社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
SS2009-39 (2009-12)

高階書換え系における引数切り落とし法と実効規則

鈴木　翔††　草刈 圭一朗†　坂部 俊樹†　酒井 正彦†　西田 直樹†

†,†† 名古屋大学大学院情報科学研究科
〒 464-8603 名古屋市千種区不老町
E-mail: †suzuki@sakabe.i.is.nagoya-u.ac.jp, ††{kusakari,sakabe,sakai,nishida}@is.nagoya-u.ac.jp

あらまし　項書換え系（TRS）の拡張として高階関数を直接扱える単純型付き項書換え系（STRS）やさらにλ抽象も扱える高階書換え系（HRS）が提案されている．静的依存対法はこれら高階の系における非常に強力な停止性証明法である．静的依存対法で停止性を証明する際には引数切り落とし法や実効規則の概念が重要となる．これらはTRS上で提案されSTRS上に拡張されているが，HRS上では知られていない．本論文ではHRS上の引数切り落とし法と実効規則の概念を与える．さらに我々の成果は単なる拡張ではなく，引数切り落とし法で要求されていた適用条件を取り除き，実効規則の定義から高階変数を介した依存性を取り除くという改良も伴っている．

キーワード　高階書換え系，停止性，静的依存対法，引数切り落とし法，実効規則

Argument Filtering and Usable Rules in Higher-Order Rewrite Systems

SUZUKI Sho††, KUSAKARI Keiichirou †, SAKABE Toshiki †,
SAKAI Masahiko †, and NISHIDA Naoki †

†,†† Nagoya University, Graduate School of Information Science
Furo-cho, Chikusa-ku, Nagoya, 464–8601 Japan
E-mail: †suzuki@sakabe.i.is.nagoya-u.ac.jp, ††{kusakari,sakabe,sakai,nishida}@is.nagoya-u.ac.jp

Abstract　Simply-typed term rewriting systems (STRSs) and higher-order rewrite systems (HRSs) have been proposed to extend term rewriting systems (TRSs). These systems can handle higher-order functions directly, and moreover HRSs are allowed to have a λ-abstraction syntax. Static dependency pair methods are very powerful methods for proving termination of these higher-order systems. Notions of argument filtering and usable rules play important roles in proving termination by the static dependency pair method. These notions are proposed on TRSs, and extended to STRSs. However, we have not known these notions on HRSs. In this paper, we extend notions of argument filtering and usable rules onto HRSs. In addition, these extensions include some enhancements: removing an applicable condition demanded by existing argument fitering methods on STRSs, and removing a dependency through higher-order variables from the definition of usable rule.

Key words　higher-order rewrite system, termination, static dependency pair method, argument filtering, usable rule

1. はじめに

項書換え系（Term Rewriting System: TRS）は計算モデルのひとつである[13]．しかしTRSは関数型プログラミング言語でよく使われる高階関数を直接扱うことができない．そこでTRSの拡張として，高階関数を直接扱える単純型付き項書換え系（Simply-typed Term Rewriting System: STRS）[5]や高階書換え系（Higher-order Rewrite System: HRS）[10]が提案された．なおHRSではSTRSと違いλ抽象を取り扱えるため，匿名関数を利用することができる．例えば，高階関数のひとつである foldl は次のHRS $R_{\mathtt{foldl}}$ で表現することが出来る：

$$\begin{cases} \mathtt{foldl}(\lambda xy.F(x,y), X, \mathtt{nil}) \to X \\ \mathtt{foldl}(\lambda xy.F(x,y), X, \mathtt{cons}(Y,L)) \\ \qquad\qquad \to \mathtt{foldl}(\lambda xy.F(x,y), F(X,Y), L) \end{cases}$$

この foldl 関数を使ってリストの総和を求める sum 関数を表現するHRS $R_{\mathtt{sum}}$ は $R_{\mathtt{foldl}}$ と次の規則の和集合として与えることができる：

$$\begin{cases} \mathtt{add}(0,Y) \to Y \\ \mathtt{add}(\mathtt{s}(X),Y) \to \mathtt{s}(\mathtt{add}(X,Y)) \\ \mathtt{sum}(L) \to \mathtt{foldl}(\lambda xy.\mathtt{add}(x,y),0,L) \end{cases}$$

TRS の停止性証明法として再帰構造解析に基づく依存対法という手法が提案され [1]，その後 STRS 上 [5] と HRS 上 [11] に拡張された．依存対法とは再帰成分と呼ばれる再帰構造の解析結果の非循環性を示すことにより停止性を示す手法である．高階の系では再帰構造の解析に関して，関数呼び出しの依存関係に基づく動的な解析法と関数定義の依存関係に基づく静的な解析法がある．文献 [5], [11] はいずれも動的な再帰構造解析に基づいている．これに対し，静的依存対法という静的な再帰構造解析に基づく手法が STRS 上で提案された [6]．静的依存対法は非常に強力な停止性証明法であるが，一般には適用できない．そのため文献 [6] では直接関数渡しというクラスを定式化しこのクラスに属す STRS に対しては静的依存対法が適用可能であることを示した．この成果は HRS 上にも拡張されている [8]．さらに適用範囲を拡張するための関数渡しの安全性の定式化が STRS 上で行われている [7].

再帰成分の非循環性を示すためには簡約化対と呼ばれる概念が重要となるが，HRS 上での簡約化対の設計法は知られていない．一方 TRS 上では簡約化対の設計法として引数切り落とし法が提案され [1]，STRS 上にも拡張されている [5]．さらに文献 [5] で発生していた型が壊れてしまうという問題を回避した STRS 上の引数切り落とし法が提案されている [7]．本論文ではこの結果を HRS 上に拡張し，HRS 上での簡約化対の設計法を与える．さらに本論文で提案する引数切り落とし法は，STRS 上で必要だった左堅固と呼ばれる条件 [7] を要求しないため，より汎用性の高い手法となっている．

与えられた再帰成分に対して書換え規則が実効的かどうかを判定し，非循環性を示す際に考慮すべき規則を実効的な規則のみとすることで効率を上げる手法が知られている．このとき実効的と判定された規則を実効規則と呼ぶ．この概念は TRS 上で提案され [3], [4]，その後 STRS 上に拡張された [12]．本論文ではこの結果を HRS 上に拡張する．この拡張の際に高階変数を介した規則の実効性を考慮せずにすむ改良をすることで実効規則を削減し証明力をさらに高める．そのため証明は非常に困難になるが，補題の改良と巧妙な帰納法の導入によりこれを示す.

