無限項書き換えシステムにおける強頭部正規化可能性および一般生成性の自動反証

岩見 宗弘　青戸 等人

遅延リストやストリームといった仮想的に無限長とみなされるデータを扱う関数型プログラムの計算モデルとして，無限項書き換えシステムが提案されている．項書き換えシステムにおける停止性に対応する基本的な性質として，無限項書き換えシステムにおける強頭部正規化可能性があり，その証明法が Zantema (2008) や Endrullis ら (2009) によって報告されている．また，Endrullis ら (2010) は無限項書き換えシステムの部分クラスであるストリーム項書き換えシステムを提案し，ある十分条件のもとでのストリームの生成性判定手続きを報告している．本論文では，強頭部正規化可能性および一般生成性に対する反証手続きを提案する．提案する手続きの基本的なアイデアは，有限表現をもつ無限項である正則項の反例を構成する点にある．反証手続きの正しさを示すとともに，手続きの実装を報告する．実験の結果，自動反証法が従来知られていない例について自動反証に成功することを確認した．

Infinitary term rewriting has been proposed to model functional programs that deal with virtually infinite data structures such as streams or lazy lists. Strong head normalization is a fundamental property of infinitary term rewriting systems and methods for proving this property have been proposed by Zantema (2008) and Endrullis et al. (2009). Endrullis et al. (2010) have proposed a class of infinitary term rewriting systems—stream term rewriting systems—and they have given a decision procedure of the productivity of streams for a class of stream term rewriting systems. In this paper, we present procedures for disproving these two properties of infinitary term rewriting systems—the strong head normalization and the productivity. The basic idea of our procedure is to construct *rational* counterexamples which are infinitary terms but have finite representations. The correctness of our procedures is proved and an implementation is reported. Our experiments reveal that our procedures successfully disprove the strong head normalization and the productivity automatically for some examples for which no automated disproving procedure is known.

1　はじめに

遅延評価にもとづくデータ構造やストリームといったデータ構造を用いたプログラムでは，データを仮想的に無限長とみなすことによって見通しのよい議論が可能になる場合が少なくない．このような無限長のデータを扱う計算モデルとして，**無限項書き換えシステム**(infinitary term rewriting systems) が提案されている [11] [22] [23] [24] [31] [33] [34]．無限項書き換えシステムは，関数型プログラムの計算モデルとして知られている**項書き換えシステム**(term rewriting systems) [1] [14] [31] の拡張である．通常の項書き換えシステムとの違いは，通常の項書き換えシステムが有限の大きさの項しか対象にしないのとは対照的に，無限項書き換えシステムでは大きさが無限であるよ

Disproving Strong Head Normalization and General Productivity Automatically in Infinitary Term Rewriting Systems

Munehiro Iwami, 島根大学総合理工学部数理・情報システム学科, Dept. of Mathematics and Computer Science, Interdisciplinary Faculty of Science and Engineering, Shimane University.

Takahito Aoto, 東北大学電気通信研究所, RIEC, Tohoku University.

コンピュータソフトウェア, Vol.27, No.0 (2010), pp.1–5.

[研究論文] 2010 年 1 月 7 日受付.

本論文は以下の既発表論文をもとに発展させたものである: 岩見宗弘，青戸等人，無限項書き換えシステムにおける強頭部正規化可能性の反証手続き，第 12 回プログラミングおよびプログラミング言語ワークショップ論文集, pp.261–273, 2010.

うな項 (**無限項**) も対象とする点である．

項書き換えシステムにもとづく様々な検証法において**停止性**(termination) は非常に有用な性質として知られており，様々な停止性の証明法や反証法が提案されている [1] [14] [31]．一方，通常の項書き換えシステムとは対照的に，無限項書き換えシステムでは停止性を考える意味があまりない．無限項を対象に含める場合には停止性をもたないことが通常のためである．例えば，(無限) 項書き換えシステム $\mathcal{R} = \{\mathsf{f}(x) \to \mathsf{g}(x)\}$ は有限項上では停止性をもつが，無限項上では $\mathsf{f}(\mathsf{f}(\mathsf{f}(\cdots))) \to_{\mathcal{R}} \mathsf{g}(\mathsf{f}(\mathsf{f}(\cdots))) \to_{\mathcal{R}} \mathsf{g}(\mathsf{g}(\mathsf{f}(\mathsf{f}(\cdots)))) \to_{\mathcal{R}} \cdots$ といった無限書き換え列を持ち，停止性をもたない．

このため，停止性に替わる無限項書き換えシステムにおける基本的な性質として，**強頭部正規化可能性**†1(strong head normalization) が考えられている [11] [24] [33]．強頭部正規化可能性は，項のどの固定位置も有限回しか書き換えが可能でないという性質であり，これによって，ストリームのプレフィックスのように観測可能な有限部分のデータについては，書き換えに対して安定な部分が求まることが保証される．先程の項書き換えシステム $\mathcal{R} = \{\mathsf{f}(x) \to \mathsf{g}(x)\}$ も強頭部正規化可能性をもつ．なぜなら，項のどの位置も高々1 回しか書き換え可能ではないからである．

無限項書き換えシステムが左線形 (どの書き換え規則の左辺にも同じ変数は 2 回以上出現しない) の場合，強頭部正規化可能性は ω-**強頭部正規化可能性**(ω-strong head normalization) と同等であることが知られている [23] [31]．ω-強頭部正規化可能性は，長さ ω(最小の無限順序数) の書き換え列に限定して強頭部正規化可能性を保証する性質である．近年，Zantema [33] は ω-強頭部正規化可能性に対して意味論にもとづく特徴付けを示し，行列解釈を用いた ω-強頭部正規化可能性の自動証明法を与えている．そして，組合せ子 [2] [6] [29] の書き換え規則のいくつかについて，ω-強頭部正規化可能性の自動証明に成功することを報告している．また，Endrullis ら [11] は，木オートマトンを使って，初期項集合を限定した局所 ω-強頭部正規化可能性の証明手法を与えている．

非 ω-強頭部正規化可能性については，縮退規則 (右辺が変数となっている書き換え規則) をもつ無限項書き換えシステムが ω-強頭部正規化可能性をもたないことが知られているが，それ以外の十分条件や証明法は従来あまり知られていない．その一方で，よく知られた組合せ子の書き換え規則の多くは，非縮退であるにもかかわらず，ω-強頭部正規化可能性をもたないことが岩見 [19] によって報告されている．このため，縮退規則をもたない無限項書き換えシステムに対しても適用可能であるような ω-強頭部正規化可能性の自動反証法は，より強力な ω-強頭部正規化可能性判定法を実現するためには不可欠と考えられる．

ストリームを生成するプログラムを記述したときに，ストリームの先頭要素をいくつ消費しても，その次の要素が必ず求まるという性質は非常に望ましい性質である．このような性質や，より一般的に，与えられた仕様による (無限の) データの定義可能性は，**生成性**(productivity) とよばれる．生成性についての議論は Dijkstra [8] や Sijtsma [28] によって開始された．特に，後者は関数型プログラムにおけるいくつかの生成性を表示的意味にもとづいて定義し，生成性および非生成性の十分条件を与えた．文献 [3] [15] [30] は抽象解釈や型システムにもとづく生成性の解析法を与えている．

Endrullis ら [9] [10] は，無限項書き換えシステムの一種であるストリーム項書き換えシステム (stream term rewriting systems) 上において，ストリーム型項の生成性の概念を提案するとともに，ある十分条件のもとでの生成性判定手続きを提案した．そして，提案した判定手続きが [3] [15] [30] の解析法を一般化していることを示した．また，石原 [17] [18] はアルゴリズム的システム (algorithmic systems) を提案し，より一般的な無限項書き換えシステムにおける生成性の特徴付けを報告している．Zantema [34] は停止性検証法を応用して，ストリーム項書き換えシステムにおける定義可能性の検証法を提案するとともに，定義可能性が生成性の必要条件になっていることを示している．

本論文では，縮退規則をもたない無限項書き換えシ

†1 無限停止性 (infinitary normalization) あるいは強収束性 (strong convergence) とよばれることもある．

ステムに対しても適用可能であるような ω-強頭部正規化可能性の反証手続きを提案する．また，Endrullis ら [10] の定義にもとづく生成性を一般化した性質である一般生成性を定義し，その自動反証法を提案する．これらの反証手続きの正しさ (健全性) を示すとともに，手続きの実装および実験を報告する．

本論文で提案する 2 つの反証手続きは，共に無限項書き換えシステムの性質に対する反証手続きであるというだけでなく，有限表現をもつ無限項である正則項 (regular または rational term) の反例を構成するという共通のアイデアにもとづいている．この正則項の反例構成の鍵となっているのは，正則項の単一化手続きである．従来知られていた正則項の単一化手続きは，単一化可能性の決定可能性を示すことを目的に提案されたものであるため手続きの効率は考慮されておらず，実装に用いるには適していない．そこで，本論文では，これらの反証手続きに先立ち，正則項の効率的な単一化手続きの実現法をあきらかにする．

無限項書き換えシステムにおいて正則項を用いた研究がいくつか知られている．Kennaway ら [22] はグラフ書き換えが正則項上の無限簡約に対応することを示している．また，Inverardi と Zilli [16] は正則項上の項書き換えの枠組みを示し，無限項書き換えシステムの強収束列を模倣する正則項書き換えシステムの変換を提案している．しかしながら，正則項を用いることによって，ω-強頭部正規化可能性や生成性の反例を構成する試みは著者らが知る限り従来提案されていない．

本論文の構成は以下の通りである．第 2 節では，無限項書き換えシステムに関する定義や記法を与え，続いて，第 3 節で第 4,5 節で与える反証手続きで用いる正則項およびその効率的な単一化手続きについて説明する．第 4 節において，ω-強頭部正規化可能性の反証手続きを与えるとともにその健全性を証明する．第 5 節において，ストリーム項書き換えシステムに対する一般生成性の反証手続きを与えるとともにその健全性を示す．その詳細な証明は付録で与える．第 6 節では，2 つの反証手続きの実装について報告し，実験結果を示す．第 7 節は結論である．

2 準備

本節では，本論文で使用する無限項書き換えシステムに関する定義や記法を与える．詳細は文献 [11] [24] [31] 等を参考にして頂きたい．

関数記号の集合を $\mathcal{F}$，**変数**の可算無限集合を $\mathcal{V}$ と記す ($\mathcal{F} \cap \mathcal{V} = \emptyset$ とする)．0 引数の関数記号を**定数**とよび，n 引数の関数記号の集合を $\mathcal{F}_n$ と記す．$\mathbb{N}^+$ を正整数集合とし，正整数の有限列の集合を $\mathbb{N}^{+*}$ と記す．有限列 $p \in \mathbb{N}^{+*}$ の長さを $|p|$，有限列 $p, q \in \mathbb{N}^{+*}$ の連結を $p.q$ と記す．部分関数 $t : \mathbb{N}^{+*} \to \mathcal{F} \cup \mathcal{V}$ のうち，以下の条件を満たすものを $\mathcal{F}, \mathcal{V}$ 上の**項**とよぶ: (1) $t(\epsilon) \in \mathcal{F} \cup \mathcal{V}$; (2) 任意の $p \in \mathbb{N}^{+*}$ について，$t(p.i) \in \mathcal{F} \cup \mathcal{V} \Longleftrightarrow t(p) \in \mathcal{F}_n$ かつ $1 \leq i \leq n$．ここで，ϵ は空列を表わす．$\mathcal{F}, \mathcal{V}$ 上の項の集合を $\mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ と表す．

項 $t \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ の定義域 $\mathrm{Pos}(t) = \{p \in \mathbb{N}^{+*} \mid t(p) \in \mathcal{F} \cup \mathcal{V}\}$ の要素を t における**位置**とよぶ．特に，位置 ϵ を**根位置**とよぶ．$t(p)$ を項 t の位置 p に出現する記号とよび，特に $t(\epsilon)$ を**根記号**とよぶ．$p \notin \mathrm{Pos}(s)$ なる $p \in \mathbb{N}^{+*}$ に対して，$s(p) = \bot$ と便宜的に定義する．ただし，$\bot$ は $\bot \notin \mathcal{F} \cup \mathcal{V}$ なる定数とする．このとき，$s = t \Leftrightarrow \forall p \in \mathrm{Pos}(s).\ s(p) = t(p)$ が成立する．項 t に出現する変数集合を $\mathcal{V}(t)$ と表す．$\mathcal{V}(t) = \emptyset$ なる項を**基底項**という．位置集合 $\mathrm{Pos}(t)$ が有限集合であるとき，項 t を**有限**であるという．有限項の集合を $\mathcal{T}_{fin}(\mathcal{F}, \mathcal{V})$ と記す．有限項と特に区別するときには，項を**無限項**とよぶこともある．位置 $p \in \mathrm{Pos}(t)$ における項 t の**部分項** $t|_p$ を $t|_p(q) = t(p.q)$ により定義する．特に，$p \neq \epsilon$ のとき，$t|_p$ を t の**真部分項**とよび，$t(p) \notin \mathcal{V}$ のとき，$t|_p$ を t の**非変数部分項**とよぶ．関数記号 $f \in \mathcal{F}_n$ および項 $t_1, \ldots, t_n \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ に対して，(1) $t(\epsilon) = f$; (2) $t(i.p) = t_i(p)$ $(1 \leq i \leq n, p \in \mathbb{N}^{+*})$ により定義される項 t を $f(t_1, \ldots, t_n)$ と記す．位置集合上の**接頭辞順序**を $p \leq q \Leftrightarrow \exists r \in \mathbb{N}^{+*}.\ q = p.r$ により定義する．このとき，この r を $q \backslash p$ と表わす．$p \leq q$ となるとき，p は q の**上に**(q は p の**下に**) 位置するという．ある位置 q および正整数 $i < j$ が存在して，$q.i \leq p_1$，$q.j \leq p_2$ となるとき，p_1 は p_2 の**左に**(p_2 は p_1 の**右に**) 位置するという．$p < q \Leftrightarrow$

$p \leq q \land p \neq q$，$p \parallel q \Leftrightarrow p \not\leq q \land q \not\leq p$ と定義する．$p \parallel q$ のとき，p と q は**並列**に位置するという．

関数 $\sigma : \mathcal{V} \to \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ を**代入**とよぶ．代入 σ を項 $t \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ に適用した結果 $\sigma(t)$ を，以下のように定義する: $\sigma(t)(p) = \sigma(t(p_0))(p_1)$ ($p = p_0.p_1$ かつ $t(p_0) \in \mathcal{V}$ を満たす $p_0, p_1 \in \mathbb{N}^{+*}$ が存在する場合); $\sigma(t)(p) = t(p)$ (それ以外の場合)．直観的には，$\sigma(t)$ は t に出現する変数 $x \in \mathcal{V}$ を $\sigma(x)$ により置き換えた結果得られる項を表わす．代入 σ の定義域 $\{x \in \mathcal{V} \mid \sigma(x) \neq x\}$ を $\mathrm{dom}(\sigma)$ と記す．$\mathrm{dom}(\sigma) = \{x_1, \ldots, x_n\}$，$\sigma(x_i) = u_i$ なる代入を $\{x_1 := u_1, \ldots, x_n := u_n\}$ とも記す．$\mathrm{dom}(\sigma)$ が有限かつ任意の $x \in \mathrm{dom}(\sigma)$ について $\sigma(x) \in \mathcal{T}_{fin}(\mathcal{F}, \mathcal{V})$ なるとき，σ を**有限代入**とよぶ．$\sigma(t)$ を $t\sigma$ とも記す．項 s, t について，$s\sigma = t$ となる代入 σ が存在するとき，項 s は t と**照合可能**であるという．そのような代入 σ のうち，$\mathrm{dom}(\sigma) = \mathcal{V}(s)$ なる σ を $\mathrm{match}_{inf}(s, t)$ と記す．また，そのような代入 σ を使わないときは，$\mathrm{match}_{inf}(s, t)$ と書いて単に項 s が t と照合可能であることを表わすものとする．代入 σ の集合 V への**制限**$\sigma\restriction_V$ を次のように定義する: $x \in V$ のとき $\sigma\restriction_V(x) = \sigma(x)$; それ以外のとき $\sigma\restriction_V(x) = x$．$\sigma_1, \sigma_2$ を $\mathrm{dom}(\sigma_1) \cap \mathrm{dom}(\sigma_2) = \emptyset$ なる代入とする．このとき，$\mathrm{dom}(\sigma_1 \cup \sigma_2) = \mathrm{dom}(\sigma_1) \cup \mathrm{dom}(\sigma_2)$ である代入 $\sigma_1 \cup \sigma_2$ を次のように定義する：$i = 1, 2$ に対して $x \in \mathrm{dom}(\sigma_i)$ のとき $(\sigma_1 \cup \sigma_2)(x) = \sigma_i(x)$．代入の**合成**を $\circ$ によって表す，すなわち，すべての変数 $x \in \mathcal{V}$ に対して $(\sigma \circ \tau)(x) = \sigma(\tau(x))$．

$\Box \notin \mathcal{F} \cup \mathcal{V}$ なる定数 $\Box$ (**ホール**とよぶ) を考える．$\mathcal{T}_{inf}(\mathcal{F} \cup \{\Box\}, \mathcal{V})$ の要素 C のうち，$\{p \in \mathbb{N}^{+*} \mid C(p) = \Box\}$ が有限集合となるものを**文脈**とよぶ．$C \in \mathcal{T}_{inf}(\mathcal{F} \cup \{\Box\}, \mathcal{V})$ が $\{p_1, \ldots, p_n\} = \{p \in \mathbb{N}^{+*} \mid C(p) = \Box\}$ かつ任意の $i < j$ について p_i が p_j より左に位置するとき，C を $C[, \ldots,]_{p_1, \ldots, p_n}$ と記す．文脈 $C[, \ldots,]_{p_1, \ldots, p_n}$ と項 $t_1, \ldots, t_n \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ に対して，項 $t = C[t_1, \ldots, t_n]_{p_1, \ldots, p_n}$ を以下のように定義する: $t(p) = t_i(q)$ ($\exists i, q.\ p = p_i.q$ の場合); $t(p) = C(p)$ (それ以外の場合)．直観的には，$C[t_1, \ldots, t_n]_{p_1, \ldots, p_n}$ は文脈 C に出現するホールを左から順に $t_1, \ldots, t_n$ に置き換えて得られる項を表わす．また，ホールの出現を丁度 1 つもつ文脈を $C[\,]_p$ と記す．さらに，項 s の位置 p の部分項 $s|_p$ をホールに置き換えて得られる文脈を $s[\,]_p$ と表わす．

等式を $s \approx t$ と記す．ただし，ここで $s, t \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ とする．等式 $s \approx t$ の左辺は s，右辺は t をさし，等式の右辺と左辺を区別する．代入 σ を等式集合 $\{x \approx \sigma(x) \mid x \in \mathrm{dom}(\sigma)\}$ と同一視することがある．等式 $l \approx r$ が以下の条件を満たすとき，これを**書き換え規則**とよび，$l \to r$ と記す: (1) l は有限項，(2) $l(\epsilon) \notin \mathcal{V}$，(3) $\mathcal{V}(r) \subseteq \mathcal{V}(l)$．書き換え規則の集合 $\mathcal{R}$ を**無限項書き換えシステム**(infinitary term rewriting systems) とよぶ．項における任意の変数の出現が高々1 回のとき, **線形**であるといい，任意の書き換え規則の左辺が線形であるとき, $\mathcal{R}$ を**左線形**であるという．ある書き換え規則の右辺が変数であるとき $\mathcal{R}$ は**縮退性**をもつという．任意の書き換え規則の右辺が有限項であるとき, $\mathcal{R}$ を**右有限**であるという．本論文では右有限な無限項書き換えシステムのみを取り扱う．**定義記号**とは，書き換え規則の左辺の根記号となっている関数記号のことをいい，それ以外の関数記号を**構成子**とよぶ．定義記号集合を $\mathcal{D}$，構成子集合を $\mathcal{C}$ と記す．すなわち，$\mathcal{D} = \{l(\epsilon) \mid l \to r \in \mathcal{R}\}$，$\mathcal{C} = \mathcal{F} \setminus \mathcal{D}$．$\mathcal{T}_{inf}(\mathcal{C}, \mathcal{V})$ の要素を**構成子項**とよぶ．任意の書き換え規則について，その左辺の真部分項がすべて構成子項となっているとき，$\mathcal{R}$ を**構成子システム**とよぶ．

項 $s, t \in \mathcal{T}_{inf}(\mathcal{F}, \mathcal{V})$ に対して，書き換え規則 $l \to r \in \mathcal{R}$, 代入 σ, 文脈 $C[\,]_p$ が存在して, $s = C[l\sigma]_p$ かつ $t = C[r\sigma]_p$ となるとき，$s \to_{p, \mathcal{R}} t$ と記す．これを位置 p における項 s から項 t への**簡約**または**書き換えステップ**とよぶ．根位置 ϵ における簡約を**根書き換え**とよぶ．部分項 $s|_p$ を**リデックス**とよぶ．文脈からあきらかな場合もしくは必要がない場合は $p, \mathcal{R}$ を省略する．$\to_{\mathcal{R}}$ の反射推移閉包を $\to^*_{\mathcal{R}}$ と記し，$\to_{\mathcal{R}}$ の推移閉包を $\to^+_{\mathcal{R}}$ と記す．項 s が**正規**であるとは，$s \to_{\mathcal{R}} t$ なる項 t が存在しないときをいう．正規項の集合を NF と記す．構成子項である正規項を**構成子正規項**とよぶ．項 s の**正規形**とは，$s \to^*_{\mathcal{R}} t \in \mathrm{NF}$ なる項 t をいう．

