIGPLAN International Conference on Principles and Practice of Declarative Programming (PPDP'03)*, 2003, pp. 148–159.

[10] Hamana, M.: Free Σ-monoids: A Higher-order Syntax with Metavariables, *Asian Symposium on Programming Languages and Systems (APLAS 2004)*, LNCS 3302, 2004, pp. 348–363.

[11] Jouannaud, J.-P. and Rubio, A.: Higher-Order Recursive Path Orderings à la carte, *International Workshop on Rewriting in Proof and Computation (RPC'01)*, 2001, pp. 161–175.

[12] Klop, J.: *Combinatory Reduction Systems*, PhD Thesis, CWI, Amsterdam, 1980. volume 127 of Mathematical Centre Tracts.

[13] Klop, J., Oostrom, V., and Raamsdonk, F.: Combinatory Reduction Systems: Introduction and Survey., *Theor. Comput. Sci.*, Vol. 121,No. 1&2(1993), pp. 279–308.

[14] Lescanne, P. and Rouyer-Degli, J.: Explicit Substitutions with de Bruijn's Levels, *Rewriting Techniques and Applications, 6th International Conference (RTA-95)*, LNCS 914, Springer, 1995, pp. 294–308.

[15] Mac Lane, S.: *Categories for the Working Mathematician*, Graduate Texts in Mathematics, Vol. 5, Springer-Verlag, New York, 1971.

[16] Nipkow, T.: Higher-Order Critical Pairs, *Proc. 6th IEEE Symp. Logic in Computer Science*, 1991, pp. 342–349.

[17] Oostrom, V.: *Confluence for Abstract and Higher-Order Rewriting*, PhD Thesis, Vrije Universiteit, Amsterdam, 1994.

[18] Oostrom, V. and Raamsdonk, F.: Comparing Combinatory Reduction Systems and Higher-order Rewrite Systems, *the First International Workshop on Higher-Order Algebra, Logic and Term Rewriting (HOA'93)*, LNCS 816, 1994.

[19] Pfenning, F. and Elliott, C.: Higher-order abstract syntax, *Proceedings of the ACM SIGPLAN '88 Symposium on Language Design and Implementation*, 1988, pp. 199–208.

[20] Pierce, B.: *Types and Programming Languages*, The MIT Press, 2002.

[21] Plotkin, G.: Binding Algebras: A Step between Universal Algebra and Type Theory (invited talk), *Rewriting Techniques and Applications, 9th International Conference, RTA'98, Tsukuba, Japan*, 1998.

[22] Pol, J.: Termination proofs for higher-order rewrite systems, *the First International Workshop on Higher-Order Algebra, Logic and Term Rewriting (HOA'93)*, LNCS 816, 1994, pp. 305–325.

[23] Pol, J.: *Termination of Higher-order Rewrite Systems*, PhD Thesis, Universiteit Utrecht, 1996.

[24] Raamsdonk, F.: Examples of higher-order rewriting systems. at `http://www.cs.vu.nl/~femke/ps/`.

[25] Sato, M., Sakurai, T., Kameyama, Y., and Igarashi, A.: Calculi of Meta-variables, *Computer Science Logic and 8th Kurt Gödel Colloquium (CSL'03 & KGC)*, LNCS 2803, 2003, pp. 484–497.

[26] Turi, D. and Plotkin, G.: Towards a Mathematical Operational Semantics, *Proceedings of the Twelfth Annual IEEE Symposium on Logic in Computer Science, LICS '97*, 1997, pp. 280–291.

[27] Zantema, H.: Termination of term rewriting: interpretation and type elimination, *Journal of Symbolic Computation*, Vol. 17(1994), pp. 23–50.

[28] 横山哲郎, 胡振江, 武市正人: 決定論的 2 階パターンとプログラム変換への応用, **コンピュータソフトウエア**, Vol. 21,No. 5(2004), pp. 71–76.

[29] 浜名誠: Σ モノイド—メタ変数と明示的環境を持つ高階構文, **コンピュータソフトウェア**, Vol. 22,No. 3(2005), pp. 201–207.