2. 準　　備

HRS は文献 [10] で提案された．本節では論文中で必要となる HRS の諸概念を文献 [8] に基づき紹介する.

単純型の集合 $\mathcal{S}$ は基本型の集合 $\mathcal{B}$ から型構成子 $\to$ で生成される．$\alpha \to \beta$ の形の単純型のことを関数型，もしくは高階型と呼ぶ．$\rhd_s$ を $\alpha_1 \to \alpha_2 \rhd_s \alpha_i$ $(i=1,2)$ により生成される型上の狭義の半順序で定義する.

準項は変数集合 $\mathcal{V}$ と関数記号の集合 Σ から λ 抽象と λ 適用により生成される．ただし，$\mathcal{V} \cap \Sigma = \emptyset$ であると仮定する．特に型付きの準項全体を $\mathcal{T}^{pre}$ で記す．単純型付き準項 t の η 拡張と β 簡約に関する正規形を $t{\downarrow}$ で記す．単純型付き項の集合 $\mathcal{T}$ を $\{ t{\downarrow} \mid t \in \mathcal{T}^{pre} \}$ で定義する．以降では単純型付き項を単に項と呼ぶ．項 t の型を $type(t)$ で記す．型 α の項全体を $\mathcal{T}_\alpha$ で記す．α 合同を $\equiv$ で，項 t 中の自由変数の集合を $FV(t)$ で表す．簡便のため，項に現れる束縛変数はすべて異なる名前を持つとし，自由変数とは重複しないとする．項 t は一般に $\lambda x_1 \cdots x_m.a\ t_1 \cdots t_n$ $(a \in \Sigma \cup \mathcal{V})$ と書ける．これを $\lambda x_1 \cdots x_m.a(t_1,\ldots,t_n)$ もしくは $\lambda\overline{x_m}.a(\overline{t_n})$ と略記することがある．項 $t \equiv \lambda\overline{x_m}.a(\overline{t_n})$ に対し，記号 a を t の先頭記号と呼び $top(t)$ で記し，$\{t_1,\ldots,t_n\}$ を t の引数と呼び $args(t)$ で記す．t の部分項の集合 $Sub(t)$ を $t \equiv \lambda x.s$ のときは $\{t\} \cup Sub(s)$，$t \equiv a(t_1,\ldots,t_n)$ のときは $\{t\} \cup \bigcup_{i=1}^{n} Sub(t_i)$ で定義する．$t \trianglerighteq_{sub} s$ で $s \in Sub(t)$ を表し，$t \rhd_{sub} s$ を $t \trianglerighteq_{sub} s$ かつ $t \neq s$ で定義する．項 t の位置の集合 $Pos(t)$ を $Pos(\lambda x.t) = \{\varepsilon\} \cup \{\, 1p \mid p \in Pos(t) \,\}$ と $Pos(a(t_1,\ldots,t_n)) = \{\varepsilon\} \cup \bigcup_{i=1}^{n} \{\, ip \mid p \in Pos(t_i) \,\}$ で帰納的に定義する．位置上の順序 $\prec$ を $p \prec q \overset{\text{def}}{\iff} \exists w(\neq \varepsilon).pw = q$ で定義する．t の位置 p における部分項を $t|_p$ と書く.

型が α である特別な定数記号 $\Box_\alpha$ を一つだけ含む項を文脈と呼び，$C[\,]$ で記す．$C[\,]$ に含まれる $\Box_\alpha$ を $t \in \mathcal{T}_\alpha$ で置き換えたものを $C[t]$ で表す．代入 θ を変数から項への写像として定義し，$\theta(X)$ は任意の変数 X に対し X と同じ型を持つとする．また $dom(\theta) = \{\, X \mid X{\downarrow} \neq \theta(X) \,\}$ と定義し，これはいつでも有限であると仮定する．代入 θ は項から項への写像に自然に拡張される．以降では $\theta(t)$ を $t\theta$ と記す.

高階書換え規則とは次の条件を満たす項の対 (l,r) であり，$l \to r$ で記す：$top(l) \in \Sigma$ かつ $type(l) = type(r) \in \mathcal{B}$ かつ $FV(l) \supseteq FV(r)$ (注1)．ある文脈 $C[\,]$ と代入 θ と書換え規則 $l \to r \in R$ が存在し，$s \equiv C[l\theta{\downarrow}]$ かつ $t \equiv C[r\theta{\downarrow}]$ のとき項 s から項 t に書換え可能といい，$s \underset{R}{\to} t$ と記す．$\underset{R}{\to}$ を書換え関係と呼び，その推移閉包と反射推移閉包を $\underset{R}{\overset{+}{\to}}$ と $\underset{R}{\overset{*}{\to}}$ で記す．高階書換え規則の集合を高階書換え系（HRS）と呼ぶ．$\{\, t' \mid t \underset{R}{\to} t' \,\}$ が無限集合となる t が存在しないとき HRS R は有限分岐であるという.

項 t から始まる HRS R の無限書換え系列が存在しなければ，t は停止性，あるいは強正規性を持つといい $SN(R,t)$ と記す．すべての項 t が $SN(R,t)$ であるとき R は停止性を持つといい，$SN(R)$ で記す．整礎で文脈と代入に閉じた関係を簡約化順序であるという．HRS R が停止性を持つことと $R \subseteqq >$ となる簡約化順序 $>$ が存在することは必要十分である.

最後に証明で必要となる命題を紹介しておく.

命題 2.1 ([9]) $t \underset{R}{\overset{*}{\to}} s \wedge \theta \underset{R}{\overset{*}{\to}} \theta' \Rightarrow t\theta{\downarrow} \underset{R}{\overset{*}{\to}} s\theta'{\downarrow}$.

3. 静的依存対法

静的依存対法は草刈・酒井により提案された静的再帰構造解析に基づく非常に強力な STRS の停止性証明法である [6]．し

(注1)：高階照合問題の決定可能性を保障するために Nipkow はパターンと呼ばれる概念を用いて規則を制限した [10]．だがそのような制限は本研究では必要としない.