等式集合 E の**単一化子**とは，任意の $s \approx t \in E$ に

ついて $s\sigma = t\sigma$ となる代入 σ をいう．等式集合 E の単一化子の集合を $\mathrm{Unif}_{\mathit{inf}}(E)$ と記す．$\mathrm{Unif}_{\mathit{inf}}(E) \neq \emptyset$ であるとき，等式集合 E は**単一化可能**であるという．等式集合 $\{s \approx t\}$ が単一化可能であるとき，s と t が単一化可能であるという．単一化可能性を判定する問題を，**単一化問題**という．代入上の擬順序 $\precsim$ を $\theta \precsim \eta \iff$ ある代入 ρ が存在して $\eta = \rho \circ \theta$ により定めるとき，単一化子のうち擬順序 $\precsim$ に関して極小となる単一化子を**最汎単一化子**とよぶ．等式集合 E が単一化可能であるときには，最汎単一化子が存在し，同値関係 $\succsim \cap \precsim$ に関して一意に定まる (cf. [5])．等式集合 E の最汎単一化子 (の 1 つ) を $\mathrm{mgu}_{\mathit{inf}}(E)$ と記す．特に，$E = \{s \approx t\}$ のとき，$\mathrm{mgu}_{\mathit{inf}}(E)$ を $\mathrm{mgu}_{\mathit{inf}}(s,t)$ と記す．等式集合 E と代入 σ に対して，$E\sigma = \{s\sigma \approx t\sigma \mid s \approx t \in E\}$ と定義する．しばしば，代入を限定して単一化可能性を考えることがある．特に，等式集合 E が**有限項上で単一化可能である**とは，有限代入であるような $\mathrm{Unif}_{\mathit{inf}}(E)$ の要素があることをいう．特に区別する場合には，単一化可能であることを**無限項上で単一化可能である**ともいう．

$l_1 \to r_1, l_2 \to r_2$ を項書き換えシステム $\mathcal{R}$ の書き換え規則とする．ただし，$l_1 \to r_1$ と $l_2 \to r_2$ が等しい場合を含むとする．$\mathcal{V}(l'_1) \cap \mathcal{V}(l_2) = \emptyset$ となるように $l_1 \to r_1$ から変数の名前替えを行って得られる書き換え規則を $l'_1 \to r'_1$ とおく．l'_1 と l_2 の非変数部分項 $l_2|_p$ が有限項上で単一化可能となるとき，$l_1 \to r_1$ と $l_2 \to r_2$ は**重なりをもつ**といい，$l_1 \to r_1$ と $l_2 \to r_2$ が等しくないまたは $p \neq \epsilon$ のとき，**非自明な重なりをもつ**という．非自明な重なりをもつ書き換え規則が $\mathcal{R}$ に含まれるとき，$\mathcal{R}$ は重なりをもつという．左線形かつ重なりをもたないとき，$\mathcal{R}$ は**直交性**をもつという．

項 $s \in \mathcal{T}_{\mathit{inf}}(\mathcal{F},\mathcal{V})$ に対して，書き換え規則 $l \to r \in \mathcal{R}$ が存在し，$\mathcal{V}(l) \cap \mathcal{V}(s) = \emptyset$ となるように $l \to r$ から変数の名前替えを行って得られる書き換え規則を $l' \to r'$ とおいたとき，s の非変数部分項 $s|_p$ と l' が単一化可能であるとする．$\theta = \mathrm{mgu}_{\mathit{inf}}(s|_p, l')$，$t = s[r']_p\theta$ とおくとき，$s \rightsquigarrow_{p,\theta,\mathcal{R}} t$ と記す．これを位置 p における項 s から項 t への**ナローイングステップ**とよぶ．$p = \epsilon$ の場合を**根ナローイング**とよぶ．文脈からあきらかな場合もしくは必要がない場合は $p, \mathcal{R}$ は省略する．$s_1 \rightsquigarrow_{\theta_1} s_2 \rightsquigarrow_{\theta_2} \cdots \rightsquigarrow_{\theta_{n-1}} s_n$ となるとき，$\sigma = \theta_{n-1} \circ \cdots \circ \theta_1$ に対して $s_1 \rightsquigarrow^*_\sigma s_n$ と記す．$s \rightsquigarrow^*_\sigma t$ ならば，$s\sigma \to^*_{\mathcal{R}} t$ となる．また，s, t, θ が有限であるとき，$s \rightsquigarrow_{p,\theta,\mathcal{R}} t$ を**有限ナローイングステップ**とよぶ．

最小の無限順序数を ω で表す．項 s と t の**距離**$d(s,t)$ を次のように定義する: $s = t$ のとき $d(s,t) = 0$; $s \neq t$ のとき $d(s,t) = 2^{-k}$ $(k = \min\{|p| \mid s(p) \neq t(p)\})$．実数列 $\alpha_1, \alpha_2, \alpha_3, \ldots$ が極限 α_ω に収束するとは，次の式が成立するときをいう: $\forall\varepsilon > 0, \exists n_0, \forall n > n_0.\ |\alpha_n - \alpha_\omega| < \varepsilon$．このとき，$\lim_{n\to\omega} \alpha_n = \alpha_\omega$ と表す．また，実数列 $\alpha_1, \alpha_2, \alpha_3, \ldots$ が無限大に発散するとは，次の式が成立するときをいう: $\forall\varepsilon > 0, \exists n_0, \forall n > n_0.\ \varepsilon < \alpha_n$．このとき，$\lim_{n\to\omega} \alpha_n = \infty$ と表す．

α を $\alpha > 0$ なる順序数とする．関数 $A : \alpha \to \mathcal{T}_{\mathit{inf}}(\mathcal{F},\mathcal{V})$ が任意の β $(0 \leq \beta + 1 < \alpha)$ について $A(\beta) \to_{\mathcal{R}} A(\beta + 1)$ を満たすとき，A を**書き換え列**(**超限書き換え列**, transfinite reduction) という．ここで，$\alpha = \{\beta \mid \beta < \alpha\}$ であることに注意する [7]．特に $\alpha < \omega$ であるとき A を**有限書き換え列**，$\alpha = \omega$ であるとき A を**無限書き換え列**とよぶ．ただし，本論文で用いるのは有限もしくは無限の書き換え列のみである．文脈からあきらかな場合，$A(i) = t_i$ $(0 \leq i \leq n)$ なる有限書き換え列 $A : n + 1 \to \mathcal{T}_{\mathit{inf}}(\mathcal{F},\mathcal{V})$ を $t_0 \to_{\mathcal{R}} t_1 \to_{\mathcal{R}} \cdots \to_{\mathcal{R}} t_n$ と書き，$A(i) = t_i$ $(0 \leq i < \omega)$ なる無限書き換え列 $A : \omega \to \mathcal{T}_{\mathit{inf}}(\mathcal{F},\mathcal{V})$ を $t_0 \to_{\mathcal{R}} t_1 \to_{\mathcal{R}} \cdots$ と書く．

無限書き換え列 $t_0 \to_{p_0,\mathcal{R}} t_1 \to_{p_1,\mathcal{R}} t_2 \to_{p_2,\mathcal{R}} \cdots$ が**極限 t_ω に弱収束する**(weakly convergent to t_ω) とは $\lim_{n\to\omega} d(t_n, t_\omega) = 0$ であるときをいい，このような極限が存在するとき，無限書き換え列は**弱収束性**をもつ (weakly convergent) という．また，$\lim_{n\to\omega} |p_n| = \infty$ であるとき，無限書き換え列は**強収束性**をもつ (strongly convergent) という．強収束な無限書き換え列は弱収束となる (cf. [31])．強収束な無限書き換え列 $t_0 \to_{p_0,\mathcal{R}} t_1 \to_{p_1,\mathcal{R}} t_2 \to_{p_2,\mathcal{R}} \cdots$ の極限が t_ω となるとき，$t_0 \to^\omega_{\mathcal{R}} t_\omega$ と記す．例えば，$\mathcal{R} = \{\mathsf{a} \to \mathsf{f}(\mathsf{a})\}$ とおくとき，$\mathsf{a} \to_{\mathcal{R}} \mathsf{f}(\mathsf{a}) \to_{\mathcal{R}} \mathsf{f}(\mathsf{f}(\mathsf{a})) \to_{\mathcal{R}} \cdots$ は強収束な無限書き換え列であり，そ

の極限は $\mathsf{f}(\mathsf{f}(\cdots))$ となる．$\mathcal{R}=\{\mathsf{f}(x)\to\mathsf{f}(x)\}$ とおくとき，$\mathsf{f}(\mathsf{a})\to_{\mathcal{R}}\mathsf{f}(\mathsf{a})\to_{\mathcal{R}}\cdots$ は弱収束だが強収束でない無限書き換え列である．項 s が項 t の **ω-正規形**(ω-normal form) であるとは，s が t の正規形であるかまたは t から始まる強収束な無限書き換え列が存在して，その極限が s となるときをいう．

3 正則項の有限表現にもとづく単一化手続き

無限項を計算機上で扱うためには，有限表現をもつものに制限する必要がある．そのような条件を保証する性質の一つとして，正則性が知られている．本節では，第 4 節および第 5 節で与える反証手続きの基礎となる正則項の単一化手続きについて説明する．

まず始めに，正則項の定義およびその性質について説明する．

定義 1 (正則項[5]) 項 t が**正則**(regular または rational) であるとは，t の部分項集合が有限集合であるときをいう．

例 2 (正則項) 位置 p^n を $p^0=\epsilon$; $p^{i+1}=p^i.p$ により定義する．$\mathsf{f}\in\mathcal{F}_1$ とし，無限項 t を以下のように定める．$t(1^n)=\mathsf{f}$ $(\forall n\geq 0)$．このとき，$\mathrm{Pos}(t)=\{\epsilon,1,11,111,\ldots\}=\{1^n\mid n\geq 0\}$．また，$t|_{1^n}=t$ より，無限項 t の任意の部分項は t のみである．よって，t は正則項である．直観的には，t は $\mathsf{f}(\mathsf{f}(\cdots))$ なる無限項を表わす．

有限項はあきらかに正則であることに注意する．代入 θ のうち，$\mathrm{dom}(\theta)$ が有限集合，かつ，任意の変数 $x\in\mathrm{dom}(\theta)$ について $\theta(x)$ が正則となるものを**正則代入**とよぶ．容易にわかるように，正則項は正則代入の適用に関して閉じており，代入の正則性は関数合成 $\circ$ によって保存される．

正則項の単一化について，以下の事実が知られている．

命題 3 (正則項の単一化[5]) 正則項の無限項上での単一化問題は決定可能であり，単一化可能であるときに最汎単一化子を求めるアルゴリズムが存在する．また，最汎単一化子は正則代入となる． □

以下では，慣例に従って,「単一化問題を解く」とは単一化可能性を判定するとともに単一化可能であるときに最汎単一化子を求めることとし,「単一化手続き」とは単一化問題を解く手続きを指すものとする．

有限項に対する無限項上の単一化手続きは，その効率性や拡張性から，論理型プロラミンング言語 Prolog II の計算原理として用いられた[4]．正則項の同値問題や照合問題は，右辺または両辺の項に含まれる変数を定数とみなすことによって単一化問題に帰着できるので，これらも決定可能である．命題 3 より，s，t を正則項とするとき，$\mathrm{mgu}_{\mathit{inf}}(s,t)$ および $\mathrm{match}_{\mathit{inf}}(s,t)$ が存在すれば，正則代入であることに注意する．なお，有限項は正則項であるから，有限項の無限項上での最汎単一化子も正則代入となるが，一般にはその最汎単一化子は有限代入とならず，また，単一化可能性も有限項上と無限項上では異なることに注意する．例えば，無限項上では，$\mathrm{mgu}_{\mathit{inf}}(\mathsf{g}(x,x),\mathsf{g}(x,\mathsf{f}(x)))=\{x:=\mathsf{f}(\mathsf{f}(\cdots))\}$ となるが，有限項上では，有限項 $\mathsf{g}(x,x)$ と $\mathsf{g}(x,\mathsf{f}(x))$ は単一化可能ではない．

正則項を計算機上で扱うには，正則項を有限の記号表現で表わすことが必要である．以下では，正則項の有限表現について説明する．

定義 4 (再帰式表現) 有限代入 $\{x_1:=t_1,\ldots,x_n:=t_n\}$ が以下の条件を満たすとき，これを**再帰式表現**とよび，$[x_1:=t_1,\ldots,x_n:=t_n]$ と記す:

$$\begin{array}{l}\neg\exists i_1,\ldots,i_k\in\{1,\ldots,n\}.\\ \qquad(\forall 1\leq j<k.\ t_{i_j}=x_{i_{j+1}})\wedge t_{i_k}=x_{i_1}\end{array}\tag{1}$$

再帰式表現を用いて，正則項を表現することができる．

定義 5 (再帰式表現の解) $\theta=[x_1:=t_1,\ldots,x_n:=t_n]$ を再帰式表現とし，任意の $1\leq i\leq n$ について，$C_i[x_{i_1},\ldots,x_{i_{k_i}}]_{p_{i_1},\ldots,p_{i_{k_i}}}=t_i$ とする．ただし，ここで，文脈 C_i には変数 $x_1,\ldots,x_n$ が出現しないとする．このとき，項 $\theta^\star(x_1),\ldots,\theta^\star(x_n)$ を以下のように相互再帰的に定義する．$\theta^\star(x_i)(p)=$

$$\begin{cases}C_i(p) & (p\in\mathrm{Pos}(C_i)\wedge C_i|_p\neq\Box\ \text{の場合})\\ \theta^\star(x_{i_j})(q) & (\exists j,q.\ p=p_{i_j}.q\ \text{の場合})\end{cases}$$

項 $\theta^\star(x_1),\ldots,\theta^\star(x_n)$ を再帰式表現の**解**とよぶ．

再帰式表現の条件 (1) より $\theta^\star(x_i)$ は定義として成立している (well-defined である) こと，また，

$C_i|_{p_{i_j}} = \square$ より項 $\theta^\star(x_1), \ldots, \theta^\star(x_n)$ は一意に定まることに注意する．定義よりあきらかに，$\theta^\star$ は $\mathrm{dom}(\theta^\star) = \{x_1, \ldots, x_n\}$ なる正則代入となることに注意する．

また，任意の再帰式表現 θ に対して，$\theta^\star = \delta^\star$ となる再帰式表現 δ で以下の条件を満たすものが存在することに注意する: $\forall x \in \mathrm{dom}(\delta).\ \delta(x) \notin \mathrm{dom}(\delta)$.

例 6 (再帰式表現) 有限代入 $\theta = \{x := \mathsf{f}(x), y := x\}$ は再帰式表現であり，有限代入 $\theta' = \{x := \mathsf{f}(x), y := z, z := y\}$ は再帰式表現でない．また，任意の $n \geq 0$ について $\theta^\star(x)(1^n) = \theta^\star(y)(1^n) = \mathsf{f}$ であるから，$\theta^\star(x) = \theta^\star(y) = \mathsf{f}(\mathsf{f}(\cdots))$ となる．$\delta = [x := \mathsf{f}(x), y := \mathsf{f}(x)]$ をとると，$\delta^\star = \theta^\star$ かつ $\forall x \in \mathrm{dom}(\delta).\ \delta(x) \notin \mathrm{dom}(\delta)$ が成立する．再帰式表現 $\eta = [x := \mathsf{f}(y), y := \mathsf{g}(x)]$ に対して，$\eta^\star(x) = \mathsf{f}(\mathsf{g}(\mathsf{f}(\mathsf{g}(\cdots))))$，$\eta^\star(y) = \mathsf{g}(\mathsf{f}(\mathsf{g}(\mathsf{f}(\cdots))))$ となる．

再帰式表現について，以下の性質が成立する．

命題 7 (正則項と再帰式表現[4][5]) 再帰式表現の解は正則項である．また，任意の正則項はある再帰式表現の解となる． $\square$

つまり，任意の正則項は，再帰式表現 θ を用いて，$\theta^\star(x)$ の形で表わすことができる．これを**正則項の再帰式表現**とよぶ．ここで，正則項の再帰式表現は必ずしも一通りとは限らないことに注意しておく．例えば，正則項 $\mathsf{f}(\mathsf{f}(\cdots))$ は $[x := \mathsf{f}(x)]^\star(x)$ とも $[x := \mathsf{f}(\mathsf{f}(x))]^\star(x)$ とも表現できる．

特定の形の正則項については，以下のように，より簡明な表記法を用いる．(1) f^ω は $[x := f(x)]^\star(x)$ を意味する．(2) $\mu x.t$(ただし，$x \neq t$ とする) は $[x := t]^\star(x)$ を意味する．慣習に従って，$\mu x.$ は関数適用より結合力が弱いものとして括弧を省略する．例えば，$[x := \mathsf{g}(y, x), y := \mathsf{f}(y)]^\star(x) = \mu x.\mathsf{g}(\mathsf{f}^\omega, x)$ である．

先に述べたように，正則項の単一化手続きは文献[4][5]で与えられている．しかしながら，前者は Huet のアルゴリズム[13]を元にしたもので，与えられている単一化手続きは必ずしも効率が保証されているものではない，すなわち，時間計算量は線形ではない[4]．後者が与えているのは理論的な証明を目的としたものである．

一方，Jaffer [20] は，文献[25][26]の手続きを拡張して，有限項の等式の有限集合 $T = \{s_1 \approx t_1, \ldots, s_n \approx t_n\}$ が与えられたときに，無限項上の T の単一化問題を解く効率が良い手続きを与えている．その手続きの時間計算量は $\mathsf{O}(n * F(n))$ であり，実用的には線形である[20]．

以下では，Jaffer の手続き ($\texttt{unif-proc}_J(T)$ と記す) にもとづいて，再帰式表現で与えられた正則項の単一化手続きが実現できることを示す．

この結果を示す鍵となるのは，再帰式表現 $\theta = [x_1 := t_1, \ldots, x_n := t_n]$ とそれに対応する等式集合 $\{x_1 \approx t_1, \ldots, x_n \approx t_n\}$ の最汎単一化子との関係である．

例 8 (再帰式表現と最汎単一化子) 再帰式表現 $\eta = [x := \mathsf{f}(y), y := \mathsf{g}(x)]$ に対応する等式集合 $E = \{x \approx \mathsf{f}(y), y \approx \mathsf{g}(x)\}$ を考える．E の単一化を試みると，$E' = \{y \approx \mathsf{g}(\mathsf{f}(y))\}$ に変形され，E' の解は $y := \mathsf{g}(\mathsf{f}(\mathsf{g}(\mathsf{f}(\cdots))))$ であるから，$\mathrm{mgu}_{\mathit{inf}}(E) = \{x := \mathsf{f}(\mathsf{g}(\mathsf{f}(\mathsf{g}(\cdots)))), y := \mathsf{g}(\mathsf{f}(\mathsf{g}(\mathsf{f}(\cdots))))\}$ となる．ここで，例 6 に示した $\eta^\star$ と較べると両者が等しいことがわかる．

実際，次の補題で示すように，再帰式表現 θ の表わす正則代入 $\theta^\star$ は再帰式表現に対応する等式集合の最汎単一化子となっている．

以下では，文脈からあきらかな場合には，再帰式表現 $\theta = [x_1 := t_1, \ldots, x_n := t_n]$ を文脈に応じてそれに対応する等式集合 $\{x_1 \approx t_1, \ldots, x_n \approx t_n\}$ とみなす．

補題 9 (再帰式表現の最汎単一化子) $\theta = [x_1 := t_1, \ldots, x_n := t_n]$ を再帰式表現とする．このとき，正則代入 $\theta^\star$ は再帰式表現 θ に対応する等式集合 $\{x_1 \approx t_1, \ldots, x_n \approx t_n\}$ の最汎単一化子である．

(証明) 例 6 の直前に述べた事実から，一般性を失うことなく，$t_1, \ldots, t_n \notin \{x_1, \ldots, x_n\} (= \mathrm{dom}(\theta))$ としてよい．以下では，まず (1), (2) で $\theta^\star \in \mathrm{Unif}_{\mathit{inf}}(\theta)$ を示し，(3) で任意の代入 $\delta \in \mathrm{Unif}_{\mathit{inf}}(\theta)$ について，$\theta^\star \precsim \delta$ となることを示す．任意の i $(1 \leq i \leq n)$ に対して，$C_i[x_{i_1}, \ldots, x_{i_{k_i}}]_{p_{i_1}, \ldots, p_{i_{k_i}}} = t_i$ とおく．我々の仮定 $t_1, \ldots, t_n \notin \{x_1, \ldots, x_n\}$ から $C_i \neq \square$ である

ことに注意する．

(1) まず，$\mathrm{Pos}(\theta^\star(x_i)) = \mathrm{Pos}(\theta^\star(t_i))$ を示す．$\theta^\star(x_i)(p)$ の定義より，$\mathrm{Pos}(\theta^\star(x_i)) = \{p \in \mathrm{Pos}(C_i) \mid C_i|_p \neq \Box\} \cup \bigcup_{1\leq j\leq k_i}\{p_{i_j}.q \mid q \in \mathrm{Pos}(\theta^\star(x_{i_j}))\}$．一方，$t_i = C_i[x_{i_1},\ldots,x_{i_{k_i}}]$ より，$\mathrm{Pos}(\theta^\star(t_i)) = \mathrm{Pos}(C_i[\theta^\star(x_{i_1}),\ldots,\theta^\star(x_{i_{k_i}})])$．よって，$\mathrm{Pos}(\theta^\star(t_i)) = \{p \in \mathrm{Pos}(C_i) \mid C_i|_p \neq \Box\} \cup \bigcup_{1\leq j\leq k_i}\{p_{i_j}.q \mid q \in \mathrm{Pos}(\theta^\star(x_{i_j}))\} = \mathrm{Pos}(\theta^\star(x_i))$．

(2) 次に，任意の位置 $p \in \mathrm{Pos}(\theta^\star(x_i))$ について，$\theta^\star(x_i)(p) = \theta^\star(t_i)(p)$ であることを p に関する帰納法で示す．$p \in \mathrm{Pos}(C_i)$ かつ $C_i|_p \neq \Box$ ならば $\theta^\star(x_i)(p) = C_i(p) = t_i(p) = \theta^\star(t_i)(p)$．また，$p = p_{i_j}.q$ かつ $q \in \mathrm{Pos}(\theta^\star(x_{i_j}))$ ならば，帰納法の仮定より，$\theta^\star(x_i)(p) = \theta^\star(x_{i_j})(q) = \theta^\star(t_{i_j})(q) = \theta^\star(t_i)(p)$．なお，任意の i について $C_i \neq \Box$ より $p_{i_j} \neq \epsilon$ であることに注意する．以上より，$\theta^\star(x_i) = \theta^\star(t_i)$ が成立する．