かし静的依存対法は一般には適用できない．そのため文献[6]では直接関数渡しと呼ばれるクラスを定式化しこのクラスで健全であることを示した．また文献[7]では静的依存対法が健全であるための関数渡しの安全性を定式化し，適用クラスを拡張した．本節では文献[6]の成果をHRS上に拡張した静的依存対法[8]を紹介する．

定義 3.1 RをHRSとし$l \to r \in R$とする．$safe(l)$を以下で定義する：

$$args(l) \cup \bigcup_{l' \in args(l)} \{\, u \in safe_{\mathcal{B}}(l', FV(l)) \mid FV(l) \supseteq FV(u) \,\},$$

ただし$safe_{\mathcal{B}}(\lambda\overline{x_m}.a(\overline{t_n}), X)$を，$a \in X$のときは$\{a(\overline{t_n})\}$，そうでないときは$\{a(\overline{t_n})\} \cup \bigcup_{i=1}^{n} safe_{\mathcal{B}}(t_i, X)$で定義する．

全ての$l \to r \in R$と$Z(r_1, \ldots, r_n) \in Sub(r)$に対し，$Z \in FV(r)$ならば$Z(r_1, \ldots, r_k)\downarrow \in safe(l)$である$k$ $(\leqq n)$が存在するとき，Rは直接関数渡し（Plain Function-Passing: PFP）であるという．以降では直接関数渡しである高階書換え系をPFP-HRSと書くことがある．

1節で紹介したHRS $R_{\mathtt{foldl}}$に対し，$l \equiv \mathtt{foldl}(\lambda xy.F(x,y), X, \mathtt{cons}(Y, L))$とすると，$safe(l) = \{\lambda xy.F(x,y), X, \mathtt{cons}(Y,L), Y, L\}$となり，$F\downarrow \equiv \lambda xy.F(x,y) \in safe(l)$となる．よって明らかに$R_{\mathtt{foldl}}$はPFPである．

以降，本論文の例題で扱うHRSは全てPFP-HRSである．

定義 3.2 HRS Rの書換え規則の左辺の先頭記号を被定義記号と呼び，それらの集合を$\mathcal{D}_R$と書く．また，Rの被定義記号でない関数記号をRの構成子記号と呼び，それらの集合を$\mathcal{C}_R$と書く．項tの印付き項$t^\sharp$を$t \equiv f(t_1, \ldots, t_n)$かつ$f \in \mathcal{D}_R$に対し$f^\sharp(t_1, \ldots, t_n)$で，それ以外の場合は$t^\sharp \equiv t$で定義する．また$Cand(\lambda\overline{x_m}.a(\overline{t_n})) = \{\lambda\overline{x_m}.a(\overline{t_n})\} \cup \bigcup_{i=1}^{n} Cand(\lambda\overline{x_m}.t_i)$とする．

組$\langle l^\sharp, a^\sharp(r_1, \ldots, r_n)\rangle$が$R$の静的依存対であるとは以下の条件を全て満たす$l \to r \in R$が存在することである：

- $\lambda x_1 \cdots x_m.a(r_1, \ldots, r_n) \in Cand(r)$
- $a \in \mathcal{D}_R$
- 全てのk $(\leqq n)$で$a(r_1, \ldots, r_k)\downarrow \notin safe(l)$

以降では$\langle u^\sharp, v^\sharp\rangle$を$u^\sharp \to v^\sharp$と記す．また$R$の静的依存対全体を$SDP(R)$で表す．

例 3.3 1節で紹介したHRS $R_{\mathtt{sum}}$に対して，集合$SDP(R_{\mathtt{sum}})$は以下のようになる：

$$\left\{\begin{array}{l} \mathtt{foldl}^\sharp(\lambda xy.F(x,y), X, \mathtt{cons}(Y,L)) \\ \qquad\qquad \to \mathtt{foldl}^\sharp(\lambda xy.F(x,y), F(X,Y), L) \\ \mathtt{add}^\sharp(\mathtt{s}(X), Y) \to \mathtt{add}^\sharp(X, Y) \\ \mathtt{sum}^\sharp(L) \to \mathtt{foldl}^\sharp(\lambda xy.\mathtt{add}(x,y), 0, L) \\ \mathtt{sum}^\sharp(L) \to \mathtt{add}^\sharp(x, y) \end{array}\right.$$

HRS Rにおいて項tが強計算性を持つ（$SC(R,t)$で記す）ことを，$type(t) \in \mathcal{B}$のときは$SN(R,t)$で，$type(t) = \alpha \to \beta$のときは$\forall u \in \mathcal{T}_\alpha.(SC(R,u) \Rightarrow SC(R,(tu)\downarrow))$で定義する．また$\mathcal{T}_{SC}^{args}(R) = \{\, t \mid \forall u \in args(t).SC(R,u) \,\}$とする．

定義 3.4 HRS Rの静的依存対の列$u_0^\sharp \to v_0^\sharp, u_1^\sharp \to v_1^\sharp, \cdots$が静的依存鎖であるとは，各$i$で$v_i^\sharp\theta_i\downarrow \xrightarrow[R]{*} u_{i+1}^\sharp\theta_{i+1}\downarrow$かつ$u_i\theta_i\downarrow, v_i\theta_i\downarrow \in \mathcal{T}_{SC}^{args}(R)$となる$\theta_0, \theta_1, \ldots$が存在することである．

静的依存対法の基本定理は次のようになる．

命題 3.5 PFP-HRS Rに対し，無限の静的依存鎖が存在しないときRは停止性を持つ．

静的依存対法は静的な再帰構造を解析することで停止性を証明する手法だと述べたが，ここでいう解析結果とは次で与える静的再帰成分のことである．

定義 3.6 全ての頂点が$SDP(R)$の元であり，かつ$u^\sharp \to v^\sharp$, $u'^\sharp \to v'^\sharp$が静的依存鎖ならば必ず$u^\sharp \to v^\sharp$から$u'^\sharp \to v'^\sharp$への辺があるような有向グラフをRの静的依存グラフと呼ぶ．

静的依存グラフの強連結部分グラフの頂点集合を静的再帰成分と呼び，HRS Rの静的再帰成分全体を$SRC(R)$で記す．

HRS Rとその静的再帰成分Cに対し，C中の静的依存対のみから構成され，全ての$u^\sharp \to v^\sharp \in C$が無限回出現するような無限静的依存鎖が存在しないとき，Cは非循環的であるという．

例 3.7 1節で紹介したHRS $R_{\mathtt{sum}}$に対し$SRC(R_{\mathtt{sum}})$は以下の二つの静的再帰成分から構成される：

$$\left\{\begin{array}{l} \mathtt{foldl}^\sharp(\lambda xy.F(x,y), X, \mathtt{cons}(Y,L)) \\ \qquad\qquad \to \mathtt{foldl}^\sharp(\lambda xy.F(x,y), F(X,Y), L) \end{array}\right\}$$

$$\left\{\ \mathtt{add}^\sharp(\mathtt{s}(X), Y) \to \mathtt{add}^\sharp(X, Y)\ \right\}$$

命題 3.8 静的依存グラフ中に無限の経路が存在しないPFP-HRS Rに対し，全ての静的再帰成分が非循環的であればRは停止性を持つ．

以上でPFP-HRSの停止性を示すためには静的再帰成分の非循環性を示せばよいことが分かった．次に非循環性を示す際に用いられる部分項基準と簡約化対を紹介する．

定義 3.9 以下の条件を満たす関係$\gtrsim$と$>$の対$(\gtrsim, >)$を簡約化対と呼ぶ：

- $>$は整礎で代入に閉じる
- $\gtrsim$は代入と文脈に閉じる
- $\gtrsim \cdot > \subseteqq >$または$> \cdot \gtrsim \subseteqq >$が成立

特に，$(\gtrsim, \gtrsim \setminus \lesssim)$が簡約化対のとき$\gtrsim$を弱簡約化順序と呼ぶ．

定義 3.10 HRS Rと$C \in SRC(R)$に対し，$\mathcal{D}_R$から以下に示す空でない正整数の列への関数πが存在するときCは部分項基準を満たすという：

- ある$u^\sharp \to v^\sharp \in C$で$u|_{\pi(top(u))} \rhd_{sub} v|_{\pi(top(v))}$
- 全ての$u^\sharp \to v^\sharp \in C$が以下の条件を全て満たす:

- $u|_{\pi(top(u))} \trianglerighteq_{sub} v|_{\pi(top(v))}$
- $\forall p \prec \pi(top(u)).top(u|_p) \notin FV(u)$
- $\forall q \prec \pi(top(v)).q = \varepsilon \vee top(v|_q) \notin FV(v) \cup \mathcal{D}_R$

命題 3.11 静的依存グラフ中に無限の経路がない PFP-HRS R に対し，各静的再帰成分 $C \in SRC(R)$ が以下のいずれかを満たすとき C は非循環性を持つ：

- C は部分項基準を満たす.
- ある簡約化対 $(\gtrsim, >)$ が存在して $R \subseteqq \gtrsim$，$C \subseteqq \gtrsim \cup >$，かつ $C \cap > \neq \emptyset$.

例 3.12 例 3.7 で示した全ての静的再帰成分は $\pi(\mathtt{foldl}) = 3$, $\pi(\mathtt{add}) = 1$ とすると，以下の下線部の部分のみが切り出され，部分項基準を満たす.

$$\left\{ \begin{array}{l} \mathtt{foldl}^\sharp(\lambda xy.F(x,y), X, \underline{\mathtt{cons}(Y,L)}) \\ \qquad \to \mathtt{foldl}^\sharp(\lambda xy.F(x,y), F(X,Y), \underline{L}) \end{array} \right\}$$
$$\left\{ \mathtt{add}^\sharp(\underline{\mathtt{s}(X)}, Y) \to \mathtt{add}^\sharp(\underline{X}, Y) \right\}$$

よって命題 3.11 から HRS $R_{\mathtt{sum}}$ は停止性を持つ.

4. 引数切り落とし法

静的依存対法を利用する場合に重要となる簡約化対を設計する手法として，引数切り落とし法がある．引数切り落とし法は Arts と Giesl により TRS 上で提案された [1]．草刈はこれを STRS 上に拡張したが，型を壊してしまうという問題が発生していた [5]．後にこの問題を解決した改良版が STRS 上で提案された [7]．本節ではこの成果を HRS 上へ拡張する．さらに本論文で提案する引数切り落とし法は，文献 [7] で必要だった左堅固と呼ばれる条件（書換え規則の左辺では変数は葉の位置にしか現れてはいけない）が必要なくなるような改良も施す.

定義 4.1 任意の型 $\alpha \in \mathcal{S}$ に対し $type(\bot_\alpha) = \alpha$ となる新しい関数記号 $\bot_\alpha$ を用意する．引数切り落とし関数 π は各関数記号 $f \in \Sigma$ に対し $\pi(f) = [i_1, \ldots, i_k]$ となる関数である．ここで $type(f) = \alpha_1 \to \cdots \to \alpha_n \to \beta$ かつ $\beta \in \mathcal{B}$ とすると，$i_1 < \cdots < i_k \leqq n$ となっている．引数切り落とし関数は $\pi(\lambda\overline{x_m}.a(\overline{t_n})) = \lambda\overline{x_m}.a(\overline{t'_n})$ で項上へ拡張される．ここで，t'_i は $a \in \Sigma$ かつ $i \notin \pi(a)$ のときは $\bot_{\alpha_i}\downarrow$ で，そうでない場合は $\pi(t_i)$ で定める．また代入 θ_π を $\theta_\pi(x) = \pi(\theta(x))$ と定義する.

引数切り落とし関数 π と二項関係 $>$ に対し，$s \gtrsim_\pi t$ を $\pi(s) \geqq \pi(t)$ で，$s >_\pi t$ を $\pi(s) > \pi(t)$ で定義する.

以下では特に明示しない場合は $type(f) = \alpha_1 \to \cdots \to \alpha_n \to \beta$ かつ $\beta \in \mathcal{B}$ とすると，$\pi(f) = [1, \ldots, n]$ であるとする.