(3) 最後に，任意の代入 $\delta \in \mathrm{Unif}_{inf}(\theta)$ について，$\theta^\star \precsim \delta$ となることを示す．今，$\delta \in \mathrm{Unif}_{inf}(\theta)$ より，$\delta(x_i) = \delta(t_i)$ $(\forall i \in \{1,\ldots,n\})$．ここで，一般性を失うことなく $\{x_1,\ldots,x_n\} \cap (\bigcup_{1\leq i\leq n}\mathcal{V}(\delta(x_i))) = \emptyset$ と仮定することができる．この理由を以下に示す．今，$\{x_1,\ldots,x_n\} \cup (\bigcup_{1\leq i\leq n}\mathcal{V}(\delta(x_i)))$ に含まれない変数 $y_1,\ldots,y_n$ をとり，$\delta' = \{x_1 := y_1,\ldots,x_n := y_n\} \circ \delta$ とおくと，$\{x_1,\ldots,x_n\} \cap (\bigcup_{1\leq i\leq n}\mathcal{V}(\delta'(x_i))) = \emptyset$ である．一方，$\delta = \{y_1 := x_1,\ldots,y_n := x_n\} \circ \delta'$ より $\delta' \precsim \delta$．よって，$\theta^\star \precsim \delta'$ を示せば，$\theta^\star \precsim \delta' \precsim \delta$ が得られるからである．

今，$\rho = \delta\restriction_{\mathrm{dom}(\delta)\setminus\{x_1,\ldots,x_n\}}$ とおく．このとき，

$$
\begin{array}{lcl}
\delta(x_i) & = & \delta(t_i)\\
 & = & \delta(C_i[x_{i_1},\ldots,x_{i_{k_i}}])\\
 & = & \delta(C_i)[\delta(x_{i_1}),\ldots,\delta(x_{i_{k_i}})]\\
 & = & \rho(C_i)[\delta(x_{i_1}),\ldots,\delta(x_{i_{k_i}})].\\
\rho(\theta^\star(x_i)) & = & \rho(\theta^\star(t_i))\\
 & = & \rho(C_i[\theta^\star(x_{i_1}),\ldots,\theta^\star(x_{i_{k_i}})])\\
 & = & \rho(C_i)[\rho(\theta^\star(x_{i_1})),\ldots,\rho(\theta^\star(x_{i_{k_i}}))].
\end{array}
$$

これより，$\mathrm{Pos}(\delta(x_i)) = \mathrm{Pos}(\rho(\theta^\star(x_i)))$，および，任意の $p \in \mathrm{Pos}(\delta(x_i))$ について $\delta(x_i)(p) = \rho(\theta^\star(x_i))(p)$ が，p に関する帰納法で容易に示される．よって，$\rho(\theta^\star(x_i)) = \delta(x_i)$ $(\forall i \in \{1,\ldots,n\})$．また，任意の $y \notin \{x_1,\ldots,x_n\}$ について，$\rho(\theta^\star(y)) = \rho(y) = \delta(y)$．以上より，$\delta = \rho \circ \theta^\star$．よって，$\theta^\star \precsim \delta$ が示せた．□

次の補題は補題 11 の証明に用いる技術的な補題である．等式集合 T の単一化問題が，その部分集合 S の最汎単一化子を用いて等価な単一化問題へ変換できることを示している．

補題 10 (単一化問題の同値変形) S, T を $S \subseteq T$ なる等式の有限集合とし，最汎単一化子 $\sigma = \mathrm{mgu}_{inf}(S)$ が存在するとする．このとき，$\theta \in \mathrm{Unif}_{inf}(T) \Leftrightarrow \exists\rho \in \mathrm{Unif}_{inf}(T\sigma).\ \theta = \rho \circ \sigma$．

(証明) ($\Rightarrow$) $\theta \in \mathrm{Unif}_{inf}(T)$ とする．$S \subseteq T$ より，$\theta \in \mathrm{Unif}_{inf}(S)$．よって，$\sigma = \mathrm{mgu}_{inf}(S)$ とすると，ある代入 ρ が存在して，$\theta = \rho \circ \sigma$．このとき，任意の等式 $s \approx t \in T$ に対して，$\theta(s) = \theta(t)$ より，$\rho(\sigma(s)) = \rho(\sigma(t))$．よって，$\rho \in \mathrm{Unif}_{inf}(T\sigma)$．($\Leftarrow$) $\rho \in \mathrm{Unif}_{inf}(T\sigma)$ かつ $\theta = \rho \circ \sigma$ とする．このとき，任意の等式 $s \approx t \in T$ に対して，$\theta(s) = \rho(\sigma(s)) = \rho(\sigma(t)) = \theta(t)$．よって，$\theta \in \mathrm{Unif}_{inf}(T)$．□

効率的な正則項の単一化手続きを実現するアイデアは，正則項の単一化手続きを正則項の再帰式表現にもとづいて実現することである．次の補題は，2 つの正則項に対する単一化問題とそれらの再帰式表現の合併に対する単一化問題の等価性を示す．以下では，再帰式表現 $\theta = [x_1 := s_1,\ldots,x_m := s_m]$ または等式集合 $\theta = \{x_1 \approx s_1,\ldots,x_m \approx s_m\}$ に対して，$\mathcal{V}(\theta) = \{x_1,\ldots,x_m\} \cup \bigcup_{1\leq i\leq m}\mathcal{V}(s_i)$ とおく．このとき，正則代入 $\theta^\star$ の定義より，任意の $1 \leq i \leq m$ について $\mathcal{V}(\theta^\star(x_i)) \subseteq \mathcal{V}(\theta) \setminus \{x_1,\ldots,x_m\}$ に注意する．

補題 11 (再帰式表現にもとづく単一化問題) $\theta = [x_1 := s_1, \ldots, x_m := s_m]$，$\gamma = [y_1 := t_1,\ldots,y_n := t_n]$ を，$x_1,\ldots,x_m \notin \mathcal{V}(\gamma)$，$y_1,\ldots,y_n \notin \mathcal{V}(\theta)$ なる再帰式表現とする．$s = \theta^\star(x_1)$，$t = \gamma^\star(y_1)$ とおく．このとき，$\tau \in \mathrm{Unif}_{inf}(\theta \cup \gamma \cup \{x_1 \approx y_1\}) \Leftrightarrow \exists\rho \in \mathrm{Unif}_{inf}(\{s \approx t\}).\ \tau = \rho \circ (\theta^\star \cup \gamma^\star)$．

(証明) $T = \theta \cup \gamma \cup \{x_1 \approx y_1\}$，$S_1 = \theta$，$S_2 = \gamma$ とおく．補題 9 より，$\theta^\star = \mathrm{mgu}_{inf}(S_1)$，$\gamma^\star = \mathrm{mgu}_{inf}(S_2)$．よって，$S_1 \subseteq T$ ならびに補題 10 より，$\tau \in \mathrm{Unif}_{inf}(T) \Leftrightarrow \exists\delta \in \mathrm{Unif}_{inf}(T\theta^\star).\ \tau = \delta \circ \theta^\star$．ここで，$\mathrm{dom}(\theta^\star) = \{x_1,\ldots,x_m\}$，$y_1 \notin \mathrm{dom}(\theta^\star)$，$x_1,\ldots,x_m \notin \mathcal{V}(\gamma) = \mathcal{V}(S_2)$，$\theta^\star(x_i) =$

$\theta^\star(s_i)$ $(1 \le i \le m)$, $\theta^\star(x_1) = s$ に注意すると，$\mathrm{Unif}_{\mathit{inf}}(T\theta^\star) = \mathrm{Unif}_{\mathit{inf}}(S_1\theta^\star \cup \{\theta^\star(x_1) \approx \theta^\star(y_1)\} \cup S_2\theta^\star) = \mathrm{Unif}_{\mathit{inf}}(S_1\theta^\star \cup \{\theta^\star(x_1) \approx y_1\} \cup S_2\theta^\star) = \mathrm{Unif}_{\mathit{inf}}(S_1\theta^\star \cup \{\theta^\star(x_1) \approx y_1\} \cup S_2) = \mathrm{Unif}_{\mathit{inf}}(\{\theta^\star(x_1) \approx y_1\} \cup S_2) = \mathrm{Unif}_{\mathit{inf}}(\{s \approx y_1\} \cup S_2)$．次に，$T' = \{s \approx y_1\} \cup S_2$ とおくと，$S_2 \subseteq T'$，先と同様に補題 10 を適用すると，$\delta \in \mathrm{Unif}_{\mathit{inf}}(T') \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(T'\gamma^\star).\ \delta = \rho\circ\gamma^\star$ が得られる．さらに，$\mathrm{dom}(\gamma^\star) = \{y_1, \ldots, y_n\}$，$\gamma^\star(y_j) = \gamma^\star(t_j)$ $(1 \le j \le n)$，$\gamma^\star(y_1) = t$，$y_1, \ldots, y_n \notin \mathcal{V}(\theta) = \mathcal{V}(S_1)$ に注意すると，$\mathrm{Unif}_{\mathit{inf}}(T'\gamma^\star) = \mathrm{Unif}_{\mathit{inf}}(\{\gamma^\star(s) \approx \gamma^\star(y_1)\} \cup S_2\gamma^\star) = \mathrm{Unif}_{\mathit{inf}}(\{\gamma^\star(s) \approx \gamma^\star(y_1)\}) = \mathrm{Unif}_{\mathit{inf}}(\{\gamma^\star(s) \approx t\}) = \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\})$．以上より，$\tau \in \mathrm{Unif}_{\mathit{inf}}(T) \Leftrightarrow \exists\delta \in \mathrm{Unif}_{\mathit{inf}}(T\theta^\star).\ \tau = \delta\circ\theta^\star \Leftrightarrow \exists\delta \in \mathrm{Unif}_{\mathit{inf}}(T').\ \tau = \delta\circ\theta^\star \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(T'\gamma^\star).\ \delta = \rho\circ\gamma^\star \wedge \tau = \delta\circ\gamma^\star \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(T'\gamma^\star).\ \tau = \rho\circ\gamma^\star\circ\theta^\star \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\}).\ \tau = \rho\circ\gamma^\star\circ\theta^\star$．ここで，任意の $1 \le j \le n$ について，正則代入 $\gamma^\star$ の定義から $\mathcal{V}(\gamma^\star(y_j)) \subseteq \mathcal{V}(\gamma)$ が成り立つから，$\mathrm{dom}(\theta^\star) \cap \mathcal{V}(\gamma) = \emptyset$ より，$\mathrm{dom}(\theta^\star) \cap \mathcal{V}(\gamma^\star(y_j)) = \emptyset$．同様に，$\mathrm{dom}(\gamma^\star) \cap \mathcal{V}(\theta^\star(x_i)) = \emptyset$ $(1 \le i \le m)$．よって，$\tau \in \mathrm{Unif}_{\mathit{inf}}(T) \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\}).\ \tau = \rho\circ\gamma^\star\circ\theta^\star = \rho\circ(\gamma^\star\cup\theta^\star)$．□

再帰式表現を構成する項は有限項であるから，補題 11 より以下の定理が成立する．

定理 12 (再帰式表現にもとづく単一化手続き) $\theta = [x_1 := s_1, \ldots, x_m := s_m\ ]$, $\gamma = [y_1 := t_1, \ldots, y_n := t_n]$ を，$x_1, \ldots, x_m \notin \mathcal{V}(\gamma)$，$y_1, \ldots, y_n \notin \mathcal{V}(\theta)$ なる再帰式表現とし，$s = \theta^\star(x_1)$，$t = \gamma^\star(y_1)$ とおく．このとき，s と t の単一化問題は $\texttt{unif-proc}_J(\theta\cup\gamma\cup\{x_1 \approx y_1\})$ によって解ける．

(証明) 正則代入 $\theta^\star$ の定義より，$\mathcal{V}(\theta^\star(x_i)) \subseteq \mathcal{V}(\theta)$ $(1 \le i \le m)$．よって，仮定 $y_1, \ldots, y_n \notin \mathcal{V}(\theta)$，$s = \theta^\star(x_1)$ より，$\mathrm{dom}(\gamma^\star) \cap \mathcal{V}(s) = \emptyset$ が成り立つ．補題 11 より，$\tau \in \mathrm{Unif}_{\mathit{inf}}(\theta\cup\gamma\cup\{x_1 \approx y_1\}) \Leftrightarrow \exists\rho \in \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\}).\ \tau = \rho\circ(\theta^\star\cup\gamma^\star)$．従って，等式集合 $\theta\cup\gamma\cup\{x_1 \approx y_1\}$ が単一化可能であり，かつ，その時に限り，等式集合 $\{s \approx t\}$ が単一化可能となる．ここで，再帰式表現を構成する項は有限項であるから，$\theta\cup\gamma\cup\{x_1 \approx y_1\}$ に対して単一化手続き $\texttt{unif-proc}_J$ を適用することができる．

$\theta\cup\gamma\cup\{x_1 \approx y_1\}$ が単一化可能な場合，単一化手続き $\texttt{unif-proc}_J$ により，$\tau = \mathrm{mgu}_{\mathit{inf}}(\theta\cup\gamma\cup\{x_1 \approx y_1\})$ が得られる．このとき，$\tau \in \mathrm{Unif}_{\mathit{inf}}(\theta\cup\gamma\cup\{x_1 \approx y_1\})$ であるから，補題 11 より，ある $\rho \in \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\})$ が存在して $\tau = \rho\circ(\theta^\star\cup\gamma^\star)$．ここで，定義および再帰式表現の変数条件より，$x_1, \ldots, x_m \notin \bigcup_{1\le i\le m}\mathcal{V}(\theta^\star(x_i)) \cup \bigcup_{1\le j\le n}\mathcal{V}(\gamma^\star(y_j))$，かつ $y_1, \ldots, y_n \notin \bigcup_{1\le i\le m}\mathcal{V}(\theta^\star(x_i)) \cup \bigcup_{1\le j\le n}\mathcal{V}(\gamma^\star(y_j))$ が成立するから，$\rho = \tau\restriction_{\mathrm{dom}(\tau)\setminus\{x_1,\ldots,x_m,y_1,\ldots,y_n\}}$ が成立する．ρ が等式集合 $\{s \approx t\}$ の最汎単一化子であることを示す．それには，$\rho' \in \mathrm{Unif}_{\mathit{inf}}(\{s \approx t\})$ をとるとき，$\rho \precsim \rho'$ を示せばよい．$\tau' = \rho'\circ(\theta^\star\cup\gamma^\star)$ ととると，補題 11 より，$\tau' \in \mathrm{Unif}_{\mathit{inf}}(\theta\cup\gamma\cup\{x_1 \approx y_1\})$．従って，$\tau$ の最汎性より，$\tau \precsim \tau'$．よって，$\rho = \tau\restriction_{\mathrm{dom}(\tau)\setminus\{x_1,\ldots,x_m,y_1,\ldots,y_n\}} \precsim \tau'\restriction_{\mathrm{dom}(\tau)\setminus\{x_1,\ldots,x_m,y_1,\ldots,y_n\}} = \rho'$ となる．□

正則項の再帰式表現 $[x_1 := s_1, \ldots, x_m := s_m]$ において，変数 $x_1, \ldots, x_m$ を名前替えしても表わす正則項は変わらない．従って，正則項 s, t の再帰式表現を変更して，定理 12 の条件を満たすような再帰式表現を得ることは容易である．よって，再帰式表現された正則項の単一化問題は，Jaffer の単一化手続きを用いて効率的に解くことができる．

4 ω-強頭部正規化可能性の反証手続き

本節では，ω-強頭部正規化可能性について説明し，ω-強頭部正規化可能性の反証についての従来結果を説明する．そして，ω-強頭部正規化可能性の反証手続きを与え，その正しさ (健全性) を示す．

定義 13 (ω-強頭部正規化可能性 [33]) $\mathcal{R}$ を無限項書き換えシステムとする．任意の無限書き換え列 $t_0 \to_{p_0,\mathcal{R}} t_1 \to_{p_1,\mathcal{R}} t_2 \to_{p_2,\mathcal{R}} \cdots$ について，あるインデックス n_0 が存在して任意の n $(n_0 \le n < \omega)$ について $p_n \ne \epsilon$ となるとき，$\mathcal{R}$ は **ω-強頭部正規化可能性**をもつという．無限項書き換えシステム $\mathcal{R}$ が ω-強頭部正規化可能性をもつことを $\mathrm{SHN}^\omega(\mathcal{R})$ と記す[†2]．

†2 文献 [33] では $\mathrm{SN}^\omega(\mathcal{R})$ とのみ記されており，ω-強頭部正規化可能性という用語は使われていない．

ω-強頭部正規化可能性を一般化した性質として強頭部正規化可能性がある[23] [31]．強頭部正規化可能性については本論文で取り扱わないが，強頭部正規化可能性の定義は，ω-強頭部正規化可能性の定義を無限書き換え列だけではなく超限書き換え列を対象とするように一般化することによって得られる．任意の無限項書き換えシステムについて，強頭部正規化可能性を持てば ω-強頭部正規化可能性をもつこと，および，無限項書き換えシステムが左線形の場合には，強頭部正規化可能性をもつことと ω-強頭部正規化可能性をもつことが同値になることが知られている[23] [31]．

Zantema [33] や Endrullis ら [11] によって，無限項書き換えシステムが ω-強頭部正規化可能性をもつことを示す手法が提案されている．一方，無限項書き換えシステムが ω-強頭部正規化可能性をもたないための十分条件として以下の条件が知られている[33]．

命題 14 (縮退システムの ω-強頭部正規化可能性) 無限項書き換えシステム $\mathcal{R}$ が縮退性をもつとき，ω-強頭部正規化可能性をもたない．

(証明) $\mathcal{R}$ の縮退性より，$C[x,\ldots,x] \to x$ (ただし $C \neq \Box$) なる書き換え規則が $\mathcal{R}$ に存在する．このとき，正則項 t を $t = \mu x.C[x,\ldots,x]$ とおく．このとき，$t = C[t,\ldots,t]$ であることに注意すると，

$$t = C[t,\ldots,t] \to_{\epsilon,\mathcal{R}} t = C[t,\ldots,t] \to_{\epsilon,\mathcal{R}} \cdots$$

となるので，$\mathcal{R}$ は ω-強頭部正規化可能性をもたない．

□

縮退性をもたない無限項書き換えシステム $\mathcal{R}$ についても，根書き換えが無限回出現するような無限書き換え列を構成することによって，$\mathcal{R}$ が ω-強頭部正規化可能性をもたないことを示せる場合がある．

岩見[19] は，反例を構成することによって，組合せ子の書き換え規則に対して ω-強頭部正規化可能性が成立しないことを報告している．以下，本論文の組合せ子の例では，適用演算子 $\cdot$ に中置記法を用いる．また，慣例にしたがい，$\cdot$ は左結合であるとして，不要な括弧は省略する．

例 15 (ω-強頭部正規化可能性の反証[19]) 無限項書き換えシステム $\mathcal{R}$ を $\mathcal{R} = \{\mathsf{L}\cdot x\cdot y \to x\cdot(y\cdot y)\}$ とおく．このとき，正則項 $t = \mu x.\mathsf{L}\cdot x$ をとる．このとき，$t = \mathsf{L}\cdot t$ であることに注意すると，

$$\begin{array}{rcl} \mathsf{L}\cdot t\cdot y & \to_{\epsilon,\mathcal{R}} & t\cdot(y\cdot y) \\ & = & \mathsf{L}\cdot t\cdot(y\cdot y) \\ & \to_{\epsilon,\mathcal{R}} & t\cdot((y\cdot y)\cdot(y\cdot y)) \\ & = & \mathsf{L}\cdot t\cdot((y\cdot y)\cdot(y\cdot y)) \\ & \to_{\epsilon,\mathcal{R}} & \cdots \end{array}$$

となるので，$\mathcal{R}$ は ω-強頭部正規化可能性をもたない．

この例では，適切な正則項 t を与えることによって，無限回の根書き換えを含むような無限書き換え列を構成している．もし，このような正則項 t を自動的に構成もしくは発見できれば，ω-強頭部正規化可能性の自動反証が可能になるはずである．このようなアイデアにもとづいて，本節の残りでは，正則項を利用した ω-強頭部正規化可能性の自動反証手続きを与える．

例 15 の無限書き換え列がどのようにすれば構成できるか考えることから考える．まず，書き換え規則の左辺 $\mathsf{L}\cdot x\cdot y$ の代入例からの，根位置での書き換え列を構成する．これには，すでに得られた書き換え列の最終項に対して書き換え規則 $\mathsf{L}\cdot x\cdot y \to x\cdot(y\cdot y)$ による書き換えが根位置で適用可能となるような代入を求める．このような代入があれば，これを書き換え列全体に適用することによって，最終項に対する根書き換えが適用可能となるため，根位置での書き換え列を 1 ステップ延長することができる．

$$\begin{array}{rl} & \mathsf{L}\cdot x_0\cdot y_0 \\ \to_{\epsilon,\mathcal{R}} & x_0\cdot(y_0\cdot y_0) \\ (\{x_0 := \mathsf{L}\cdot x_1\}\text{ を適用}) & \mathsf{L}\cdot x_1\cdot(y_0\cdot y_0) \\ \to_{\epsilon,\mathcal{R}} & x_1\cdot((y_0\cdot y_0)\cdot(y_0\cdot y_0)) \\ (\{x_1 := \mathsf{L}\cdot x_2\}\text{ を適用}) & \mathsf{L}\cdot x_2\cdot((y_0\cdot y_0)\cdot(y_0\cdot y_0)) \\ \to_{\epsilon,\mathcal{R}} & \cdots \end{array}$$

このような根位置での書き換えステップに必要な代入は，根ナローイングを用いることによって求めることができる．

$$\begin{array}{rcl} \mathsf{L}\cdot x_0\cdot y_0 & \rightsquigarrow_{\epsilon,\theta_0} & \mathsf{L}\cdot x_1\cdot(y_0\cdot y_0) \\ & & (\text{ここで }\theta_0 = \{x_0 := \mathsf{L}\cdot x_1\}) \\ & \rightsquigarrow_{\epsilon,\theta_1} & \mathsf{L}\cdot x_2\cdot((y_0\cdot y_0)\cdot(y_0\cdot y_0)) \\ & & (\text{ここで }\theta_1 = \{x_1 := \mathsf{L}\cdot x_2\}) \\ & \rightsquigarrow_{\epsilon,\theta_2} & \cdots \end{array}$$

このように次々に代入を求めていくと，より長い根位置での書き換え列を得ることができる．しかし，こ

```
fun disprove-omega-shn (𝓡) =
let fun ckeck (l, t, σ) =
            if ∃θ. θ = mgu_inf(REN(lσ), t), ∃δ ▶ θ. match_inf(lσδ⋆, tδ⋆)
            then SOME lσδ⋆ else NONE
      fun narrow (l, t, σ) = [(l, t′, ρ ∘ σ) | t ⇝_ρ t′]
      fun step [] = FAIL
         | step ((l, t, σ)::xs) = case check (l, t, σ) of
                     SOME t ⇒ SUCCESS t | NONE ⇒ step (xs @ narrow (l, t, σ))
in step [(l, r, ∅) | l → r ∈ 𝓡] end
```

プログラム 1　ω-強頭部正規化可能性の反証手続き

れを繰り返しても無限書き換え列が求まるわけではない．

例 15 では，無限書き換え列を構成するには $t = \mathsf{L} \cdot t$ なる正則項を変数 x へ代入する必要があった．ここで，ナローイングで求まる代入 $\theta_0, \theta_1, \ldots$ が $\{x_i := \mathsf{L} \cdot x_{i+1}\}$ の形をしていることと，$t = \mathsf{L} \cdot t$ なる正則項が再帰式表現 $\{x := \mathsf{L} \cdot x\}$ の解であることに着目すると，ナローイングによって得られる代入をいくつかの変数を同一視することによって変更すれば，適当な再帰式表現が構成できるのではないかと想像される．

代入 $\{x := \mathsf{L} \cdot x'\}$ から再帰式表現に対応する代入 $\{x := \mathsf{L} \cdot x\}$ を構成するために，再帰式表現を表わす代入 δ が代入 θ から，いくつかの変数を同一視することによって得られることを表わす関係 $\delta \blacktriangleright \theta$ を定義する．

定義 16 (有限代入上の関係 ▶) $\theta = \{x_1 := u_1, \ldots, x_n := u_n\}$ を有限代入，$X = \bigcup_{1 \le i \le n} \mathcal{V}(u_i)$，$Y = X \cup \{x_1, \ldots, x_n\}$ とおく．ある代入 $\xi : X \to Y$ について，$\delta = \xi \circ \theta$ となり，δ が再帰式表現であるとき，$\delta \blacktriangleright \theta$ と記す．

ここで，代入 ξ は異なる変数を同一にするために用いている．また，有限代入 θ が固定されたときに，$\delta \blacktriangleright \theta$ となる代入 δ の個数は有限であることに注意する．

例 17 (関係 ▶) $\{x_0 := \mathsf{L} \cdot x_0, x_1 := x_0\} \blacktriangleright \{x_0 := \mathsf{L} \cdot x_1\}$ が成立する．$X = \{x_1\}, Y = \{x_0, x_1\}$ より $\xi = \{x_1 := x_0\}$ をとればよい．

例 15 にもどると，このようにして得られた代入 $\delta = \{x := \mathsf{L} \cdot x\} \blacktriangleright \theta_i$ から $\{x := t\}$ を得る．これには，$\delta^\star = \{x := t\}$ の関係を用いればよい．

得られた $l\delta^\star$ から根位置における無限書き換え列の存在は，$\mathrm{match}_{inf}(l\delta^\star, r\delta^\star)$ によって判断する．実際，$\mathrm{match}_{inf}(l\delta^\star, r\delta^\star)$ が成立するとき，ある正則代入 η が存在して $r\delta^\star = l\delta^\star\eta$ となる．よって，$l\delta^\star \to_{\mathcal{R}} r\delta^\star = l\delta^\star\eta$ となり，ここから根位置における無限書き換え列 $l\delta^\star \to_{\epsilon,\mathcal{R}} r\delta^\star = l\delta^\star\eta \to_{\epsilon,\mathcal{R}} r\delta^\star\eta = l\delta^\star\eta\eta \to_{\epsilon,\mathcal{R}} \cdots$ が構成できる．

最後に，ω-強頭部正規化可能性の反証では，根位置以外の書き換えステップを併用することもできるため，以上の説明で用いた書き換え規則 $l \to r \in \mathcal{R}$ の左辺 l と右辺 r の替わりに，$l\sigma \to_{\mathcal{R}} r\sigma \to^*_{\mathcal{R}} t$ なる項 $l\sigma$ と t を用いてもよいことに注意する．

このようにして得られたのがプログラム 1 の反証手続きである．

手続き 1 (ω-強頭部正規化可能性反証手続き) プログラム 1 に SML プログラミング言語[27] 風のシンタックスをもつ関数型プログラムとして ω-強頭部正規化可能性の反証手続き `disprove-omega-shn` を与える．手続きへの入力は，右有限な無限項書き換えシステム $\mathcal{R}$ である．手続きが成功した場合にはある正則項 t について `SUCCESS` t を，失敗した場合は `FAIL` を返す．また手続きが停止しない (発散する) 場合もある．手続き中の $\mathrm{REN}(t)$ は，項 t に出現する変数を新しい変数に名前換えした項を表す．また，$[x \mid P(x)]$ はリストの内包表記，つまり $P(x)$ を満たす要素 x のリス

トを表わす[†3].

手続きにおける実質的な反証条件の判定は手続き2行目の check 関数で行っている．手続き3行目の $\theta = \mathrm{mgu}_{\mathit{inf}}(\mathrm{REN}(l\sigma), t)$ は，ナローイングによって得られる代入を求めている．ここではナローイングを用いるため，変数の名前換えを行っている．次に，変数を同一視した $\delta \blacktriangleright \theta$ なる代入 δ を求め，根位置における無限書き換え列の存在を保証するために照合 $\mathrm{match}_{\mathit{inf}}(l\sigma\delta^\star, t\delta^\star)$ を行う．なお，上で説明したように，無限書き換え列の存在を保証するための照合は，ナローイングの場合と違い，変数の名前換えを行わないで行う必要があることに注意する．

6行目の step 関数は幅優先探索で check 関数による判定を試みる候補を構成し，check 関数を候補に適用する．判定候補となる3つ組 (l, t, σ) は，$l \to r \in \mathcal{R}$ かつ $r \rightsquigarrow^*_\sigma t$ を満たす．check 関数にて反証条件を調べ，判定に失敗した場合にはナローイング列 $r \rightsquigarrow^*_\sigma t$ を延長し，候補リストに追加する．

以下で，プログラム1の反証手続きの正しさ (健全性)，つまり，手続きに成功する場合には必ず ω-強頭部正規化可能性が成立しないことを示す．

定理 18 (SHN$^\omega$ 反証法の健全性) 右有限な無限項書き換えシステム $\mathcal{R}$ に対して，プログラム1の ω-強頭部正規化可能性の反証手続きが成功するとき，$\mathcal{R}$ は ω-強頭部正規化可能性をもたない．

(証明) step xs がよびだされるとき，任意の3つ組 $(l, t, \sigma) \in xs$ について，$l \to r \in \mathcal{R}$ かつ $r \rightsquigarrow^*_\sigma t$ が成立することは容易に確認できる．手続きが成功するときは，ある3つ組 $(l, t, \sigma) \in xs$ について，check (l, t, σ) が成功したときである．よって，SUCCESS $l\sigma\delta^\star$ が導出されるとき，ある代入 σ, θ, δ について，$\theta = \mathrm{mgu}_{\mathit{inf}}(\mathrm{REN}(l\sigma), t)$，$\delta \blacktriangleright \theta$ かつ $\mathrm{match}_{\mathit{inf}}(l\sigma\delta^\star, t\delta^\star)$ が成立する．$\eta = \mathrm{match}_{\mathit{inf}}(l\sigma\delta^\star, t\delta^\star)$ とおくと，$l\sigma\delta^\star\eta = t\delta^\star$．また，$r \rightsquigarrow^*_\sigma t$ より $r\sigma \to^*_{\mathcal{R}} t$．以上より，$l\sigma\delta^\star \to_{\epsilon,\mathcal{R}} r\sigma\delta^\star \to^*_{\mathcal{R}} t\delta^\star = l\sigma\delta^\star\eta \to_{\epsilon,\mathcal{R}} r\sigma\delta^\star\eta \to^*_{\mathcal{R}} t\delta^\star\eta = l\sigma\delta^\star\eta\eta \to_{\epsilon,\mathcal{R}} \cdots$ が成立する．よって，$\mathcal{R}$ は ω-強頭部正規化可能性をもたない． □

†3 リスト内包表記は SML では採用されていないが，例えばプログラミング言語 Haskell などで採用されている [32].

以下では，例15の書き換えシステムについて，上記の手続きにもとづいて ω-強頭部正規化可能性の反証に成功することを示す．

例 19 (SHN$^\omega$ 反証手続きの適用例 (1)) 無限項書き換えシステム $\mathcal{R}$ を $\mathcal{R} = \{\mathsf{L} \cdot x \cdot y \to x \cdot (y \cdot y)\}$ とおく．9行目で step $[(\mathsf{L} \cdot x \cdot y, x \cdot (y \cdot y), \emptyset)]$ がよびだされ，2行目の check $(\mathsf{L} \cdot x \cdot y, x \cdot (y \cdot y), \emptyset)$ の評価に入る．このとき，$\theta = \mathrm{mgu}_{\mathit{inf}}(\mathsf{L} \cdot x' \cdot y', x \cdot (y \cdot y)) = \{x := \mathsf{L} \cdot x', y' := y \cdot y\}$ が得られる．次に，$\delta = \{x := \mathsf{L} \cdot x, y' := y \cdot y, x' := x\} \blacktriangleright \theta$ をとると，$\delta^\star = \{x := \mu x.\mathsf{L} \cdot x, y' := y \cdot y, x' := \mu x.\mathsf{L} \cdot x\}$ が求まる．今，$u = \mu x.\mathsf{L} \cdot x$ とおいて，$u = \mathsf{L} \cdot u$ に注意すると，

$$
\begin{aligned}
& \mathrm{match}_{\mathit{inf}}((\mathsf{L} \cdot x \cdot y)\emptyset\delta^\star, (x \cdot (y \cdot y))\delta^\star) \\
= \;& \mathrm{match}_{\mathit{inf}}((\mathsf{L} \cdot x \cdot y)\delta^\star, (x \cdot (y \cdot y))\delta^\star) \\
= \;& \mathrm{match}_{\mathit{inf}}(\mathsf{L} \cdot u \cdot y, u \cdot (y \cdot y)) \\
= \;& \mathrm{match}_{\mathit{inf}}(u \cdot y, u \cdot (y \cdot y)) \\
= \;& \{y := y \cdot y\}.
\end{aligned}
$$

よって，check $(\mathsf{L}\cdot x\cdot y, x\cdot(y\cdot y), \emptyset)$ は SOME $(\mu x.\mathsf{L}\cdot x)\cdot y$ と評価され，SUCCESS $(\mu x.\mathsf{L} \cdot x) \cdot y$ が返され，手続きが成功する．

組合せ子でない項書き換えシステムに対して反証手続きを適用した例を1つ示す．

例 20 (SHN$^\omega$ 反証手続きの適用例 (2)) 無限項書き換えシステム $\mathcal{R}$ を $\mathcal{R} = \{\mathsf{f}(x) \to \mathsf{g}(x), \mathsf{g}(\mathsf{h}(x)) \to \mathsf{f}(x)\}$ とおく．最初，9行目の step $[(\mathsf{f}(x), \mathsf{g}(x), \emptyset), (\mathsf{g}(\mathsf{h}(x)), \mathsf{f}(x), \emptyset)]$ がよびだされ，2行目の check $(\mathsf{f}(x), \mathsf{g}(x), \emptyset)$ の評価に入るが，手続き3行目で $\mathsf{f}(x')$ と $\mathsf{g}(x)$ との単一化に失敗するため，NONE が返る．$\mathsf{g}(x)$ から唯一可能なナローイングステップは，$\mathsf{g}(x) \rightsquigarrow_{\epsilon,\{x:=\mathsf{h}(x_1)\}} \mathsf{f}(x_1)$．よって，次に手続き8行目で，step $[(\mathsf{g}(\mathsf{h}(x)), \mathsf{f}(x), \emptyset), (\mathsf{f}(x), \mathsf{f}(x_1), \{x := \mathsf{h}(x_1)\})]$ がよびだされる．先と同様に，check $(\mathsf{g}(\mathsf{h}(x)), \mathsf{f}(x), \emptyset)$ も NONE と評価され，手続き8行目で step $[(\mathsf{f}(x), \mathsf{f}(x_1), \{x := \mathsf{h}(x_1)\}), (\mathsf{g}(\mathsf{h}(x)), \mathsf{g}(x_2), \{x := x_2\})]$ がよびだされる．check $(\mathsf{f}(x), \mathsf{f}(x_1), \{x := \mathsf{h}(x_1)\})$ は，$\theta = \mathrm{mgu}_{\mathit{inf}}(\mathsf{f}(\mathsf{h}(x'_1)), \mathsf{f}(x_1)) = \{x_1 := \mathsf{h}(x'_1)\}$ となる．ここで，$\delta = \{x_1 := \mathsf{h}(x_1), x'_1 := x_1\} \blacktriangleright \theta$ をとると，$\delta^\star = \{x_1 := \mathsf{h}^\omega, x'_1 := \mathsf{h}^\omega\}$．よっ

て，$\mathrm{match}_{inf}(\mathsf{f}(\mathsf{h}(x_1))\delta^\star, \mathsf{f}(x_1)\delta^\star) = \mathrm{match}_{inf}(\mathsf{f}(\mathsf{h}^\omega), \mathsf{f}(\mathsf{h}^\omega)) = \emptyset$ より，SUCCESS $\mathsf{f}(\mathsf{h}^\omega)$ が返り，手続きが成功する．実際，$\mathsf{f}(\mathsf{h}^\omega) \to_{\epsilon,\mathcal{R}} \mathsf{g}(\mathsf{h}^\omega) = \mathsf{g}(\mathsf{h}(\mathsf{h}^\omega)) \to_{\epsilon,\mathcal{R}} \mathsf{f}(\mathsf{h}^\omega) \to_{\epsilon,\mathcal{R}} \cdots$ であるから，得られた項は $\mathcal{R}$ の ω-強頭部正規化可能性の反例になっている．

ω-強頭部正規化可能性をもたない無限項書き換えシステムは非強頭部正規化可能であるから，提案手続きは強頭部正規化可能性の反証手続きにもなっていることに注意する．

5 一般生成性の反証手続き

本節では，ストリーム項書き換えシステムの生成性について説明し，生成性問題を拡張して，一般生成性問題を与える．そして，一般生成性問題の反証手続きを与え，その正しさ (健全性) を与える．

本節では，多ソート言語を考える．ソート集合を $\mathcal{S}$ と記す．つまり，関数記号 $f \in \mathcal{F}_n$ のそれぞれは $f : S_1 \times \cdots \times S_n \to S_0$ $(S_0, \ldots, S_n \in \mathcal{S})$ に型付けされており，型整合な項のみを考えるものとする．なお，通常の項書き換えシステムにおいても多ソート言語への拡張を考えることは普通であり (多ソート項書き換えシステム) [1] [14]，無限項書き換えシステムも多ソート言語上に自然に拡張される [10]．

定義 21 (ストリーム項書き換えシステム [10]) 以下を満たすソート集合 $\mathcal{S}$，関数記号集合 $\mathcal{F}$ 上の多ソート無限項書き換えシステム $\mathcal{R}$ を**ストリーム項書き換えシステム**(stream term rewriting system) とよぶ．

1. $\mathcal{S} = \{D, S\}$．D を**データ型**，S を**ストリーム型**とよぶ．
2. (a) $\mathcal{F} = \mathcal{F}_D \cup \mathcal{F}_S$．(b) 任意の $f \in \mathcal{F}_D$ について $f : D \times \cdots \times D \to D$．$\mathcal{F}_D$ の要素を**データ記号**とよぶ．(c) 任意の $f \in \mathcal{F}_S$ について $f : D \times \cdots \times D \times S \times \cdots \times S \to S$．$\mathcal{F}_S$ の要素を**ストリーム記号**とよぶ．
3. $\mathsf{cons} : D \times S \to S$ が唯一つの構成子ストリーム記号である．つまり，$\mathcal{C} \cap \mathcal{F}_S = \{\mathsf{cons} : D \times S \to S\}$．また，以下では，cons の中置記法として :: を用いる．
4. $\mathcal{R}$ は右有限な直交構成子システムである．

定義の条件 3 より，ストリーム項書き換えシステムのストリーム型基底構成子項は $u_0 :: u_1 :: \cdots$ の形 $(\forall i.\ u_i \in \mathcal{T}_{inf}(\mathcal{C}, \emptyset))$ をした無限項となることに注意する．

定義 22 (生成性 [10]) $\mathcal{R}$ をストリーム項書き換えシステム，t をストリーム型項とする．t が $\mathcal{R}$ において**生成的**(productive) である (t が**生成性をもつ**) とは，t が ω-正規形 s をもち，かつ，s が構成子項となるときをいう．t が $\mathcal{R}$ において生成的かどうかを問う問題を**生成性問題**とよぶ．

ここで，$\mathcal{R}$ の直交性より，任意の項の ω-正規形は存在すれば唯一となる (cf. [10]) ことに注意する．

例 23 (ストリーム項書き換えシステムと生成性 (1)) 関数記号集合を $\mathcal{F} = \{\mathsf{zero}, \mathsf{m} : S \to S,\ \mathsf{A} : S,\ 0, 1 : D,\ \mathsf{cons} : D \times S \to S\}$ として，以下の書き換え規則をもつ無限項書き換えシステム $\mathcal{R}$ を考える．

$$\left\{\begin{array}{l} \mathsf{zero}(\mathsf{cons}(0, xs)) \to \mathsf{cons}(0, \mathsf{zero}(xs)) \\ \mathsf{zero}(\mathsf{cons}(1, xs)) \to \mathsf{zero}(xs) \\ \mathsf{m}(\mathsf{cons}(0, xs)) \to \mathsf{cons}(0, \mathsf{m}(xs)) \\ \mathsf{m}(\mathsf{cons}(1, xs)) \to \mathsf{cons}(1, \mathsf{cons}(1, \mathsf{m}(xs))) \\ \mathsf{A} \to \mathsf{cons}(0, \mathsf{cons}(1, \mathsf{m}(\mathsf{A}))) \end{array}\right\}$$

このとき，$\mathcal{R}$ は定義 21 の条件をすべて満たすので，ストリーム項書き換えシステムである．$\mathsf{zero}(xs)$ はストリーム xs から要素 0 のみを取り出す．$\mathsf{m}(xs)$ は xs 中の要素 1 の出現を倍に増やす．A はストリーム $0 :: 1 :: 0 :: 1^2 :: 0 :: 1^4 :: 0 :: 1^8 :: \cdots$ を生成する．ここで，便宜的に，x^n は要素 x の n 連続を表わすものとする．このストリームでは，連続する 0 の出現間の 1 の出現回数が指数的に増加していくが，0 の出現する回数は無限回である．従って，項 $\mathsf{zero}(\mathsf{A})$ を考えると，その ω-正規形は $0 :: 0 :: \cdots$ であり，$\mathsf{zero}(\mathsf{A})$ は生成的である．

例 24 (ストリーム項書き換えシステムと生成性 (2)) 関数記号集合を $\mathcal{F} = \{\mathsf{m}_0, \mathsf{m}_1 : S \to S,\ \mathsf{B} : S,\ 0 : D,\ \mathsf{cons} : D \times S \to S\}$ として，以下の書き換え規則をもつ無限項書き換えシステム $\mathcal{R}$ を考える．

$$\left\{\begin{array}{l} \mathsf{m}_0(\mathsf{cons}(0, xs)) \to \mathsf{m}_1(xs) \\ \mathsf{m}_1(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{m}_0(xs)) \\ \mathsf{B} \to \mathsf{cons}(0, \mathsf{cons}(0, \cdots \mathsf{cons}(0, \mathsf{m}_0(\mathsf{B})) \cdots)) \end{array}\right\}$$

ただし，最後の規則で 0 は 2^n 個続いているものとする．このとき，$\mathcal{R}$ は定義 21 の条件をすべて満たすの

で，ストリーム項書き換えシステムである．$\mathsf{m}_0(xs)$ はストリーム xs から要素 0 の後ろの要素のみを取り出すので，要素 0 の 2^n 回連続を 2^{n-1} 回連続へ減らす．従って，

$$
\begin{array}{lll}
\mathsf{B} & \to^*_{\mathcal{R}} & 0^{2^n} :: 0^{2^{n-1}} :: \cdots :: 0 :: {\mathsf{m}_0}^{n+1}(B) \\
& \to^*_{\mathcal{R}} & 0^{2^n} :: 0^{2^{n-1}} :: \cdots :: 0 :: \mathsf{m}_1({\mathsf{m}_0}^{n+1}(B)) \\
& \to^*_{\mathcal{R}} & 0^{2^n} :: 0^{2^{n-1}} :: \cdots :: 0 :: {\mathsf{m}_1}^2({\mathsf{m}_0}^{n+1}(B)) \\
& \to^*_{\mathcal{R}} & 0^{2^n} :: 0^{2^{n-1}} :: \cdots :: 0 :: {\mathsf{m}_1}^3({\mathsf{m}_0}^{n+1}(B)) \\
& \to^*_{\mathcal{R}} & \cdots
\end{array}
$$