例 4.2 HRS $R_{\mathtt{div}}$ を次のように与える：

$$\left\{ \begin{array}{l} \mathtt{sub}(X, 0) \to X \\ \mathtt{sub}(0, Y) \to 0 \\ \mathtt{sub}(\mathtt{s}(X), \mathtt{s}(Y)) \to \mathtt{sub}(X, Y) \\ \mathtt{div}(0, \mathtt{s}(Y)) \to 0 \\ \mathtt{div}(\mathtt{s}(X), \mathtt{s}(Y)) \to \mathtt{s}(\mathtt{div}(\mathtt{sub}(X, Y), \mathtt{s}(Y))) \end{array} \right.$$

このとき $\pi(\mathtt{sub}) = [1]$ とすると $\pi(\mathtt{sub}(X, Y)) = \mathtt{sub}(X, \bot_N\downarrow)$ となる.

以下では引数切り落とし法が弱簡約化順序設計法として健全であることを示す.

まず $>_\pi$ と $\gtrsim_\pi$ が代入に閉じることを示すために必要となる補題を用意する．この補題は高階の系で引数切り落とし法の健全性を示す際に重要な役割を担う.

補題 4.3 $\pi(t\theta\downarrow) = \pi(t)\theta_\pi\downarrow$.

証明 $\xrightarrow[\beta]{} \cup \rhd_{sub}$ による $t\theta$ の構造に関する帰納法で示す．$t \equiv X(\overline{t_n})$ かつ $X \in \mathcal{V}$ かつ $n > 0$ の場合のみ示す．$X\theta = \lambda\overline{y_n}.a(\overline{u_k})$ とおく．$t\theta \equiv (\lambda\overline{y_n}.a(\overline{u_k}))(\overline{t_n\theta}) \xrightarrow[\beta]{+} a(\overline{u_k})\{y_i := t_i\theta\downarrow \mid i = 1, \cdots, n\}$ かつ $t\theta \rhd_{sub} t_i\theta$ なので帰納法の仮定から $\pi(a(\overline{u_k})\{y_i := t_i\theta\downarrow\}\downarrow) = \pi(a(\overline{u_k}))\{y_i := \pi(t_i\theta\downarrow)\}\downarrow$ かつ $\pi(t_i\theta\downarrow) = \pi(t_i)\theta_\pi\downarrow$. よって $\pi(t\theta\downarrow) = \pi(X(\overline{t_n})\theta\downarrow) = \pi((\lambda\overline{y_n}.a(\overline{u_k}))(\overline{t_n\theta\downarrow})\downarrow) = \pi(a(\overline{u_k})\{y_i := t_i\theta\downarrow\}\downarrow) = \pi(a(\overline{u_k}))\{y_i := \pi(t_i\theta\downarrow)\}\downarrow = \pi(a(\overline{u_k}))\{y_i := \pi(t_i)\theta_\pi\downarrow\}\downarrow = \pi(\lambda\overline{y_n}.a(\overline{u_k}))(\overline{\pi(t_n)\theta_\pi\downarrow})\downarrow = X(\overline{\pi(t_n)})\theta_\pi\downarrow = \pi(X(\overline{t_n}))\theta_\pi\downarrow = \pi(t)\theta_\pi\downarrow$. □

実はこの補題は STRS 上では $\pi(t\theta) \geqq \pi(t)\theta_\pi$ の形でしか成立しない．そのため左堅固と呼ばれる性質を持つ STRS にしか適用できない [7].

定理 4.4 任意の簡約化順序 $>$ と引数切り落とし関数 π に対し，$\gtrsim_\pi$ は弱簡約化順序になる.

証明 $\gtrsim_\pi$ が整礎であることは明らかであり，$\gtrsim_\pi$ が文脈に閉じることは $s \gtrsim_\pi t \Rightarrow C[s] \gtrsim_\pi C[t]$ を $C[]$ の構造に関する帰納法で示すことで証明できる．また $\gtrsim_\pi$ が代入に閉じることは補題 4.3 を用いて $s \gtrsim_\pi t \Rightarrow \pi(s) \geqq \pi(t) \Rightarrow \pi(s)\theta_\pi\downarrow \geqq \pi(t)\theta_\pi\downarrow \Rightarrow \pi(s\theta\downarrow) \geqq \pi(t\theta\downarrow) \Rightarrow s\theta\downarrow \gtrsim_\pi t\theta\downarrow$ と示すことができ，$>_\pi$ が代入に閉じることも同様に示せる．よって $\gtrsim_\pi$ は弱簡約化順序. □

例 4.5 $SRC(R_{\mathtt{div}})$ は次の静的再帰成分から構成される：

$$\left\{ \mathtt{sub}^\sharp(\mathtt{s}(X), \mathtt{s}(Y)) \to \mathtt{sub}^\sharp(X, Y) \right\}$$
$$\left\{ \mathtt{div}^\sharp(\mathtt{s}(X), \mathtt{s}(Y)) \to \mathtt{div}^\sharp(\mathtt{sub}(X, Y), \mathtt{s}(Y)) \right\}$$

一つめの静的再帰成分は部分項基準を満たしている．一方二つめは満たしていないので，引数切り落とし法を用いて非循環性を示す．簡約化順序としては文献 [2] で提案されている再帰経路順序 $>^{rpo}$ を $\mathtt{div} \rhd \mathtt{s} \rhd \mathtt{sum}$ に基づき利用し，引数切り落とし関数を $\pi(\mathtt{sub}) = [1]$ とする．このとき，

$$\begin{array}{rcl} \pi(\mathtt{div}^\sharp(\mathtt{s}(X), \mathtt{s}(Y))) & = & \mathtt{div}^\sharp(\mathtt{s}(X), \mathtt{s}(Y)) \\ & >^{rpo} & \mathtt{div}^\sharp(\mathtt{sub}(X, \bot_N\downarrow), \mathtt{s}(Y))) \\ & = & \pi(\mathtt{div}^\sharp(\mathtt{sub}(X, Y), \mathtt{s}(Y))) \end{array}$$

となる．さらに $R_{\mathtt{div}} \subseteqq \gtrsim_\pi$ がいえるので，命題 3.11 より div に関する静的再帰成分は非循環性を持つ．以上から命題 3.8 より $R_{\mathtt{div}}$ の停止性を示すことが出来る.