となる．従って，B の ω-正規形は構成子項ではない．よって，B は生成的でない．

このように，zero(A) も B も，最初のうちは両者とも 0 を出力するが，後者はいずれ出力がストップする[†4]．

本節では，生成性およびそれを一般化した性質である一般生成性の反証法を示す．そこで，まず，どのような場合に生成性が失われるかを具体例を参考に考えてみる．

例 25 (非生成性の要件 (1)) 例 24 において，B は生成的でない．これは，B から生成されるストリームにおいて，0 を $\sum_{i=0}^{n} 2^i$ 個目を出力して以降，出力する要素が得られない (根記号が cons となる項が書き換えによって得られない) ためである．$\sum_{i=0}^{n} 2^i + 1$ 個目の要素の計算は，${\mathsf{m}_0}^{n+1}(\mathsf{B}) \to^*_{\mathcal{R}} \mathsf{m}_1({\mathsf{m}_0}^{n+1}(\mathsf{B})) \to^*_{\mathcal{R}} {\mathsf{m}_1}^2({\mathsf{m}_0}^{n+1}(\mathsf{B})) \to^*_{\mathcal{R}} {\mathsf{m}_1}^3({\mathsf{m}_0}^{n+1}(\mathsf{B})) \to^*_{\mathcal{R}} \cdots$ となっている．このように cons を根記号に持たない文脈 C に対して，$t \to^*_{\mathcal{R}} C[t] \to^*_{\mathcal{R}} C[C[t]] \to^*_{\mathcal{R}} C[C[C[t]]] \to^*_{\mathcal{R}} \cdots$ なる書き換えが続くとき，生成性が失われることがわかる．

例 25 よりも精緻な検証が必要となる場合を次の例で示す．

例 26 (非生成性の要件 (2)) 関数記号集合を $\mathcal{F} = \{\mathsf{g} : S \times S \to S,\ \mathsf{A}, \mathsf{B} : S,\ 0 : D,\ \mathsf{cons} : D \times S \to S\}$ として，以下の書き換え規則をもつ無限項書き換えシステム $\mathcal{R}$ を考える．

$$
\left\{
\begin{array}{l}
\mathsf{g}(\mathsf{cons}(x, xs), ys) \to \mathsf{cons}(x, \mathsf{g}(ys, xs)) \\
\mathsf{A} \to \mathsf{g}(\mathsf{B}, \mathsf{A}) \\
\mathsf{B} \to \mathsf{cons}(0, \mathsf{A})
\end{array}
\right\}
$$

このとき，$\mathcal{R}$ は定義 21 の条件をすべて満たすので，ストリーム項書き換えシステムである．$\mathcal{R}$ による以下の書き換え列を考える．

$$
\begin{array}{lll}
\mathsf{A} & \to_{\mathcal{R}} & \mathsf{g}(\mathsf{B}, \mathsf{A}) \\
& \to_{\mathcal{R}} & \mathsf{g}(\mathsf{B}, \mathsf{g}(\mathsf{B}, \mathsf{A})) \\
& \to_{\mathcal{R}} & \cdots
\end{array}
$$

文脈 $C[\,] = \mathsf{g}(\mathsf{B}, \Box)$，項 $t = \mathsf{A}$ をとれば，$C(\epsilon) \neq \mathsf{cons}$．また，$\mathsf{A} = t \to^*_{\mathcal{R}} C[t] \to^*_{\mathcal{R}} C[C[t]] \to^*_{\mathcal{R}} C[C[C[t]]] \to^*_{\mathcal{R}} \cdots$ なる書き換えが構成できる．従って，前の例 25 で示した要件から，A は生成的でないと考えてしまうかもしれない．しかし，

$$
\begin{array}{lll}
\mathsf{A} & \to_{\mathcal{R}} & \mathsf{g}(\mathsf{B}, \mathsf{A}) \\
& \to_{\mathcal{R}} & \mathsf{g}(0 :: \mathsf{A}, \mathsf{A}) \\
& \to_{\mathcal{R}} & 0 :: \mathsf{g}(\mathsf{A}, \mathsf{A}) \\
& \to_{\mathcal{R}} & 0 :: \mathsf{g}(\mathsf{g}(\mathsf{B}, \mathsf{A}), \mathsf{A}) \\
& \to_{\mathcal{R}} & 0 :: \mathsf{g}(\mathsf{g}(0 :: \mathsf{A}, \mathsf{A}), \mathsf{A}) \\
& \to_{\mathcal{R}} & 0 :: \mathsf{g}(0 :: \mathsf{g}(\mathsf{A}, \mathsf{A}), \mathsf{A}) \\
& \to_{\mathcal{R}} & 0 :: 0 :: \mathsf{g}(\mathsf{A}, \mathsf{g}(\mathsf{A}, \mathsf{A})) \\
& \to_{\mathcal{R}} & \cdots
\end{array}
$$

のような書き換え列を考えると，0 が出力されることがわかる．実際，関数記号として g, A のみを含む完全 2 分木の形をもつ高さ n の有限基底項を t とすると，完全 2 分木の形をもつ高さ $n+1$ の有限項 s に対して，$t \to^*_{\mathcal{R}} 0 :: \cdots :: 0 :: s$ となることを n に関する帰納法で示すことができる．従って，A の ω-正規形は 0^ω となり，A は生成的である．

この例のように，$t \to^*_{\mathcal{R}} C[t]_p$ となる書き換え列が存在しても生成的である場合がある．それは，先の例 26 に示したように，文脈 C において，位置 p とは並列な位置での書き換えによって，$C[t]_p$ における根位置での書き換えが可能になる場合である．従って，$t \to^*_{\mathcal{R}} C[t]_p \to^*_{\mathcal{R}} \cdots$ が生成性の反例となるためには，文脈 C の p より上の位置で，書き換えが可能にならないことを保証する必要がある．

項 t とその位置 p について，p より上の位置で書き換えが起こらず，位置 p が保存される条件を考える．

†4 もっとも，これは理論的な保証を意味するだけであって，前者が現実的な時間で 0 を出力し続けることを意味するわけではない．このような，計算の効率性を考慮した生成性も興味深い問題ではあるが，本論文では考えない．

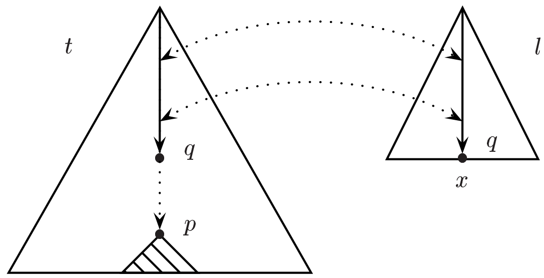

図 1　経路的照合性

先のように位置 p と並列な位置での書き換えが起こるとすると，位置 p と並列な位置についての情報を用いることはできない．従って，根位置 ε から位置 p への経路上の情報のみにもとづき，書き換え規則との照合可能性を排除する必要がある．このようなアイデアにもとづき，次のように経路的照合性および経路保存位置の概念を導入する．

定義 27 (経路的照合性と経路保存位置)　1. l, t を項，$p \in \mathrm{Pos}(t)$ とする．性質 $\exists q \leq p.\ [l(q) \in \mathcal{V} \wedge \forall q' < q.\ t(q') = l(q')]$ が成立するとき，位置 p に関して t は l に**経路的に照合する**という．(経路的に照合しないことを経路的に非照合であるという．)

2. $\mathcal{R}$ をストリーム項書き換えシステム，t を項とする．位置 $p \in \mathrm{Pos}(t)$ が**経路保存**であるとは，任意の位置 $o < p$ と任意の書き換え規則 $l \to r \in \mathcal{R}$ について，位置 $p \backslash o$ に関して $t|_o$ が l に経路的に照合しないときをいう．経路保存であるような項 t の位置を項 t の**経路保存位置**とよぶ．

図 1 に，経路的照合性の概念図を示す．項 t が項 l に (位置 p に関して) 経路的に照合する場合には，l は t と照合する可能性がある．また，l が t と照合していなくても，p とは並列な位置の書き換えによっては t を書き換えた結果と l が照合する可能性がある．一方，経路的に照合しない場合には，p とは並列な位置で書き換えがどのように起ころうとも，l が照合する可能性がない．従って，位置 p が項 t の経路保存位置であれば，p とは並列な位置で書き換えがどのように起ころうとも，根位置 ε から位置 p への経路上では書き換えが起こらない．

定義 27 では，位置 p と並列な位置での書き換えが起きる場合を考えた．次に，p より下の位置での書き換えによって，位置 p より上の位置での書き換えが可能になる場合を考える．これは，p より下の位置での書き換えによっても，それまで可能でなかった位置 p より上の位置での書き換えが可能となる場合があるためである．このための条件を与えることは一般的には困難であるが，生成性の反例の候補の場合には，構成子 cons が根位置で出現しないという特徴があるため，この事実を利用することができる．このようなアイデアにもとづき，次のようにストリーム照合性およびストリーム安定位置の概念を導入する．

定義 28 (ストリーム照合性とストリーム安定位置)

1. l, t を項，$p \in \mathrm{Pos}(t)$ とする．$l(p) \neq \mathsf{cons}$ かつ性質 $\forall q < p.\ t(q) = l(q)$ が成立するとき，位置 p に関して t が l に**ストリーム照合する**という．(ストリーム照合しないことをストリーム非照合であるという．)
2. $\mathcal{R}$ をストリーム項書き換えシステム，t を項とする．位置 $p \in \mathrm{Pos}(t)$ が**ストリーム安定**であるとは，任意の位置 $o < p$ と任意の書き換え規則 $l \to r \in \mathcal{R}$ について，位置 $p \backslash o$ に関して $t|_o$ が l にストリーム照合しないときをいう．ストリーム安定であるような項 t の位置を項 t の**ストリーム安定位置**とよぶ．

図 2 に，ストリーム照合性の概念図を示す．項 t が項 l に (位置 p に関して) ストリーム照合する場合は，l は t と照合する可能性がある．l が t と照合していなくても，部分項 $t|_p$ の書き換えによって，l が照合する可能性がある．一方，項 t が項 l に (位置 p に関

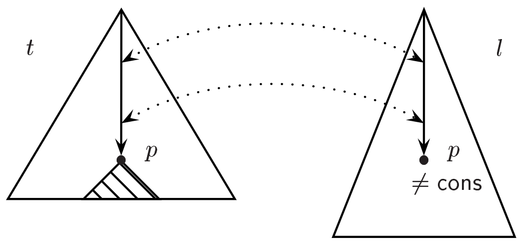

図 2　ストリーム照合性

して) ストリーム照合しない場合には，部分項 $t|_p$ がどのように書き換えられたとしても，$l(p) \neq \mathsf{cons}$ であるような l が照合することはない．従って，位置 p が項 t のストリーム安定位置であれば，部分項 $t|_p$ 内の位置で書き換えがどのように起ころうとも，$t(p)$ が cons となることがなければ，$l(p) \neq \mathsf{cons}$ であるような書き換え規則の左辺 l とも $l(p) = \mathsf{cons}$ であるような書き換え規則の左辺 l とも照合しないので，根位置 ε から位置 p への経路上では書き換えが起こらない．

最後に，経路保存性とストリーム安定性を合わせて，位置の安定性を定義する．

定義 29 (安定位置) $\mathcal{R}$ をストリーム項書き換えシステム，t を項とする．$p \in \mathrm{Pos}(t)$ が経路保存かつストリーム安定であるとき，位置 p を項 t における**安定位置**とよぶ．位置 p が項 t における安定位置であることを $\mathrm{stable\text{-}pos}(t, p)$ と記す．また，簡便のため，次の定義を用いる．l, t を項，$p \in \mathrm{Pos}(t)$ とする．位置 p に関して t が l に，経路的に非照合かつストリーム非照合であるとき，位置 p に関して t は l に**経路的にストリーム非照合である**という．(経路的にストリーム非照合でないとき，経路的にストリーム照合であるという．) あきらかに，任意の位置 $o < p$ と任意の書き換え規則 $l \to r \in \mathcal{R}$ について，位置 $p \backslash o$ に関して $t|_o$ が l に経路的にストリーム非照合であるとき，その時に限り，p は安定位置である．

定義よりあきらかに位置 $\epsilon \in \mathrm{Pos}(t)$ は項 t における安定位置であることに注意する．

例 30 (安定位置 (1)) $\mathcal{F} = \{\mathsf{even}, \mathsf{odd} : S \to S, \mathsf{J} : S, 0, 1 : D, \mathsf{cons} : D \times S \to S\}$ として，次のストリーム項書き換えシステム $\mathcal{R}$ ([10]) を考える．

$$\mathcal{R} = \left\{ \begin{array}{l} \mathsf{even}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{odd}(xs)) \\ \mathsf{odd}(\mathsf{cons}(x, xs)) \to \mathsf{even}(xs) \\ \mathsf{J} \to \mathsf{cons}(0, \mathsf{cons}(1, \mathsf{even}(\mathsf{J}))) \end{array} \right\}$$

今，位置 $1 \in \mathrm{Pos}(t)$ が $t = \mathsf{even}(\mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J})))))$ における安定位置であることを示す．これには，任意の $l \to r \in \mathcal{R}$ について，位置 $1 \backslash \epsilon (= 1)$ に関して $t|_\epsilon (= t)$ が l に経路的にストリーム非照合であることを示せばよい．

<u>(1) $l = \mathsf{even}(\mathsf{cons}(x,\ xs))$ の場合</u>．$q \leq 1$ とすると，$l(q) = \mathsf{even}$ または $l(q) = \mathsf{cons}$．よって，位置 1 に関して t は l に経路的に照合しない．また，$l(1) = \mathsf{cons}$ が成立するので，位置 1 に関して t は l にストリーム照合しない．

<u>(2) $l = \mathsf{odd}(\mathsf{cons}(x, xs))$ の場合</u>．$q \leq 1$ とすると，$l(q) = \mathsf{odd}$ または $l(q) = \mathsf{cons}$．よって，位置 1 に関して t は l に経路的に照合しない．また，$l(1) = \mathsf{cons}$ が成立するので，位置 1 に関して t は l にストリーム照合しない．

<u>(3) $l = \mathsf{J}$ の場合</u>．$q \leq 1$ とすると，$l(q) = \mathsf{J}$ または $l(q) = \perp$．よって，位置 1 に関して t は l に経路的に照合しない．また，$t(\epsilon) = \mathsf{even} \neq \mathsf{J} = l(\epsilon)$ であるから，位置 1 に関して t は l にストリーム照合しない．

以上より，$\mathrm{stable\text{-}pos}(t, 1)$ が成立する．

例 31 (安定位置 (2)) $\mathcal{F} = \{\mathsf{f} : S \to S, \bullet : D, \mathsf{cons} : D \times S \to S\}$ として，次のストリーム項書き換えシステム $\mathcal{R}$ ([18]) を考える．

$$\mathcal{R} = \left\{ \ \mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs))) \to \mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs))) \ \right\}$$

今，位置 $12 \in \mathrm{Pos}(t)$ が $t = \mathsf{f}(\mathsf{cons}(\bullet,\ \mathsf{f}(\mathsf{f}(xs))))$ に

```
fun disprove-gen-prod (t_0, R) =
    fun ckeck (x, s_0, s, σ) =
        if ∃p. (x = true ∧ p = ϵ) ∨ (∃q < p. s(q) ∈ D), stable-pos(s, p),
           ∃θ. θ = mgu_inf(REN(s_0 σ), s|_p), ∃δ ▶ θ. match_inf(s_0 σ δ*, s|_p δ*)
        then SOME s_0 σ δ* else NONE
    fun narrow (x, s_0, s, σ) = [((x ∨ q = ϵ), s_0, s', ρ ∘ σ) | s ⇝°_{q,ρ} s']
    fun start s_0 = narrow (false, s_0, s_0, ∅)
    fun tail (cons(h, s)) = if s(ϵ) = cons then (tail s) else s
    fun step [] = FAIL
      | step ((x, s_0, s, σ) :: xs) =
        if s(ϵ) = cons then step (xs @ (start (tail s)))
        else case check (x, s_0, s, σ) of
                 SOME u ⇒ SUCCESS u | NONE ⇒ step (xs @ (narrow (x, s_0, s, σ)))
in step (start t_0) end
```

プログラム 2　一般生成性の反証手続き

おける安定位置であることを示す．これには，任意の位置 $q < 12$ について，位置 $12 \backslash q$ に関して $t|_q$ が $l = \mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs)))$ に経路的にストリーム非照合であることを示せばよい．

(1) $q = \epsilon$ の場合．$q' \leq 12$ とすると，$l(q') = \mathsf{f}$ または $l(q') = \mathsf{cons}$．よって，$t|_\epsilon$ は l に経路的に照合しない．また，$l(12) = \mathsf{cons}$ であるから，位置 12 に関して項 $t|_\epsilon$ は l にストリーム照合しない．

(2) $q = 1$ の場合．位置 2 に関して項 $t|_1 = \mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs)))$ が l に経路的にストリーム照合しないことを示す．$q' \leq 2$ とすると，$l(q') = \mathsf{f}$ または $l(q') = \bot$．よって，$t|_1$ は l に経路的に照合しない．また，$t|_1(\epsilon) = \mathsf{cons} \neq \mathsf{f} = l(\epsilon)$ であるから，位置 2 に関して項 $t|_1$ が l にストリーム照合しない．

以上より，$\text{stable-pos}(t, 12)$ が成立する．

次に，安定位置の概念を用いた生成性の十分条件を与えるが，その前に条件で用いる最外書き換え列について紹介する．$s \to_p t$ を書き換えステップとする．任意の位置 $q < p$ なる $q \in \mathrm{Pos}(s)$ について $s \to_q t'$ となる項 t' が存在しないとき，$s \to_p t$ を**最外書き換えステップ**とよび，$s \overset{\circ}{\to}_p t$ と記す．このとき，部分項 $s|_p$ を**最外リデックス**とよぶ．また，文脈から明かな場合や必要ない場合は p を省略する．最外書き換えステップで構成される書き換え列 $s \overset{\circ}{\to}{}^* t$ を**最外書き換え列**とよぶ．

安定位置の概念を用いて次のような非生成性の十分条件を与えることが出来る．この補題は，反証手続きの健全性証明の鍵となる．

補題 32 (非生成性の十分条件) 位置 $p\ (\neq \epsilon) \in \mathrm{Pos}(t)$ を項 t における安定位置とし，$s(\epsilon), t(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$，ある位置 $q < p$ について $t(q) \in \mathcal{D}$ とする．また，任意の代入 η について，書き換え列 $s\eta \overset{\circ}{\to}{}^+_{\mathcal{R}} t\eta[s\theta\eta]_p$ が存在するものとする．このとき，項 s は $\mathcal{R}$ において生成的でない．

(証明) 付録 A に示す．□

次に，本節で反証手続きを与える一般生成性について説明する．

定義 33 (一般生成性) $\mathcal{R}$ をストリーム項書き換えシステム，t をストリーム型項とする．t が $\mathcal{R}$ において**一般生成的**(generally productive) である (t が**一般生成性をもつ**) とは，任意の代入 σ について $t\sigma$ が $\mathcal{R}$ において生成的であるときをいう．t が $\mathcal{R}$ において一般生成的かどうかを問う問題を**一般生成性問題**とよぶ．

定義からあきらかに項が一般生成的であれば生成的であるが，逆は成立しない．また，基底項については，一般生成性問題と生成性問題は同値である．以下の例で，一般生成性問題を考える動機を説明する．

例 34 例 24 を変更して得られる以下のストリーム項

書き換えシステムを考える．

$$\left\{\begin{array}{l}\mathsf{m}_0(\mathsf{cons}(0,xs)) \to \mathsf{m}_1(xs)\\ \mathsf{m}_1(\mathsf{cons}(x,xs)) \to \mathsf{cons}(x,\mathsf{m}_0(xs))\\ \mathsf{m}_0(\mathsf{cons}(1,xs)) \to \mathsf{cons}(1,\mathsf{m}_0(xs))\\ \mathsf{C}(\mathsf{cons}(x,\mathsf{cons}(y,xs)))\\ \qquad\qquad \to \mathsf{cons}(x,\mathsf{cons}(y,\mathsf{m}_0(\mathsf{C}(xs))))\end{array}\right\}$$

このとき，例 24 と同様にして，$\mathsf{C}(0 :: 0 :: \cdots)$ は生成的でない．一方，$\mathsf{C}(1 :: 1 :: \cdots)$ は生成的である．従って，$\mathsf{C}(xs)$ は一般生成的でない．生成性は，ある特定の項に対してストリームの生成を考えるが，それでは，外部からストリームを入力としてとる場合や入力となるストリームが定まっていない場合などに対応できない．このような場合には，どのような入力についても生成的となることを保証する一般生成性の方がより自然な性質ではないかと考えられる．

一般生成性の反証手続きを与える前に，手続きで用いる最外ナローイングについて紹介する．$s \rightsquigarrow_{p,\sigma} t$ をナローイングステップとする．任意の $q < p$ なる $q \in \mathrm{Pos}(s)$ について $s \rightsquigarrow_{q,\sigma'} t'$ となる t' が存在しないとき，$s \rightsquigarrow_{p,\sigma} t$ を**最外ナローイングステップ**とよぶ．$s \rightsquigarrow_{p,\sigma} t$ が最外ナローイングステップであるとき $s \overset{\circ}{\rightsquigarrow}_{p,\sigma} t$ と記す．$s \overset{\circ}{\rightsquigarrow}{}^{*}_{\sigma} t$ は $s = s_1 \overset{\circ}{\rightsquigarrow}_{\theta_1} s_2 \overset{\circ}{\rightsquigarrow}_{\theta_2} \cdots \overset{\circ}{\rightsquigarrow}_{\theta_{n-1}} s_n = t$, $\sigma = \theta_{n-1} \circ \cdots \circ \theta_1$ $(n \geq 1)$ となることを表わし，$n \geq 2$ のとき，$s \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} t$ と表わす．

手続き 2 (一般生成性の反証手続き) 一般生成性の反証手続き disprove-gen-prod をプログラム 2 に与える．手続きへの入力は有限ストリーム型項 t_0 とストリーム項書き換えシステム $\mathcal{R}$ である．反証に成功した場合にはある正則項 t について SUCCESS t を，失敗した場合は FAIL を返す．SUCCESS t を返す場合，t は，ある代入 σ について $t_0\sigma \to^* c_0 :: c_1 :: \cdots :: C[t]$ となり，かつ，根記号が定義記号であるような ω-正規形をもつ項となっている．また手続きが停止しない (発散する) 場合もある．

手続きにおける実質的な反証条件の判定は 2 行目の check 関数で行っている．check 関数では，4 節で与えた ω-強頭部正規化可能性の反証のときと同様のアイデアを用いて反例候補を構成するとともに，補題 32 の非生成性十分条件をチェックする．9 行目の step 関数は，この判定を試みる候補を幅優先探索で構成し，check 関数を候補に適用する．判定候補となる 4 つ組 (x, s_0, s, σ) は，$s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} s$ を満たし，x はこのナローイング列が根ナローイングステップを含むかどうかを示すブール値となっている．$s(\epsilon) = \mathtt{cons}$ のときは，11 行目でストリームの頭部を削除し，新たに候補を構成して末尾に追加する．そうでないときは，check 関数にて，一般生成性の反証条件を調べる．check 関数による判定に失敗した場合には，ナローイング列 $s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} s$ を延長し，候補リストに入れる．

以下の定理は，プログラム 2 の反証手続きの正しさ (健全性) を述べている．つまり，手続きに成功する場合には一般生成的でないことが保証される．

定理 35 (一般生成性反証手続きの健全性) ストリーム項書き換えシステム $\mathcal{R}$ と有限項 t_0 に対して，プログラム 2 の一般生成性の反証手続きが成功するとき，項 t_0 は $\mathcal{R}$ において一般生成的でない．

(証明) 付録 A に示す． □

手続き 2 にもとづいて，一般生成性の反証に成功する例を以下に示す．

例 36 (一般生成性反証手続きの適用例 (1)) $\mathcal{R}$ を例 30 で与えたストリーム項書き換えシステムとする．disprove-gen-prod $(\mathsf{J}, \mathcal{R})$ が成功することを示す．最初，step (start J) がよびだされる．J からの唯一つ可能な最外ナローイングステップは $\mathsf{J} \overset{\circ}{\rightsquigarrow}_{\epsilon,\emptyset} \mathsf{cons}(0,\mathsf{cons}(1,\mathsf{even}(\mathsf{J})))$．11 行目で $\mathsf{cons}(0,\mathsf{cons}(1,\mathsf{even}(\mathsf{J})))$ に tail が適用され，step (start even(J)) がよびだされる．even(J) からの唯一つ可能な最外ナローイングステップは $\mathsf{even}(\mathsf{J}) \overset{\circ}{\rightsquigarrow}_{1,\emptyset} \mathsf{even}(\mathsf{cons}(0,\mathsf{cons}(1,\mathsf{even}(\mathsf{J}))))$．13 行目で step $[(\mathtt{false}, \mathsf{even}(\mathsf{J}), \mathsf{even}(\mathsf{cons}(0,\mathsf{cons}(1,\mathsf{even}(\mathsf{J})))), \emptyset)]$ がよびだされる．次の check $(\mathtt{false}, \mathsf{even}(\mathsf{J}), \mathsf{even}(\mathsf{cons}(0,\mathsf{cons}(1,\mathsf{even}(\mathsf{J})))), \emptyset)$ の評価は 4 行目で単一化に失敗するため，NONE となる．そこで，また可能な最外ナローイングステップを計算する．その後，同様の処理を 8 回繰り返すと，step $[(\mathtt{true}, \mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1,\mathsf{even}(\mathsf{J})))) \ (= s_0), \mathsf{even}(\mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1,\mathsf{even}(\mathsf{J}))))) \ (= s), \emptyset)]$ のよびだしになる．check $(\mathtt{true}, s_0, s, \emptyset)$ は $p = 1$ ととると，$s(\epsilon) = \mathsf{even}$ であるから $\exists q < p.\ s(q) \in \mathcal{D}$ が成立し，

例 30 で示したように stable-pos(s,p) が成立する．また，$\theta = \mathrm{mgu}_{\mathit{inf}}(s_0, s|_1) = \emptyset$ となり，$\delta = \emptyset \blacktriangleright \theta$ をとると，$\delta^\star = \emptyset$．よって，$\mathrm{match}_{\mathit{inf}}(s_0\sigma\delta^\star, s|_1\delta^\star) = \emptyset$．従って，5 行目で SUCCESS $\mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J}))))$ が返り，手続きが成功する．

実際，次のように項 J は $\mathcal{R}$ において (一般) 生成的でない．(以下で，$q = 2222$，$C[\,] = \mathsf{cons}(0, \mathsf{cons}(1, \mathsf{cons}(0, \mathsf{cons}(0, \Box))))$．) 下線は書き換え箇所を示す．

J
$\stackrel{\circ}{\to}^+ C[s_0]_q = C[\mathsf{odd}(\underline{\mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J})))})]_q$
$\stackrel{\circ}{\to}_{q.1} C[\mathsf{odd}(\mathsf{even}(\mathsf{even}(\underline{\mathsf{J}})))]_q$
$\stackrel{\circ}{\to}_{q.111} C[\mathsf{odd}(\mathsf{even}(\underline{\mathsf{even}(\mathsf{cons}(0, \mathsf{cons}(1, \mathsf{even}(\mathsf{J}))))}))]_q$
$\stackrel{\circ}{\to}_{q.11} C[\mathsf{odd}(\underline{\mathsf{even}(\mathsf{cons}(0, \mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J})))))})]_q$
$\stackrel{\circ}{\to}_{q.1} C[\underline{\mathsf{odd}(\mathsf{cons}(0, \mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J}))))))}]_q$
$\stackrel{\circ}{\to}_{q.\epsilon} C[\mathsf{even}(\mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1, \mathsf{even}(\mathsf{J})))))]_q$
$= C[\mathsf{even}(s_0)]_q$
$\stackrel{\circ}{\to}^+ C[\mathsf{even}(\mathsf{even}(s_0))]_q$
$\stackrel{\circ}{\to}^+ \cdots \stackrel{\circ}{\to}^\omega C[\mathsf{even}^\omega]_q$．よって，項 J の (唯一つの) ω-正規形は構成子項とならないため，項 J は $\mathcal{R}$ において生成的でない．

例 37 (一般生成性反証手続きの適用例 (2)) $\mathcal{R}$ を例 31 で与えたストリーム項書き換えシステムとする．`disprove-gen-prod` $(\mathsf{f}(xs), \mathcal{R})$ が成功することを示す．最初に，`step` (`start` $\mathsf{f}(xs)$) がよびだされる．$\mathsf{f}(xs)$ からの唯一つ可能な最外ナローイングステップは $\mathsf{f}(xs) \stackrel{\circ}{\rightsquigarrow}_{\epsilon,\rho} \mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs_1)))$ ($\rho = \{xs := \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_1))\}$)．11 行目で $\mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs_1)))$ に `tail` が適用され，`step` (`start` $\mathsf{f}(\mathsf{f}(xs_1))$) がよびだされる．$\mathsf{f}(\mathsf{f}(xs_1))$ からの唯一つ可能な最外ナローイングステップは $\mathsf{f}(\mathsf{f}(xs_1)) \stackrel{\circ}{\rightsquigarrow}_{1,\rho} \mathsf{f}(\mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs_2))))$ ($\rho = \{xs_1 := \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_2))\}$)．次に，`step` [(`false`, $\mathsf{f}(\mathsf{f}(xs_1))$ $(= s_0)$, $\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs_2))))$ $(= s)$, $\{\ xs_1 := \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_2))\ \}$ $(= \sigma)$,)] がよびだされ，`check` (`false`, s_0, s, σ) の評価に入る．ここで，$p = 12$ をとると，$s(\epsilon) = \mathsf{f} \in \mathcal{D}$ より $\exists q < p.\, s(q) \in \mathcal{D}$ が成立し，例 31 で示したように stable-pos(s,p) が成立する．また $\theta = \mathrm{mgu}_{\mathit{inf}}(\mathrm{REN}(s_0\sigma), s|_p) = \mathrm{mgu}_{\mathit{inf}}(\mathsf{f}(\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_2')))), \mathsf{f}(\mathsf{f}(xs_2))) = \{\ xs_2 := \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_2'))\ \}$．ここで，$\delta = \{\ xs_2 := \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs_2)),\ xs_2' := xs_2\} \blacktriangleright \theta$ をとると，$\delta^\star = \{\ xs_2 := \mu xs.\, \mathsf{cons}(\bullet, xs),\ xs_2' := \mu xs.\, \mathsf{cons}(\bullet, xs)\ \}$ となり，$\mathrm{match}_{\mathit{inf}}(s_0\sigma\delta^\star, s|_p\delta^\star) = \mathrm{match}_{\mathit{inf}}(\mathsf{f}(\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \mu xs.\, \mathsf{cons}(\bullet, xs))))), \mathsf{f}(\mathsf{f}(\mu xs.\, \mathsf{cons}(\bullet, xs)))) = \emptyset$ となる．5 行目で SUCCESS $\mathsf{f}(\mathsf{f}(\mu xs.\, \mathsf{cons}(\bullet, xs)))$ が返り，手続きが成功する．

実際，次のように項 $\mathsf{f}(\mathsf{f}(\mu xs.\, \mathsf{cons}(\bullet, xs)))$ は $\mathcal{R}$ において生成的でない．(以下で，$C[\,] = \mathsf{f}(\mathsf{cons}(\bullet, \Box))$．)

$\mathsf{f}(\underline{\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \cdots))))})$
$\stackrel{\circ}{\to}_1 \mathsf{f}(\mathsf{cons}(\bullet, \mathsf{f}(\ \underline{\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \cdots))))})))$
$= C[\mathsf{f}(\underline{\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \cdots))))})]$
$\stackrel{\circ}{\to}_{121} C[C[\mathsf{f}(\underline{\mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \mathsf{cons}(\bullet, \cdots))))})]]$
$\stackrel{\circ}{\to}_{12121} \cdots$
$\stackrel{\circ}{\to}^\omega \mu xs.\, C[xs]$．従って，項 $\mathsf{f}(xs)$ は $\mathcal{R}$ において一般生成的でない．

6 実装および実験

4 節で与えた ω-強頭部正規化可能性の反証手続き (プログラム 1) および 5 節で与えた一般生成性の反証手続き (プログラム 2) を SML/NJ[†5]を用いて実装した．

まず，3 節の結果にもとづき，両者の手続きで用いる $\mathrm{mgu}_{\mathit{inf}}$ および $\mathrm{match}_{\mathit{inf}}$ の計算手続きには Jaffer [20] の手続きを用いた．

次に，我々の実装した反証手続きで用いた近似について説明する．まず，両者の反証手続きでは，与えられた代入 θ から，適当な条件を満たす $\delta \blacktriangleright \theta$ なる再帰式表現 δ を探す手続きがある (プログラム 1 の 3 行目，プログラム 2 の 4 行目)．関係 $\blacktriangleright$ の定義より，与えられた代入 θ に対して $\delta \blacktriangleright \theta$ を満たす再帰式表現 δ の個数は有限であるが，有限集合 X, Y について関数 $\sigma : X \to Y$ の個数は $|Y|^{|X|}$ 個あるため，すべての δ に対して条件を満たすかを調べるのは現実的でないことはあきらかである．このため，以下のヒューリスティクスを用いて，δ の候補の絞り込みを行った．

> 代入 σ として，空代入，および，それぞれの $x := t \in \theta$ に対して得られる $\sigma = \{y := x \mid y \in \mathcal{V}(t) \setminus \mathrm{dom}(\theta),$

†5 http://www.smlnj.org/

x, y は同じソートをもつ (プログラム 2 の場合)$\}$ のみを対象として，$\delta = \sigma \circ \theta$ を構成する．

また，両者の反証手続きでは，可能なナローイングステップの結果を $[t \mid s \rightsquigarrow t, \cdots]$ のようにリストにする手続きがある (プログラム 1 の 5 行目，プログラム 2 の 6 行目)．s を正則項とするとき，一般に，このリスト $[t \mid s \rightsquigarrow t, \cdots]$ は有限リストにならない．(例えば，$\mathcal{R} = \{\mathsf{f}(x) \rightarrow \mathsf{g}(x)\}$ とおくとき，$\mathsf{f}^\omega \rightsquigarrow_{1^i,\mathcal{R}} \mathsf{f}^i(\mathsf{g}(\mathsf{f}^\omega))$ $(i = 0, 1, 2, \ldots)$ となってしまう．) また，s が有限項の場合でも，無限項上のナローイングを行うとその結果が有限項とならない場合がある．このため，我々の実装では，有限ナローイングステップのみを用いた．

次に，この実装を用いて行った反証手続きの実験について報告する．実験には，1.2GHz Intel Pentium プロセッサの CPU，1GB のメモリをもつ PC を用いた．step 関数の適用を 1 ステップとカウントし，ステップ数の最大を 20 ステップとした．この最大値を越えた場合は，手続きは停止しない (“発散”) とみなした．実行時間 (CPU 計算時間) の単位は全てミリ秒である．

まず，ω-強頭部正規化可能性をもたない項書き換えシステム 12 例を構成し，ω-強頭部正規化可能性の反証実験を行った．結果を表 1 に示す．実験の結果，12 例のうち 10 例について成功した．

例 2 と例 3 は似た書き換え規則をもつが，例 2 で生成される反例は有限項となっている一方，例 3 では反例として無限項が生成される．例 4 以降は，複数のナローイングステップが必要になる例である．例 6 は，ナローイングによって得られた代入 σ によって，無限正則項が構成される例である．例 7 は根でないナローイングステップが必要になる場合である．例 8 はナローイングの戦略によって，発散または成功する例である．

例 $10, 12$ については反証手続きが成功しなかった．(例 10 は停止して失敗し，例 12 は発散した．) 例 $10, 12$ において，無限列 $\mathsf{f}(\mathsf{g}^\omega) \rightarrow^\epsilon_\mathcal{R} \mathsf{h}(\mathsf{g}^\omega, \mathsf{g}^\omega) \rightarrow^\epsilon_\mathcal{R} \mathsf{f}(\mathsf{g}^\omega) \rightarrow^\epsilon_\mathcal{R} \cdots$ を構成するためには，無限項のナローイングステップ $\mathsf{h}(\mathsf{g}(x), \mathsf{g}(x)) \rightsquigarrow^\epsilon_{\{x \mapsto \mathsf{g}^\omega\}} \mathsf{f}(\mathsf{g}^\omega)$ が必要であるが，我々の手続きにおけるナローイングは有限ナローイングの近似を用いたためである．なお，本例の場合には，例えば，非有限項については根位置でのみの無限項上のナローイングステップを行う，などとすれば扱うことができるが，例 7 のように根でないナローイングステップが必要になる場合もあるため，本質的な解決にはならない．一方，例 11 では，無限項のナローイングステップがなくても，書き換え規則 $\mathsf{g}(\mathsf{a}) \rightarrow \mathsf{a}$ があるため，$\mathsf{f}(x) \rightsquigarrow \mathsf{h}(\mathsf{g}(x), \mathsf{g}(x)) \rightsquigarrow \mathsf{f}(\mathsf{a}, \mathsf{g}(\mathsf{a})) \rightsquigarrow \mathsf{f}(\mathsf{a})$ と反例の発見に成功し，反証に成功する．

岩見[19] は組合せ子の書き換え規則に対して，ω-強頭部正規化可能性が成立しないことを，反例を構成することによって報告している．文献[19] で示された組合せ子の書き換え規則の例に対して提案手法を適用した結果，ω-強頭部正規化可能性が成立しない 25 例に対して，提案手法にもとづく ω-強頭部正規化可能性の自動反証に成功した．また，ω-強頭部正規化可能性が従来検証されていなかった組合せ子 F^* ([12]) についても自動反証に成功した．

実験を行った組合せ子と，その書き換え規則，実験結果および構成された反例を表 2 に示す．なお，表中では組合せ子の適用演算子 $\cdot$ は省略している．表下段の組合せ子 Y, U, U^2, O の書き換え規則は ω-強頭部正規化可能性をもつ ([19] [33]) ため，反証には成功しない．

次に，一般生成性の反証実験を行った結果について報告する．実験例としては，文献[3] [10] [15] [18] [28] [34] から抜粋した (主として) 一般生成的でない 20 例を用いた．表 3 に実験に用いたストリーム項書き換えシステム，そして，表 4 に実験結果を示す．一般生成的である 8 例については，反証手続きは発散した．一般生成的でない 12 例について，11 例について反証手続きが成功し，1 例については発散した．

一般生成的でないにもかかわらず反証手続きが発散したのは，例 20 である．例 20 が一般生成的でない原因は，ストリーム項書き換えシステム $\mathcal{R}_{20}$ の 1 番目の書き換え規則の右辺 $\mathsf{cons}(1, \mathsf{cons}(\mathsf{hd}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L})))), \mathsf{L}))$ のストリーム第 2 要素 $\mathsf{hd}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L}))))$ が ω-正規形をもたないことにある．しかし，我々の一般生成性反証手続きでは，ストリームの要素については ω-正規形をもつとみなしてチェックを行わないため，この部分項から生じる非生成性を検出できない．

表 1 ω-強頭部正規化可能性の反証実験

番号	項書き換えシステム $\mathcal{R}$	結果	実行時間	ステップ数	構成された反例
1	$\{ \mathsf{a} \to \mathsf{a} \}$	成功	0	1	a
2	$\{ \mathsf{f}(x) \to \mathsf{f}(\mathsf{f}(x)) \}$	成功	0	1	$\mathsf{f}(x)$
3	$\{ \mathsf{f}(\mathsf{f}(x)) \to \mathsf{f}(x) \}$	成功	0	1	f^ω
4	$\{ \mathsf{a} \to \mathsf{b}, \ \mathsf{b} \to \mathsf{a} \}$	成功	0	3	a
5	$\{ \mathsf{f}(x) \to \mathsf{g}(x), \ \mathsf{g}(x) \to \mathsf{h}(x), \ \mathsf{h}(x) \to \mathsf{f}(x) \}$	成功	0	7	$\mathsf{f}(x)$
6	$\{ \mathsf{f}(x) \to \mathsf{g}(x), \ \mathsf{g}(\mathsf{h}(x)) \to \mathsf{f}(x) \}$	成功	0	3	$\mathsf{f}(\mathsf{h}^\omega)$
7	$\{ \mathsf{f}(x) \to \mathsf{g}(\mathsf{a}), \ \mathsf{a} \to \mathsf{b}, \ \mathsf{g}(\mathsf{b}) \to \mathsf{f}(\mathsf{a}) \}$	成功	0	7	$\mathsf{f}(x)$
8	$\{ \mathsf{f}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{f}(x)), \ \mathsf{g}(\mathsf{f}(x)) \to \mathsf{f}(x) \}$	成功	0	3	$\mathsf{f}(\mathsf{g}^\omega)$
9	$\{ \mathsf{f}(\mathsf{g}(x)) \to \mathsf{g}(\mathsf{f}(x)), \ \mathsf{g}(\mathsf{f}(x)) \to \mathsf{f}(x), \ \mathsf{a} \to \mathsf{g}(\mathsf{f}(\mathsf{a})) \}$	成功	0	4	$\mathsf{f}(\mathsf{g}^\omega)$
10	$\{ \mathsf{f}(x) \to \mathsf{h}(\mathsf{g}(x), \mathsf{g}(x)), \ \mathsf{h}(x, \mathsf{g}(x)) \to \mathsf{f}(x) \}$	失敗	0	3	—
11	$\{ \mathsf{f}(x) \to \mathsf{h}(\mathsf{g}(x), \mathsf{g}(x)), \ \mathsf{h}(x, \mathsf{g}(x)) \to \mathsf{f}(x), \ \mathsf{g}(\mathsf{a}) \to \mathsf{a} \}$	成功	0	6	$\mathsf{f}(\mathsf{g}^\omega)$
12	$\{ \mathsf{f}(x) \to \mathsf{h}(\mathsf{g}(x), \mathsf{g}(x)), \ \mathsf{h}(x, \mathsf{g}(x)) \to \mathsf{f}(x), \ \mathsf{a} \to \mathsf{g}(\mathsf{a}) \}$	発散	1	20	—

また，文献[9][10]にもとづく生成性判定ツール[†6]を用いて，文献[9][10]の手法との比較実験を行った．結果を表4の最右欄に示す．ただし，例3, 4, 7, 9–11については，一般生成性問題を生成性問題に変更するために，発見された反例もしくは対象項を生成する適当な書き換え規則を追加して検証を行なった．(例えば，例3では，$\mathcal{R}_3$ に書き換え規則 $\mathsf{M} \to \mathsf{c}$, $\mathsf{c} \to \mathsf{cons}(1, \mathsf{c})$ を追加し，M に関する生成性問題を試みた．) 文献[9][10]の手法では，生成的でない12例のうち5例の反証に成功した．反証に成功した5例は我々の反証手続きでも成功する．我々の反証手続きで成功し，文献[9][10]で反証に失敗する7例は，文献[9][10]にもとづく従来手法では理論的に取り扱えない例である．

文献[9][10]の手法はデータ盲目 (data-oblivious, データ型の項を同一視する) という近似にもとづいている．そして，この近似が問題にならないように，対象となるストリーム項書き換えシステムに制限を課している．上記の7例はこの制限のため取り扱えない．また，データ盲目による近似は，判定可能なアルゴリズムの構築を可能にするという利点もあるが，逆に，この近似のため，データ型の項の差異が生成性の有無に影響する場合，彼らのアプローチは原理的に有効でない．対照的に，我々の反証手続きはより直接的な解析を用いているため，このような欠点はない．しかし，我々の反証手続きは，文献[9][10]の手法と異なり，判定手続きではない．また，一般生成性を持たない場合であっても反証手続きが発散することがある．