例 4.6 HRS $R_{\mathtt{ave}}$ を $R_{\mathtt{sum}}$ と $R_{\mathtt{div}}$ と次の規則の和集合として与える：

$$\left\{\begin{array}{l}\mathtt{s}'(X,Y)\to\mathtt{s}(X)\\ \mathtt{len}\to\mathtt{foldl}(\mathtt{s}',0)\\ \mathtt{ave}(L)\to\mathtt{div}(\mathtt{sum}(L),\mathtt{len}(L))\end{array}\right.$$

このとき $SRC(R_{\mathtt{ave}})$ は次の静的再帰成分で構成される：

$$\left\{\begin{array}{l}\mathtt{foldl}^\sharp(\lambda xy.F(x,y),X,\mathtt{cons}(Y,L))\\ \qquad\to\mathtt{foldl}^\sharp(\lambda xy.F(x,y),F(X,Y),L)\end{array}\right\}$$
$$\left\{\mathtt{add}^\sharp(\mathtt{s}(X),Y)\to\mathtt{add}^\sharp(X,Y)\right\}$$
$$\left\{\mathtt{sub}^\sharp(\mathtt{s}(X),\mathtt{s}(Y))\to\mathtt{sub}^\sharp(X,Y)\right\}$$
$$\left\{\mathtt{div}^\sharp(\mathtt{s}(X),\mathtt{s}(Y))\to\mathtt{div}^\sharp(\mathtt{sub}(X,Y),\mathtt{s}(Y))\right\}$$

このとき，上から三つめまでの静的再帰成分は部分項基準を満たすため非循環性を持つ（例 3.12,4.5 参照）．しかし div に関する静的再帰成分の非循環性は，例 4.5 の場合と異なり $R_{\mathtt{foldl}}\subseteqq\gtrsim_\pi^{rpo}$ がいえないために示すことが出来ない．

$R_{\mathtt{ave}}$ の停止性を示すためには次節で与える実効規則の概念を用いる必要がある．

5. 実効規則

前節の例 4.5 で div に関する静的再帰成分

$$\left\{\mathtt{div}^\sharp(\mathtt{s}(X),\mathtt{s}(Y))\to\mathtt{div}^\sharp(\mathtt{sub}(X,Y),\mathtt{s}(Y))\right\}$$

の非循環性を示すことが出来なかった理由は $R_{\mathtt{foldl}}\subseteqq\gtrsim_\pi^{rpo}$ が成立しないことだった．実効規則の概念を用いるとこの問題は解決できる．この例では再帰成分中で用いられている sub の規則のみを実効的な規則（実効規則）と考え，他の規則を実効的でないとして証明から省くことができる．

実効規則は最初に TRS 上で提案された [3], [4]．その後 STRS 上に拡張された [12]．この STRS 上の実効規則は高階変数を介した規則の実効性も考慮していた．例えば静的再帰成分

$$\left\{\begin{array}{l}\mathtt{foldl}^\sharp(\lambda xy.F(x,y),X,\mathtt{cons}(Y,L))\\ \qquad\to\mathtt{foldl}^\sharp(\lambda xy.F(x,y),F(X,Y),L)\end{array}\right\}$$

の実効規則として，高階変数 F に代入されうる add の規則も実効規則と考える必要があった．本節ではこの文献 [12] の結果を HRS 上に拡張する．この際に高階変数を介した規則の実効性を考慮せずにすむように改良を施す．したがって add の規則は本節の実効規則からは取り除くことができる．

定義 5.1 ある $l\to r\in R$ が存在して $top(l)=f$ かつ g が r 中に出現するとき $f>_{\mathtt{def}}g$ と記す．項 t に対し，$\mathcal{U}(t)=\{l\to r\in R\mid$ ある t 中に出現する $f\in\mathcal{D}_R$ に対し $f>_{\mathtt{def}}^* top(l)\}$ とする．再帰成分 C の実効規則の集合 $\mathcal{U}(C)$ を $\mathcal{U}(C)=\bigcup_{u^\sharp\to v^\sharp\in C}\mathcal{U}(v^\sharp)$ で定義する．

定義 5.2 任意の $\alpha\in\mathcal{B}$ に対して $\bot_\alpha$，$\mathtt{c}_\alpha$ を $type(\bot_\alpha)=\alpha$，$type(\mathtt{c}_\alpha)=\alpha\to\alpha\to\alpha$ である新しい関数記号とする．項書換え系 C_e を $C_e=\{\mathtt{c}_\alpha(x_1{\downarrow},x_2{\downarrow})\to x_i{\downarrow}\mid\alpha\in\mathcal{B},i=1,2\}$ で定義する．但し以降では混乱のない限り添え字を省略する．

命題 3.11 では静的再帰成分 C の非循環性を簡約化対を用いて示す際には $R\subseteqq\gtrsim$ を示す必要があった．この R を単純に $\mathcal{U}(C)$ に置き換えたものを示しても非循環性は保障されない．保障するためには実効規則とともに C_e を追加する必要がある．

定理 5.3 HRS R が有限分岐でかつ以下の条件を満たす簡約化対 $(\gtrsim,>)$ が存在するならば $C\in SRC(R)$ は非循環性を持つ．

（1） $\forall l\to r\in C\cup\mathcal{U}(C)\cup C_e.\,l\gtrsim r$

（2） $\exists u^\sharp\to v^\sharp\in C.\,u^\sharp>v^\sharp$

証明は後述する．

例 5.4 4 節で紹介した $SRC(R_{\mathtt{ave}})$ のうち

$$C=\left\{\mathtt{div}^\sharp(\mathtt{s}(X),\mathtt{s}(Y))\to\mathtt{div}^\sharp(\mathtt{sub}(X,Y),\mathtt{s}(Y))\right\}$$

以外の静的再帰成分は部分項基準を満たす．C の非循環性を示すためには $C\subseteqq>_\pi^{rpo}$ に加えて，例 4.6 での $R_{\mathtt{foldl}}\subseteqq\gtrsim_\pi^{rpo}$ の代わりに $\mathcal{U}(C)\cup C_e\subseteqq\gtrsim_\pi^{rpo}$ を示せばよい．すなわち

$$\begin{array}{rcl}\mathtt{div}^\sharp(\mathtt{s}(X),\mathtt{s}(Y)) & >_\pi^{rpo} & \mathtt{div}^\sharp(\mathtt{sub}(X,Y),\mathtt{s}(Y))\\ \mathtt{sub}(X,0) & \gtrsim_\pi^{rpo} & X\\ \mathtt{sub}(0,Y) & \gtrsim_\pi^{rpo} & 0\\ \mathtt{sub}(\mathtt{s}(X),\mathtt{s}(Y)) & \gtrsim_\pi^{rpo} & \mathtt{sub}(X,Y)\end{array}$$