†6 `http://infinity.few.vu.nl/productivity/`

表 2　組合せ子の書き換え規則に対する ω-強頭部正規化可能性の反証実験

番号	組合せ子	書き換え規則	結果	実行時間	ステップ数	構成された反例
1	S	$\mathsf{S}xyz \to xz(yz)$	成功	0	1	$\mathsf{S}(\mu x.\mathsf{S}x)yz$
2	K	$\mathsf{K}xy \to x$	成功	0	1	$\mathsf{K}(\mu x.\mathsf{K}xx)y$
3	I	$\mathsf{I}x \to x$	成功	0	1	$\mu x.\mathsf{I}x$
4	L	$\mathsf{L}xy \to x(yy)$	成功	0	1	$(\mu x.\mathsf{L}x)y$
5	J	$\mathsf{J}xyzw \to xy(xwz)$	成功	0	1	$\mathsf{J}(\mu x.\mathsf{J}xx)yzw$
6	H	$\mathsf{H}xyz \to xyzy$	成功	0	2	HHHH
7	M	$\mathsf{M}x \to xx$	成功	0	1	MM
8	W	$\mathsf{W}xy \to xyy$	成功	0	2	WWW
9	W^1	$\mathsf{W}^1xy \to yxx$	成功	0	2	$\mathsf{W}^1\mathsf{W}^1\mathsf{W}^1$
10	W^*	$\mathsf{W}^*xyz \to xyzz$	成功	2	2	$\mathsf{W}^*\mathsf{W}^*\mathsf{W}^*\mathsf{W}^*$
11	W^{**}	$\mathsf{W}^{**}xyzw \to xyzww$	成功	2	6	$\mathsf{W}^{**}\mathsf{W}^{**}\mathsf{W}^{**}\mathsf{W}^{**}\mathsf{W}^{**}$
12	B	$\mathsf{B}xyz \to x(yz)$	成功	0	1	$\mathsf{B}(\mu x.\mathsf{B}xx)yz$
13	C	$\mathsf{C}xyz \to xzy$	成功	0	1	$\mathsf{C}(\mu x.\mathsf{C}x)yz$
14	D	$\mathsf{D}xyzw \to xy(zw)$	成功	0	1	$\mathsf{D}(\mu x.\mathsf{D}xx)yzw$
15	E	$\mathsf{E}xyzwv \to xy(zwv)$	成功	1	1	$\mathsf{E}(\mu x.\mathsf{E}xxx)yzwv$
16	F	$\mathsf{F}xyz \to zyx$	成功	0	2	$\mathsf{F}(\mu x.\mathsf{F}x)y(\mathsf{F}(\mu x.\mathsf{F}x))$
17	G	$\mathsf{G}xyzw \to xw(yz)$	成功	0	1	$\mathsf{G}(\mu x.\mathsf{G}xx)yzw$
18	Q	$\mathsf{Q}xyz \to y(xz)$	成功	0	1	$\mathsf{Q}x(\mu y.\mathsf{Q}yy)z$
19	Q_1	$\mathsf{Q}_1xyz \to x(zy)$	成功	0	1	$\mathsf{Q}_1(\mu y.\mathsf{Q}_1xx)yz$
20	Q_3	$\mathsf{Q}_3xyz \to z(xy)$	成功	0	2	$\mathsf{Q}_3(\mu x.\mathsf{Q}_3x)y((\mu x.\mathsf{Q}_3x)y)$
21	R	$\mathsf{R}xyz \to yzx$	成功	0	2	$(\mu x.\mathsf{R}x)(\mu x.\mathsf{R}x)(\mu x.\mathsf{R}x)$
22	T	$\mathsf{T}xy \to yx$	成功	0	2	$(\mu x.\mathsf{T}x)(\mu x.\mathsf{T}x)$
23	V	$\mathsf{V}xyz \to zxy$	成功	0	2	$(\mu x.\mathsf{V}x)(\mu x.\mathsf{V}x)(\mu x.\mathsf{V}x)$
24	C^*	$\mathsf{C}^*xyzw \to xywz$	成功	0	1	$\mathsf{C}^*(\mu x.\mathsf{C}^*x)yzw$
25	C^{**}	$\mathsf{C}^{**}xyzwv \to xyzvw$	成功	0	1	$(\mu x.\mathsf{C}^{**}x)yzwv$
26	F^* ([12])	$\mathsf{F}^*xyzkf \to x(y(k(fz)))$	成功	0	1	$(\mu x.\mathsf{F}^*x)yzkf$
27	Y	$\mathsf{Y}x \to x(\mathsf{Y}x)$	発散	53	20	—
28	U	$\mathsf{U}xy \to y(xxy)$	発散	7	20	—
29	U^2	$\mathsf{U}^2xy \to y(xxxy)$	発散	9	20	—
30	O	$\mathsf{O}xy \to y(xy)$	発散	5	20	—

7 おわりに

本論文では，無限項書き換えシステムにおける ω-強頭部正規化可能性およびストリーム項書き換えシステムにおける一般生成性の反証手続きを与えるとともに，これらの手続きの正しさ (健全性) を証明した．また，この手続きに必要となる正則項の単一化手続きが，Jaffer [20] の手続きを利用して効率良く実現できることを示した．また，提案手法を実装し，関連文献から抜粋した例を用いて反証実験を行った．

ω-強頭部正規化可能性の反証法は，簡単な十分条件を除いて従来知られていなかったが，我々の手続きにもとづいて，その十分条件を満たさない例についても自動反証が可能となる場合があることを確認した．特に，文献 [19] において非 ω-強頭部正規化可能性が報告されている組合せ子の書き換えシステム 25 例に対して提案手続きを適用した結果，全 25 例に対して提案手法が有効であることを確認した．また，ω-強頭部正規化可能性が従来検証されていなかった組合せ子 F^* ([12]) についても ω-強頭部正規化可能性の自動反証に成功した．

従来の生成性判定法 [9] [10] は，ストリーム項書き換えシステム一般には適用可能でない．生成性に関する文献 [3] [10] [15] [18] [28] [34] より抜粋した例について実験を行い，従来法では取り扱えない例について効率的に反証に成功する場合があることを確認した．

無限項上のナローイングの有限表現の考案や，セミユニフィケーション [21] の正則項への拡張により，ω-強頭部正規化可能性の自動反証法の改良を試みたり，文献 [9] [10] の判定法のアイデアと我々の反証法のアイデアを融合して，より強力な一般生成性証明法や反証法を実現することは今後の課題である．

表 3 一般生成性の反証実験に用いたストリーム項書き換えシステム

$\mathcal{R}_1$ ([15], p.410) $\left\{ \begin{array}{l} \mathsf{ones} \to \mathsf{cons}(1, \mathsf{ones}) \end{array} \right\}$	$\mathcal{R}_2$ ([15], p.411) $\left\{ \begin{array}{l} \mathsf{ones2} \to \mathsf{cons}(1, \mathsf{tl}(\mathsf{ones2})) \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$
$\mathcal{R}_3$ $\left\{ \begin{array}{l} \mathsf{zero}(\mathsf{cons}(0, xs)) \to \mathsf{cons}(0, \mathsf{zero}(xs)) \\ \mathsf{zero}(\mathsf{cons}(1, xs)) \to \mathsf{zero}(xs) \end{array} \right\}$	$\mathcal{R}_4$ ([18], p.7) $\left\{ \begin{array}{l} \mathsf{f}(\mathsf{cons}(\bullet, \mathsf{cons}(\bullet, xs))) \to \mathsf{cons}(\bullet, \mathsf{f}(\mathsf{f}(xs))) \end{array} \right\}$
$\mathcal{R}_5$ ([10], p.769) $\left\{ \begin{array}{l} \mathsf{alt} \to \mathsf{tl}(\mathsf{alt2}) \\ \mathsf{alt2} \to \mathsf{cons}(0, \mathsf{cons}(1, \mathsf{alt2})) \end{array} \right\}$	$\mathcal{R}_6$ ([10], p.769) $\left\{ \begin{array}{l} \mathsf{zero} \to \mathsf{f}(\mathsf{c}),\ \mathsf{c} \to \mathsf{c} \\ \mathsf{f}(xs) \to \mathsf{cons}(0, \mathsf{f}(xs)) \end{array} \right\}$
$\mathcal{R}_7$ ([3], p.86) $\left\{ \begin{array}{l} \mathsf{F}_2(\mathsf{cons}(x, xs)) \\ \qquad\qquad \to \mathsf{cons}(\mathsf{plus}(\mathsf{s}(0), x), \mathsf{F}_2(\mathsf{F}_2(\mathsf{tl}(\mathsf{tl}(xs))))) \\ \mathsf{plus}(0, y) \to y \\ \mathsf{plus}(\mathsf{s}(x), y) \to \mathsf{s}(\mathsf{plus}(x, y)) \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$	$\mathcal{R}_8$ ([10], p.771) $\left\{ \begin{array}{l} \mathsf{J} \to \mathsf{cons}(0, \mathsf{cons}(1, \mathsf{even}(\mathsf{J}))) \\ \mathsf{even}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{odd}(xs)) \\ \mathsf{odd}(\mathsf{cons}(x, xs)) \to \mathsf{even}(xs) \end{array} \right\}$
$\mathcal{R}_9$ ([34], p.165) $\left\{ \begin{array}{l} \mathsf{f}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{tl}(\mathsf{f}(xs))) \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$	$\mathcal{R}_{10}$ ([34], p.171) $\left\{ \begin{array}{l} \mathsf{c} \to \mathsf{cons}(1, \mathsf{c}),\ \mathsf{f}(\mathsf{cons}(x, xs)) \to \mathsf{g}(x, xs) \\ \mathsf{g}(0, xs) \to \mathsf{f}(xs) \\ \mathsf{g}(1, xs) \to \mathsf{cons}(1, \mathsf{f}(xs)) \end{array} \right\}$
$\mathcal{R}_{11}$ ([34], p.175) $\left\{ \begin{array}{l} \mathsf{c} \to \mathsf{cons}(1, \mathsf{c}) \\ \mathsf{f}(\mathsf{cons}(x, xs)) \to \mathsf{g}(\mathsf{f}(xs)) \\ \mathsf{g}(\mathsf{cons}(x, xs)) \to \mathsf{c} \end{array} \right\}$	$\mathcal{R}_{12}$ ([34], p.177) $\left\{ \begin{array}{l} \mathsf{c} \to \mathsf{cons}(1, \mathsf{c}) \\ \mathsf{f}(\mathsf{cons}(0, xs)) \to \mathsf{cons}(1, \mathsf{f}(xs)) \\ \mathsf{f}(\mathsf{cons}(1, xs)) \to \mathsf{cons}(0, \mathsf{f}(\mathsf{f}(xs))) \end{array} \right\}$
$\mathcal{R}_{13}$ ([10], p.767) $\left\{ \begin{array}{l} \mathsf{M} \to \mathsf{cons}(0, \mathsf{zip}(\mathsf{inv}(\mathsf{even}(\mathsf{M})), \mathsf{tail}(\mathsf{M}))) \\ \mathsf{zip}(\mathsf{cons}(x, xs), ys) \to \mathsf{cons}(x, \mathsf{zip}(ys, xs)) \\ \mathsf{inv}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(\mathsf{i}(x), \mathsf{inv}(xs)) \\ \mathsf{even}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{odd}(xs)) \\ \mathsf{odd}(\mathsf{cons}(x, xs)) \to \mathsf{even}(xs) \\ \mathsf{tail}(\mathsf{cons}(x, xs)) \to xs \\ \mathsf{i}(0) \to 1,\ \mathsf{i}(1) \to 0 \end{array} \right\}$	$\mathcal{R}_{14}$ ([10], p.770) $\left\{ \begin{array}{l} \mathsf{M} \to \mathsf{cons}(0, \mathsf{zip}^*(\mathsf{inv}(\mathsf{even}(\mathsf{M})), \mathsf{tail}(\mathsf{M}))) \\ \mathsf{zip}^*(\mathsf{cons}(x, xs), \mathsf{cons}(y, ys)) \\ \qquad\qquad \to \mathsf{cons}(x, \mathsf{cons}(y, \mathsf{zip}^*(xs, ys))) \\ \mathsf{inv}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(\mathsf{i}(x), \mathsf{inv}(xs)) \\ \mathsf{even}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(x, \mathsf{odd}(xs)) \\ \mathsf{odd}(\mathsf{cons}(x, xs)) \to \mathsf{even}(xs) \\ \mathsf{tail}(\mathsf{cons}(x, xs)) \to xs \\ \mathsf{i}(0) \to 1,\ \mathsf{i}(1) \to 0 \end{array} \right\}$
$\mathcal{R}_{15}$ ([10], p.770) $\left\{ \begin{array}{l} \mathsf{M} \to \mathsf{cons}(0, \mathsf{zip}(\mathsf{inv}(\mathsf{M}), \mathsf{tail}(\mathsf{M}))) \\ \mathsf{zip}(\mathsf{cons}(x, xs), ys) \to \mathsf{cons}(x, \mathsf{zip}(ys, xs)) \\ \mathsf{inv}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(\mathsf{i}(x), \mathsf{inv}(xs)) \\ \mathsf{tail}(\mathsf{cons}(x, xs)) \to xs \\ \mathsf{i}(0) \to 1,\ \mathsf{i}(1) \to 0 \end{array} \right\}$	$\mathcal{R}_{16}$ ([10], p.770) $\left\{ \begin{array}{l} \mathsf{M} \to \mathsf{cons}(0, \mathsf{zip}^*(\mathsf{inv}(\mathsf{M}), \mathsf{tail}(\mathsf{M}))) \\ \mathsf{zip}^*(\mathsf{cons}(x, xs), \mathsf{cons}(y, ys)) \\ \qquad\qquad \to \mathsf{cons}(x, \mathsf{cons}(y, \mathsf{zip}^*(xs, ys))) \\ \mathsf{inv}(\mathsf{cons}(x, xs)) \to \mathsf{cons}(\mathsf{i}(x), \mathsf{inv}(xs)) \\ \mathsf{tail}(\mathsf{cons}(x, xs)) \to xs \\ \mathsf{i}(0) \to 1,\ \mathsf{i}(1) \to 0 \end{array} \right\}$
$\mathcal{R}_{17}$ ([28], p.634) $\left\{ \begin{array}{l} \mathsf{L} \to \mathsf{cons}(0, \mathsf{cons}(\mathsf{s}(0), \mathsf{plusl}(\mathsf{L}, \mathsf{tl}(\mathsf{L})))) \\ \mathsf{plusl}(\mathsf{cons}(x, xs), \mathsf{cons}(y, ys)) \\ \qquad\qquad \to \mathsf{cons}(\mathsf{plus}(x, y), \mathsf{plusl}(xs, ys)) \\ \mathsf{plus}(0, y) \to y \\ \mathsf{plus}(\mathsf{s}(x), y) \to \mathsf{s}(\mathsf{plus}(x, y)) \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$	$\mathcal{R}_{18}$ ([28], p.634) $\left\{ \begin{array}{l} \mathsf{L} \to \mathsf{cons}(0, \mathsf{cons}(\mathsf{s}(0), \mathsf{minl}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L}))), \mathsf{tl}(\mathsf{L})))) \\ \mathsf{minl}(\mathsf{cons}(x, xs), \mathsf{cons}(y, ys)) \\ \qquad\qquad \to \mathsf{cons}(\mathsf{minus}(x, y), \mathsf{minl}(xs, ys)) \\ \mathsf{minus}(x, 0) \to x \\ \mathsf{minus}(0, \mathsf{s}(y)) \to 0 \\ \mathsf{minus}(\mathsf{s}(x), \mathsf{s}(y)) \to \mathsf{minus}(x, y) \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$
$\mathcal{R}_{19}$ ([28], p.642) $\left\{ \begin{array}{l} \mathsf{L} \to \mathsf{cons}(\mathsf{s}(0), \mathsf{cons}(\mathsf{hd}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L}))))), \mathsf{L})) \\ \mathsf{hd}(\mathsf{cons}(x, xs)) \to x \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$	$\mathcal{R}_{20}$ ([28], p.642) $\left\{ \begin{array}{l} \mathsf{L} \to \mathsf{cons}(\mathsf{s}(0), \mathsf{cons}(\mathsf{hd}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L})))), \mathsf{L})) \\ \mathsf{hd}(\mathsf{cons}(x, xs)) \to x \\ \mathsf{tl}(\mathsf{cons}(x, xs)) \to xs \end{array} \right\}$

表 4　一般生成性の反証実験

番号	ストリーム TRS	対象項	一般生成性	結果	実行時間	ステップ数	構成された反例	[9] [10]
1	$\mathcal{R}_1$	ones	有	発散	(0)	(20)	—	成功
2	$\mathcal{R}_2$	ones2	無	成功	0	3	$\mathsf{tl}(\mathsf{ones2})$	成功
3	$\mathcal{R}_3$	$\mathsf{zero}(xs)$	無	成功	0	2	$\mathsf{zero}(\mu x.\ \mathsf{cons}(1,x))$	失敗
4	$\mathcal{R}_4$	$\mathsf{f}(xs)$	無	成功	0	2	$\mathsf{f}(\mathsf{f}(\mu x.\ \mathsf{cons}(\bullet,x)))$	失敗
5	$\mathcal{R}_5$	alt	有	発散	(0)	(20)	—	成功
6	$\mathcal{R}_6$	zeros	有	発散	(0)	(20)	—	失敗
7	$\mathcal{R}_7$	$\mathsf{F}_2(xs)$	無	成功	0	2	$\mathsf{F}_2(\mathsf{F}_2(\mathsf{tl}(\mathsf{tl}(\mathsf{cons}(x,\mu ys.\ \mathsf{cons}(y,ys))))))$	失敗
8	$\mathcal{R}_8$	J	無	成功	1	12	$\mathsf{odd}(\mathsf{odd}(\mathsf{cons}(1,\mathsf{even}(\mathsf{J}))))$	成功
9	$\mathcal{R}_9$	$\mathsf{f}(xs)$	無	成功	0	3	$\mathsf{tl}(\mathsf{f}(\mathsf{cons}(x,\mu ys.\ \mathsf{cons}(y,ys))))$	失敗
10	$\mathcal{R}_{10}$	$\mathsf{f}(xs)$	無	成功	0	2	$\mathsf{f}(\mu xs.\ \mathsf{cons}(0,xs))$	失敗
11	$\mathcal{R}_{11}$	$\mathsf{f}(\mathsf{c})$	無	成功	0	2	$\mathsf{f}(\mathsf{c})$	失敗
12	$\mathcal{R}_{12}$	$\mathsf{f}(\mathsf{c})$	有	発散	(2)	(20)	—	成功
13	$\mathcal{R}_{13}$	M	有	発散	(5)	(20)	—	成功
14	$\mathcal{R}_{14}$	M	無	成功	2	7	$\mathsf{zip}^*(\mathsf{inv}(\mathsf{even}(\mathsf{M})),\mathsf{tail}(\mathsf{M}))$	成功
15	$\mathcal{R}_{15}$	M	有	発散	(4)	(20)	—	成功
16	$\mathcal{R}_{16}$	M	無	成功	1	7	$\mathsf{zip}^*(\mathsf{inv}(\mathsf{M}),\mathsf{tail}(\mathsf{M}))$	成功
17	$\mathcal{R}_{17}$	L	有	発散	(7)	(20)	—	成功
18	$\mathcal{R}_{18}$	L	無	成功	2	8	$\mathsf{minl}(\mathsf{tl}(\mathsf{tl}(\mathsf{tl}(\mathsf{L}))),\mathsf{tl}(\mathsf{L}))$	成功
19	$\mathcal{R}_{19}$	L	有	発散	(0)	(20)	—	失敗
20	$\mathcal{R}_{20}$	L	無	発散	(0)	(20)	—	失敗

謝辞 本研究に対してアドバイスやコメントをいただきました外山芳人教授に感謝致します．また，本論文を改善するための貴重なコメントをいただきました査読者に感謝致します．本研究は日本学術振興会科学研究費 20500002, 21700017 の補助を一部受けて行われた．

参考文献

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

[2] Barendregt, H. P.: *The Lambda Calculus, Its Syntax and Semantics*, North-Holland, 2nd revised edition, 1984.

[3] Buchholz, W.: A term calculus for (co-)recursive definitions on streamlike data structures, *Ann. Pure Appl. Log.*, Vol. 136, No. 1-2(2005), pp. 75–90.

[4] Colmerauer, A.: Equations and inequations on finite and infinite trees, *Proc. of FGCS 1984*, 1984, pp. 85–99.

[5] Courcelle, B.: Fundamental properties of infinite trees, *Theor. Comput. Sci.*, Vol. 25(1983), pp. 95–169.

[6] Curry, H. B. and Feys, R.: *Combinatory Logic*, Vol. 1, North-Holland, 1958.

[7] Devlin, K.: *The Joy of Sets*, Springer-Verlag, 1993.

[8] Dijkstra, E. W.: On the productivity of recursive definitions, Personal note EWD 749, University of Texas at Austin, 1980.

[9] Endrullis, J., Grabmayer, C., and Hendriks, D.: Data-oblivious stream productivity, *Proc. of LPAR 2008*, LNCS, Vol. 5330, Springer-Verlag, 2008, pp. 79–96.

[10] Endrullis, J., Grabmayer, C., Hendriks, D., Ishihara, A., and Klop, J. W.: Productivity of stream definitions, *Theor. Comput. Sci.*, Vol. 411(2010), pp. 765–782.

[11] Endrullis, J., Grabmayer, C., Hendriks, D., Klop, J. W., and de Vrijer, R.: Proving infinitary normalization, *Proc. of TYPES 2008*, LNCS, Vol. 5497, Springer-Verlag, 2009, pp. 64–82.

[12] Fujita, K.-E.: An interpretation of $\lambda\mu$-calculus in λ-calculus, *Inf. Process. Lett.*, Vol. 84, No. 5(2002), pp. 261–264.

[13] Huet, G.: *Résolution d'équations dans les languages d'ordre* $1, 2, \ldots, \omega$, PhD Thesis, University Paris-7, 1976.

[14] Huet, G. and Oppen, D. C.: Equations and rewrite rules: A survey, Technical report, Stanford University, Stanford, CA, USA, 1980.

[15] Hughes, J., Pareto, L., and Sabry, A.: Proving the correctness of reactive systems using sized types, *Proc. of POPL 1996*, ACM, 1996, pp. 410–423.

[16] Inverardi, P. and Zilli, M. V.: Rational rewriting, *Proc. of MFCS 1994*, LNCS, Vol. 841, Springer-Verlag, 1994, pp. 433–442.

[17] Ishihara, A.: Productivity of algorithmic systems, *Proc. of SCSS 2008*, 2008, pp. 81–95.