と $C_e\subseteqq\gtrsim_\pi^{rpo}$ を示せばよい．ここで例 4.5 と同様に再帰経路順序で用いる優先順位を $\mathtt{div}\rhd\mathtt{s}\rhd\mathtt{sum}$ とし $\pi(\mathtt{sub})=[1]$ とすれば，これらの制約は全て満たされるので定理 5.3 から非循環性が示される．以上より全ての静的再帰成分の非循環性が示せたので定理 5.3 から HRS $R_{\mathtt{ave}}$ の停止性がいえる．

本節の残りでは定理 5.3 の証明を与える．以降では R を有限分岐 HRS，C を R の静的再帰成分とし，$\Delta=\{top(l)\mid l\to r\in R\backslash\mathcal{U}(C)\}$ とする．最初に証明の鍵となる解釈 I を与える．整列可能定理を用いることにより任意の空でない集合 T の最小元 $least(T)$ を与えることができる．

定義 5.5 停止性を持つ項 $t\in\mathcal{T}_\alpha$ に対し，$I(t)$ を以下で定義．

$$\left\{\begin{array}{ll}\lambda x.I(u) & t\equiv\lambda x.u\text{ のとき}\\ a(I(\overline{t_n})) & t\equiv a(\overline{t_n})\text{ かつ }a\notin\Delta\text{のとき}\\ \mathtt{c}_\alpha(a(I(\overline{t_n})),Red_\alpha(\{I(t')\mid t\to t'\})) & \\ & t\equiv a(\overline{t_n})\text{ かつ }a\in\Delta\text{のとき}\end{array}\right.$$

ここで $Red_\alpha(T)$ は $T=\emptyset$ の場合は $\bot_\alpha$ で，そうでない場合は $\mathtt{c}_\alpha(least(T),Red_\alpha(T\setminus\{least(T)\}))$ で定義される．停止性を持つ代入 θ に対し，θ^I を $\theta^I(x)=I(\theta(x))$ で定義する．

補題 5.6 $t\theta$ が停止性を持つような項 t と代入 θ に対し，$I(t\theta{\downarrow})\xrightarrow[C_e]{}{}^*I(t)\theta^I{\downarrow}\xrightarrow[C_e]{}{}^*t\theta^I{\downarrow}$.

証明 $\rhd_s$ の多重集合上への拡張である $\rhd_s^{mul}$ と $\rhd_{sub}\cup\underset{R}{\to}$ の辞書式結合による順序を用いて $(\{type(x)\mid x\in dom(\theta)\},t)$ に関する帰納法で示す．$t\equiv X(\overline{t_n})$ かつ $X\in dom(\theta)$ かつ $n>0$ の場

合のみ示す．$X\theta\downarrow = \lambda\overline{y_n}.a(\overline{u_k})$，$\sigma = \{y_i := t_i\theta\downarrow\}$ とおく．このとき $type(X) = \alpha_1 \to \cdots \to \alpha_n \to \beta \rhd_s \alpha_i = type(y_i)$ なので帰納法の仮定から $I(a(\overline{u_k})\sigma\downarrow) \xrightarrow[Ce]{*} I(a(\overline{u_k}))\sigma^I\downarrow$．また $t \rhd_{sub} t_i$ なので帰納法の仮定から $I(t_i\theta\downarrow) \xrightarrow[Ce]{*} I(t_i)\theta^I\downarrow \xrightarrow[Ce]{*} t_i\theta^I\downarrow$．よって $I(t\theta\downarrow) = I(X(\overline{t_n})\theta\downarrow) = I((\lambda\overline{y_n}.a(\overline{u_k}))(\overline{t_n\theta\downarrow})\downarrow) = I(a(\overline{u_k})\{y_i := t_i\theta\downarrow\}\downarrow) \xrightarrow[Ce]{*} I(a(\overline{u_k}))\{y_i := I(t_i\theta\downarrow)\}\downarrow \xrightarrow[Ce]{*} I(a(\overline{u_k}))\{y_i := I(t_i)\theta^I\downarrow\}\downarrow = (\lambda\overline{y_n}.I(a(\overline{u_k})))(\overline{I(t_n)\theta^I\downarrow})\downarrow = X(\overline{I(t_n)})\theta^I\downarrow = I(X(\overline{t_n}))\theta^I\downarrow = I(t)\theta^I\downarrow$．また $I(a(\overline{u_k}))\{y_i := I(t_i)\theta^I\downarrow\}\downarrow \xrightarrow[Ce]{*} I(a(\overline{u_k}))\{y_i := t_i\theta^I\downarrow\}\downarrow = \lambda\overline{y_n}.I(a(\overline{u_k}))(\overline{t_n\theta^I\downarrow})\downarrow = I(\lambda\overline{y_n}.a(\overline{u_k}))(\overline{t_n\theta^I\downarrow})\downarrow = X(\overline{t_n})\theta^I\downarrow = t\theta^I\downarrow$． □

本来この補題では $I(t\theta\downarrow) \xrightarrow[Ce]{*} t\theta^I\downarrow$ を示すことができれば十分である．実際，文献 [12] での STRS 上での実効規則の正当性証明の際には直接これを示していた．しかし HRS 上では帰納法の仮定を適用するためにより強い形で証明する必要があった．そのために非常に巧妙な帰納法が必要になった．一方この改良により文献 [12] の場合に必要だった高階変数を介した規則の実効性を考慮しなくてよくなった．

補題 5.7 $r\theta$ が停止性を持つような $l \to r \in C \cup \mathcal{U}(C)$ と代入 θ に対し，$I(r\theta\downarrow) \equiv I(r)\theta^I\downarrow \equiv r\theta^I\downarrow$．

証明 任意の $t \in Sub(r)$ で $I(t\theta\downarrow) \equiv I(t)\theta^I\downarrow \equiv t\theta^I\downarrow$ となることを示せばよい．これは $t \equiv f(\overline{t_n})$ かつ $f \in \Delta$ の場合がないことに注意して補題 5.6 と同様に示すことが出来る． □

補題 5.8 s が停止性を持ち $s \xrightarrow[R]{} t$ ならば $I(s) \xrightarrow[\mathcal{U}(C)\cup C_e]{+} I(t)$．

証明 $s \xrightarrow[R]{} t$ よりある規則 $l \to r \in R$ と文脈 $C[]$ と代入 θ が存在して $s \equiv C[l\theta]$ かつ $t \equiv C[r\theta]$．補題 5.6 と補題 5.7 を用いて $C[]$ の構造に関する帰納法で示せる． □

証明 (定理 5.3) C 中の依存鎖のみから構成され，C 中の各依存対をそれぞれ無限回含む無限依存鎖 $u_0^\sharp \to v_0^\sharp, u_1^\sharp \to v_1^\sharp, u_2^\sharp \to v_2^\sharp, \cdots$ が存在すると仮定する．依存鎖の定義より，ある代入 $\theta_0, \theta_1, \ldots$ が存在して $v_i^\sharp\theta_i\downarrow \xrightarrow[R]{*} u_{i+1}^\sharp\theta_{i+1}\downarrow$ かつ $u_i^\sharp\theta_i\downarrow$ と $v_i^\sharp\theta_i\downarrow$ は全て停止性を持つ．各 i に対し，補題 5.6,5.7,5.8 より $v_i^\sharp\theta_i^I\downarrow \equiv I(v_i^\sharp\theta_i\downarrow) \xrightarrow[\mathcal{U}(C)\cup C_e]{*} I(u_{i+1}^\sharp\theta_{i+1}\downarrow) \xrightarrow[Ce]{*} u_{i+1}^\sharp\theta_{i+1}^I\downarrow$．よって，条件 (1) より $v_i^\sharp\theta_i^I\downarrow \gtrsim u_{i+1}^\sharp\theta_{i+1}^I\downarrow \gtrsim v_{i+1}^\sharp\theta_{i+1}^I\downarrow$．また条件 (2) より無限個の j で $u_j^\sharp\theta_j^I\downarrow > v_j^\sharp\theta_j^I\downarrow$ が成立．これは $>$ の整礎性に矛盾． □

6. おわりに

本論文では STRS 上で提案されていた引数切り落とし法 [7] と実効規則の概念 [12] を HRS 上へ拡張した．また単なる拡張ではなく，文献 [5], [7] で要求されていた左堅固という引数切り落とし法の適用条件を取り除き，文献 [12] で考慮していた高階変数を介した依存性を実効規則の定義から取り除くという改良を施した．これらの改良を STRS 上に還元するのは今後の課題である．また引数切り落とし法を実効規則に組み合わせることで停止性証明法を強力にするアプローチが TRS 上で提案され [3]，STRS 上に拡張されている [7]．この成果を HRS 上に拡張するのも今後の課題である．STRS 上では文献 [6] で要求している静的依存対法の健全性を保障するための関数渡しの安全性が文献 [7] で緩和され，適用クラスが拡張されている．文献 [8] で提案された HRS 上の静的依存対法の適用条件は文献 [6] に相当するものなので STRS の場合と同様の緩和が可能であることが予測される．これも今後の課題である．

謝辞：本研究は一部，科研費#18500011，#20300010，#20500008，#21700011，及び栢森情報科学振興財団の助成を受けたものである．

References

[1] Arts,T. and Giesl,J., Termination of Term Rewriting Using Dependency Pairs, *Theoretical Computer Science*, Vol.236, pp.133–178, 2000.

[2] Blanqui,F., Jouannaud,J.-P., and Rubio,A., The Computability Path Ordering: The End of a Quest, In *Proc. of the 17th EACSL Annual Conf. on Computer Science Logic*, LNCS 5213 (CSL08), pp.1–14, 2008.

[3] Giesl,J., Thiemann,R., Schneider-Kamp,P., and Falke,S., Mechanizing and Improving Dependency Pairs, *Journal of Automated Reasoning*, Vol.37(3), pp.155–203, 2006.

[4] Hirokawa,N., and Middeldorp,A., Dependency Pairs Revisited, In *Proc. of the 15th Int. Conf. on Rewriting Techniques and Applications*, LNCS 3091 (RTA04), pp.249–268, 2004.

[5] Kusakari,K., On Proving Termination of Term Rewriting Systems with Higher-Order Variables, *IPSJ Transactions on Programming*, Vol.42, No.SIG 7 (PRO 11), pp.35–45, 2001.

[6] Kusakari,K., and Sakai,M., Enhancing Dependency Pair Method using Strong Computability in Simply-Typed Term Rewriting Systems, *Applicable Algebra in Engineering, Communication and Computing*, Vol.18, No.5, pp.407–431, 2007.

[7] Kusakari,K., and Sakai,M., Static Dependency Pair Method for Simply-Typed Term Rewriting and Related Techniques, *IEICE Transactions on Information and Systems*, Vol.E92-D, No.2, pp.235–247, 2009.

[8] Kusakari,K., Isogai,Y., Sakai,M., and Blanqui,F., Static Dependency Pair Method based on Strong Computability for Higher-Order Rewrite Systems, *IEICE Transactions on Information and Systems*, Vol.E92-D, No.10, pp.2007–2015, 2009.

[9] Mayr,R., and Nipkow,N., Higher-Order Rewrite Systems and their Confluence, *Theoretical Computer Science*, Vol.192, No.2, pp.3–29, 1998.

[10] Nipkow,N., Higher-order Critical Pairs, In *Proc. 6th Annual IEEE Symposium on Logic in Computer Science*, pp.342–349, 1991.

[11] Sakai,M., Watanabe,Y., and Sakabe,T., An Extension of the Dependency Pair Method for Proving Termination of Higher-Order Rewrite Systems, *IEICE Transactions on Information and Systems*, Vol.E84-D, No.8, pp.1025–1032, 2001.

[12] Sakurai,T., Kusakari,K., Sakai,M., Sakabe,T., and Nishida,N., Usable Rules and Labeling Product-Typed Terms for Dependency Pair Method in Simply-Typed Term Rewriting Systems, *IEICE Transactions on Information and Systems*, Vol.J90-D, No.4, pp.978–989, 2007. (in Japanese)

[13] Terese, Term Rewriting Systems, Cambridge Tracts in Theoretical Computer Science, Vol.55, *Cambridge University Press*, 2003.