[18] Ishihara, A.: *Algorithmic Term Rewriting Systems*, PhD Thesis, Vrije Universiteit Amsterdam,

2010.
[19] 岩見宗弘: 組合せ子の強収束性, *FIT2009 第 1 分冊*, 2009, pp. 251–258. A-011.
[20] Jaffar, J.: Efficient unification over infinite trees, *New Gen. Comput.*, Vol. 2(1984), pp. 207–219.
[21] Kapur, D., Musser, D., Narendran, P., and Stillman, J.: Semi-unification, *Theor. Comput. Sci.*, Vol. 81(1991), pp. 169–187.
[22] Kennaway, J. R., Klop, J. W., Sleep, M. R., and de Vries, F. J.: On the adequacy of graph rewriting for simulating term rewriting, *ACM Trans. Program. Lang. Syst.*, Vol. 16, No. 3(1994), pp. 493–523.
[23] Kennaway, J. R., Klop, J. W., Sleep, M. R., and de Vries, F. J.: Transfinite reductions in orthogonal term rewriting systems, *Inf. Comput.*, Vol. 119(1995), pp. 18–38.
[24] Klop, J. W. and de Vrijer, R.: Infinitary normalization, *We Will Show Them! Essays in Honour of D. Gabbay*, Vol. 2, College Publications, 2005, pp. 169–192.
[25] Martelli, A. and Montanari, U.: An efficient unification algorithm, *ACM Trans. Program. Lang. Syst.*, Vol. 4, No. 2(1982), pp. 239–257.
[26] Mukai, K.: A unification algorithm for infinite trees, *Proc. of IJCAI 1983*, 1983, pp. 547–549.
[27] Paulson, L. C.: *ML for the Working Programmer*, Cambridge University Press, 2nd edition, 1996.
[28] Sijtsma, B. A.: On the productivity of recursive list definitions, *ACM Trans. Program. Lang. Syst.*, Vol. 11, No. 4(1989), pp. 633–649.
[29] Smullyan, R.: *To Mock a Mockingbird*, Knopf, 1985.
[30] Telford, A. and Turner, D.: Ensuring the productivity of infinite structure, Technical Report 14-97, University of Kent at Canterbury, 1998.
[31] Terese: *Term Rewriting Systems*, Cambridge University Press, 2003.
[32] Thompson, S.: *Haskell The Craft of Functional Programming*, Addison Wesley, 2nd edition, 1999.
[33] Zantema, H.: Normalization of infinite terms, *Proc. of RTA 2008*, LNCS, Vol. 5117, Springer-Verlag, 2008, pp. 441–455.
[34] Zantema, H.: Well-definedness of streams by termination, *Proc. of RTA 2009*, LNCS, Vol. 5595, Springer-Verlag, 2009, pp. 164–178.

A 補題 32 および定理 35 の証明

まず，経路的ストリーム非照合性をもちいて，非照合性の十分条件を与える．

補題 38 (非照合性の十分条件) 項 t は $p \in \mathrm{Pos}(t)$ に関して項 l に経路的にストリーム非照合であるとする．このとき，$t(p) \neq \mathsf{cons}$ であるならば，任意の代入 σ について，l は $t\sigma$ と照合不能である．

(証明) 代入 η に対して $l\eta = t\sigma$ とする．このとき，$p \in \mathrm{Pos}(l)$ とすると，$p \in \mathrm{Pos}(t)$ より，$\forall q < p.\ l(q) = l\eta(q) = t\sigma(q) = t(q)$．よって，ストリーム非照合の仮定より，$l(p) = \mathsf{cons}$．すると，$l(p) = l\eta(p) = t\sigma(p) = t(p)$ であるから，$t(p) \neq \mathsf{cons}$ に矛盾．一方，$p \notin \mathrm{Pos}(l)$ とすると，$l\eta = t\sigma$ より，ある $q < p$ が存在して，$l(q) \in \mathcal{V}$．また，$\forall q' < q.\ l(q') = t\sigma(q')$．このとき，$q' < p \in \mathrm{Pos}(t)$ より，$\forall q' < q.\ l(q') = t(q')$．これは，項 t が p に関して項 l に経路的に照合するということになり，仮定に矛盾． □

次に，安定位置の性質をいくつか示す．

補題 39 (安定位置の性質) $p \in \mathrm{Pos}(s) \cap \mathrm{Pos}(t)$ とし，任意の位置 $q < p$ について $s(q) = t(q)$ とする．このとき，位置 p が項 s の安定位置ならば，位置 p は項 t の安定位置である．

(証明) 仮定より，任意の書き換え規則 $l \to r \in \mathcal{R}$ と任意の位置 $q < p$ について，$s|_q$ は $p\backslash q$ に関して l に経路的にストリーム照合しない．よって，(1) $\forall p' \leq p\backslash q.\ [l(p') \in \mathcal{V} \Rightarrow \exists q' < p'.\ s|_q(q') \neq l(q')]$，かつ，(2) $(\forall q' < p\backslash q.\ s|_q(q') = l(q')) \Rightarrow l(p\backslash q) = \mathsf{cons}$ が成立する．このとき，任意の位置 $q' < p\backslash q$ について，$q.q' < p$ であるから，仮定より $s|_q(q') = s(q.q') = t(q.q') = t|_q(q')$ が成立する．よって，(1) より，(1′) $\forall p' \leq p\backslash q.\ [l(p') \in \mathcal{V} \Rightarrow \exists q' < p'.\ t|_q(q') \neq l(q')]$，(2) より，(2′) $(\forall q' < p\backslash q.\ t|_q(q') = l(q')) \Rightarrow l(p\backslash q) = \mathsf{cons}$ が成立する．よって，任意の書き換え規則 $l \to r \in \mathcal{R}$ と任意の位置 $q < p$ について，$t|_q$ は $p\backslash q$ に関して l に経路的にストリーム照合しない．従って，位置 p は項 t の安定位置である． □

補題 40 (安定位置の安定性) 位置 $p \in \mathrm{Pos}(t)$ を項 t の安定位置とし，$t(p) \neq \mathsf{cons}$ とする．このとき，任意の代入 σ と位置 $q < p$ について，項 $t|_q\sigma$ はリデックスでない．

(証明) 安定位置の定義から，任意の書き換え規則 $l \to r \in \mathcal{R}$ について，$t|_q$ は $p\backslash q$ に関して l に経路的にストリーム照合しない．また，$t(p) = t|_q(p\backslash q)$ であるから，$t|_q(p\backslash q) \neq \mathsf{cons}$．一方，項 $t|_q\sigma$ がリデックスとすると，ある $l \to r \in \mathcal{R}$ が存在して，l が $t|_q\sigma$

と照合する．これは補題 38 に矛盾する． □

補題 41 (安定位置の合成) 位置 $p \in \mathrm{Pos}(t)$ を項 t の安定位置，位置 $q \in \mathrm{Pos}(s)$ が項 s の安定位置，$s(\epsilon) \neq \mathsf{cons}$ ならば，位置 $p.q$ は項 $t[s]_p$ の安定位置．

(証明) $l \to r \in \mathcal{R}$，$o < p.q$ として，位置 $(p.q)\backslash o$ に関して項 $t[s]_p|_o$ が l に経路的にストリーム照合すると仮定して矛盾を導く．$o \geq p$ のとき，$t[s]_p|_o = s|_{o\backslash p}$ かつ $(p.q)\backslash o = q\backslash(o\backslash p)$ より，位置 $q\backslash(o\backslash p)$ に関して項 $s|_{o\backslash p}$ が l に経路的にストリーム照合する．これは q が項 s の安定位置であることに矛盾する．よって，$o < p$ のときを考えればよい．このとき，定義 27, 28, 29 より，<u>(a)</u> $\exists p' \leq (p.q)\backslash o.\ [l(p') \in \mathcal{V} \wedge \forall q' < p'.\ t[s]_p|_o(q') = l(q')]$ または <u>(b)</u> $(\forall p' < (p.q)\backslash o.\ t[s]_p|_o(p') = l(p')) \wedge l((p.q)\backslash o) \neq \mathsf{cons}$ が成立するから，次の 2 つの場合に分けて考える．

<u>(a) の場合</u>．$p'(\leq (p.q)\backslash o)$ の位置により次の 3 通りに場合分けする．

- $p' < p\backslash o$ の場合．このとき，任意の位置 $q' < p\backslash o(\leq (p.q)\backslash o)$ について，$t|_o(q') = t[s]_p|_o(q')$ が成立するから，$\exists p' \leq p\backslash o.\ [l(p') \in \mathcal{V} \wedge \forall q' < p'.\ t|_o(q') = t[s]_p|_o(q') = l(q')]$ が成立．これは位置 p が項 t の安定位置であることに矛盾．
- $p' = p\backslash o$ の場合．このとき，$p\backslash o \leq (p.q)\backslash o$ より，$\forall q' < p\backslash o.\ t[s]_p|_o(q') = l(q')$ が成立する．ここで，任意の位置 $q' < p\backslash o$ について，$t|_o(q') = t[s]_p|_o(q')$ が成立するから，$\forall q' < p\backslash o.\ t|_o(q') = t[s]_p|_o(q') = l(q')$．従って，位置 p が項 t の安定位置であることから，$l(p\backslash o) = \mathsf{cons}$．これは，$l(p\backslash o) = l(p') \in \mathcal{V}$ に矛盾．
- $p' > p\backslash o$ の場合．このとき，$p' \leq (p.q)\backslash o$ より，$\forall q' < p'.\ t[s]_p|_o(q') = l(q')$ が成立するから，特に，$\forall q' < p\backslash o.\ t[s]_p|_o(q') = l(q')$ および $s(\epsilon) = t[s]_p|_o(p\backslash o) = l(p\backslash o)$ が成立する．ここで，任意の位置 $q' < p\backslash o$ について，$t|_o(q') = t[s]_p|_o(q')$ が成立するから，前者より $\forall q' < p\backslash o.\ t|_o(q') = l(q')$．よって，位置 p が項 t の安定位置であることから，$l(p\backslash o) = \mathsf{cons}$，すなわち，$s(\epsilon) = l(p\backslash o) = \mathsf{cons}$．これは仮定 $s(\epsilon) \neq \mathsf{cons}$ に矛盾．

<u>(b) の場合</u>．このとき，$p\backslash o \leq (p.q)\backslash o$ より，$\forall p' < p\backslash o.\ t|_o(p') = t[s]_p|_o(p') = l(p')$ が成立する．よって，位置 p が項 t の安定位置であることから，$l(p\backslash o) = \mathsf{cons}$．項 s の安定位置 q に関して，次の 2 つの場合に分けて考える．

- $q = \epsilon$ の場合．$l(p\backslash o) = l((p.q)\backslash o) \neq \mathsf{cons}$ に矛盾．
- $q \neq \epsilon$ の場合．$p\backslash o < (p.q)\backslash o$ より，$t[s]_p|_o(p\backslash o) = l(p\backslash o)$．よって，$s(\epsilon) = t[s]_p|_o(p\backslash o) = \mathsf{cons}$ となり，仮定 $s(\epsilon) \neq \mathsf{cons}$ に矛盾． □

特に，位置 $p \in \mathrm{Pos}(t)$ が項 t の安定位置かつ $s(\epsilon) \neq \mathsf{cons}$ ならば位置 p は項 $t[s]_p$ の安定位置である．書き換え列 $s \to_{q_0} s_1 \to_{q_1} \cdots \to_{q_n} t$ であるとき，$\Gamma = \{q_0, \ldots, q_n\}$ を用いて $s \to^*_\Gamma t$ と記す．

補題 42 (安定位置の保存) 位置 $p \in \mathrm{Pos}(s)$ を項 s の安定位置かつ $s(p) \neq \mathsf{cons}$ とする．

(1) $s \to_{q,\mathcal{R}} t$ かつ $q \not\geq p$ とするとき，$q \not< p$，$p \in \mathrm{Pos}(t)$，$s|_p = t|_p$，$t(p) \neq \mathsf{cons}$，かつ，位置 p は項 t の安定位置である．

(2) 書き換え列 $s \to^*_\Gamma t$ について，$\forall q \in \Gamma.\ q \not\geq p$ ならば $p \in \mathrm{Pos}(t)$，$s|_p = t|_p$，かつ，位置 p は項 t の安定位置である．

(証明) (1) 仮定および補題 40 より，任意の位置 $q < p$ について，項 $s|_q$ はリデックスでない．よって，$q \not< p$．このとき，$q \not\geq p$ より位置 p と q は並列な位置関係にあるから，$s \to_{q,\mathcal{R}} t$ より $t(p) \neq \mathsf{cons}$ かつ $s|_p = t|_p$ が成立し，また $\forall q' < p.\ s(q') = t(q')$ が成立する．後者から，補題 39 より位置 p は項 t の安定位置であることが導かれる．(2) (1) よりあきらか． □

$p_1, \ldots, p_n \in \mathrm{Pos}(s)$ が項 s の最外リデックスの位置であり，$s \xrightarrow{\circ}_{p_1} s_1 \xrightarrow{\circ}_{p_2} \cdots \xrightarrow{\circ}_{p_n} t$ となるとき，$\Delta = \{p_1, \ldots, p_n\}$ を用いて，$s \xrightarrow{\circ}{}^*_\Delta t$ と記す．リデックスの最外性より，位置 $p_1, \ldots, p_n$ での書き換えはその順番に依存しないことに注意する．有限または無限書き換え列 $s_0 \xrightarrow{\circ}_{p_0} s_1 \xrightarrow{\circ}_{p_1} s_2 \xrightarrow{\circ}_{p_2} \cdots$ は，どの項 s_i の最外リデックスもいずれは簡約されるか最外リデックスではなくなるとき，つまり，任意の位置 $p \in \mathrm{Pos}(s_i)$ について $s_i|_p$ が最外リデックスならばある $j \geq i$ が存在して，$p = p_j$ であるか，もしくは $s_j|_p$ は s_j の最外リデックスでない，が成立するとき，こ

れを**最外公平書き換え**とよぶ．次の性質は，ストリーム型項 t が生成的であるかの判定の基礎となる．

命題 43 (**ω-正規形の計算** [10]) $\mathcal{R}$ をストリーム項書き換えシステム，t をストリーム型項とする．このとき，t が ω-正規形 s をもつならば，t から s へは最外公平書き換えにより到達可能である． □

補題 44 (**最外書き換えによる生成性の保存**) $\mathcal{R}$ をストリーム項書き換えシステム，t をストリーム型項，$t \overset{\circ}{\to}^* h_1 :: \cdots :: h_n :: s$ とする．このとき，s が生成的でないならば，t も生成的でない．

(証明) s が生成的でないとする．s が ω-正規形をもたないとき，t も ω-正規形をもたない．s が ω-正規形をもつときは，t が ω-正規形をもつならば，ω-正規形の一意性から t と $h_1 :: \cdots :: h_n :: s$ は同一の ω-正規形をもつ．よって，s の ω-正規形が構成子項でなければ，t の ω-正規形も構成子項でない．よって，t も生成的でない． □

最外ナローイングステップの次の性質を用いる．

補題 45 (**最外ナローイングステップの性質**) $s \overset{\circ}{\rightsquigarrow}^*_\sigma t$ となるとき，任意の代入 η について，$s\sigma\eta \overset{\circ}{\to}^* t\eta$ となる．また，$s \overset{\circ}{\rightsquigarrow}^*_\sigma t$ が根ナローイングステップを含むならば，任意の代入 η について，$s\sigma\eta \overset{\circ}{\to}^* t\eta$ は根書き換えステップを含む．

(証明) まず，任意の代入 η について，

$$s \overset{\circ}{\rightsquigarrow}_{p,\theta} t \text{ ならば } s\theta\eta \overset{\circ}{\to}_p t\eta \tag{2}$$

を示す．$s \overset{\circ}{\rightsquigarrow}_{p,\theta} t$ とする．このとき，書き換え規則 $l \to r \in \mathcal{R}$ が存在して，$\theta = \mathrm{mgu}_{inf}(s|_p, l)$ かつ $t = s[r]_p\theta$ ($s|_p \notin \mathcal{V}$)．よって，$s\theta\eta = s[s|_p]_p\theta\eta = s\theta[s|_p\theta]_p\eta = s\theta[l\theta]_p\eta \to_p s\theta[r\theta]_p\eta = s[r]_p\theta\eta = t\eta$．また，$s\theta[l\theta]_p\eta \to_p s\theta[r\theta]_p\eta$ が最外書き換えステップでないとすると，ある位置 $q < p$ と項 t' が存在して，$s\theta\eta \to_q t'$ となる．このとき，ある書き換え規則 $l' \to r' \in \mathcal{R}$ と代入 σ が存在して，$s\theta\eta|_q = s\theta|_q\eta = s|_q\theta\eta = l'\sigma$．また，$s|_p \notin \mathcal{V}$ かつ $q < p$ より $s|_q \notin \mathcal{V}$．ここで，$\mathcal{V}(s|_q) \cap \mathcal{V}(l') = \emptyset$ としてよいから，項 $s|_q$ と l' は単一化可能．よって，$\theta' = \mathrm{mgu}_{inf}(s|_q, l')$ とすると，$s \rightsquigarrow_{q,\theta'} s[r']_q\theta'$ となる．これは，$s \rightsquigarrow_{p,\theta} t$ が最外ナローイングステップであることに矛盾する．

次に，$s_1 \overset{\circ}{\rightsquigarrow}_{p_1,\theta_1} s_2 \overset{\circ}{\rightsquigarrow}_{p_2,\theta_2} \cdots \overset{\circ}{\rightsquigarrow}_{p_{n-1},\theta_{n-1}} s_n$，$\sigma_i = \theta_{n-1} \circ \cdots \circ \theta_i$ ($1 \leq i \leq n-1$) のとき，任意の代入 η について，$s_1\sigma_1\eta \overset{\circ}{\to}_{p_1} s_2\sigma_2\eta \overset{\circ}{\to}_{p_2} \cdots \overset{\circ}{\to}_{p_{n-2}} s_{n-1}\sigma_{n-1}\eta \overset{\circ}{\to}_{p_{n-1}} s_n\eta$ となることを n に関する帰納法で示す．$n = 1$ の場合はあきらか．$n > 1$ の場合を考える．帰納法の仮定から，$\xi_i = \theta_{n-2} \circ \cdots \circ \theta_i$ ($1 \leq i \leq n-2$) とおくと，任意の代入 η' に対して，$s_1\xi_1\eta' \overset{\circ}{\to}_{p_1} s_2\xi_2\eta' \overset{\circ}{\to}_{p_2} \cdots \overset{\circ}{\to}_{p_{n-3}} s_{n-2}\xi_{n-2}\eta' = s_{n-2}\theta_{n-2}\eta' \overset{\circ}{\to}_{p_{n-2}} s_{n-1}\eta'$．また，$s_{n-1} \overset{\circ}{\rightsquigarrow}_{p_{n-1},\theta_{n-1}} s_n$ と上記の性質 (2) から，任意の代入 η に対して，$s_{n-1}\theta_{n-1}\eta \overset{\circ}{\to}_{p_{n-1}} s_n\eta$．よって，$\eta' = \eta \circ \theta_{n-1}$ とおくと，$s_1\xi_1\theta_{n-1}\eta \overset{\circ}{\to}_{p_1} s_2\xi_2\theta_{n-1}\eta \overset{\circ}{\to}_{p_2} \cdots \overset{\circ}{\to}_{p_{n-3}} s_{n-2}\xi_{n-2}\theta_{n-1}\eta = s_{n-2}\theta_{n-2}\theta_{n-1}\eta \overset{\circ}{\to}_{p_{n-2}} s_{n-1}\theta_{n-1}\eta \overset{\circ}{\to}_{p_{n-1}} s_n\eta$．ここで，$\sigma_i = \theta_{n-1} \circ \xi_i$ ($1 \leq i \leq n-2$) かつ $\sigma_{n-1} = \theta_{n-1}$ に注意すると，$s_1\sigma_1\eta \overset{\circ}{\to}_{p_1} s_2\sigma_2\eta \overset{\circ}{\to}_{p_2} \cdots \overset{\circ}{\to}_{p_{n-2}} s_{n-1}\sigma_{n-1}\eta \overset{\circ}{\to}_{p_{n-1}} s_n\eta$． □

補題 46 (**安定位置における最外書き換え**) 位置 $p \in \mathrm{Pos}(s)$ を項 $s[u]_p$ の安定位置とする．このとき，$u \overset{\circ}{\to}_{\mathcal{R}} v$ かつ $v(\epsilon) \neq \mathsf{cons}$ ならば，$s[u]_p \overset{\circ}{\to}_{\mathcal{R}} s[v]_p$ かつ位置 p は項 $s[v]_p$ の安定位置である．

(証明) cons は構成子なので，$v(\epsilon) \neq \mathsf{cons}$ より，$u(\epsilon) \neq \mathsf{cons}$．すなわち，$s[u]_p(p) \neq \mathsf{cons}$．補題 40 より，任意の位置 $q < p$ について，部分項 $s[u]_p|_q$ はリデックスでない．よって，$s[u]_p \overset{\circ}{\to}_{\mathcal{R}} s[v]_p$．位置 p が項 $s[v]_p$ の安定位置であることは，補題 39 よりあきらか． □

(補題 32 の証明) 位置 p を項 t における安定位置，任意の代入 η について，$s\eta \overset{\circ}{\to}^+_{\mathcal{R}} t\eta[s\theta\eta]_p$ が成立するとする．代入 θ^n を $\theta^0 = \emptyset$; $\theta^{i+1} = \theta^i \circ \theta$ により定義する．このとき，以下のような無限書き換え列を考える．

$$\begin{array}{llll}
s & \overset{\circ}{\to}^+ & t[s\theta]_p & \overset{\circ}{\to}^*_{\Delta_1} & t_1[s\theta]_p \\
 & \overset{\circ}{\to}^+ & t_1[t\theta[s\theta^2]_p]_p & \overset{\circ}{\to}^*_{\Delta_2} & t_2[s\theta^2]_{p^2} \\
 & \overset{\circ}{\to}^+ & t_2[t\theta^2[s\theta^3]_p]_{p^2} & \overset{\circ}{\to}^*_{\Delta_3} & t_3[s\theta^3]_{p^3} \\
 & \overset{\circ}{\to}^+ & t_3[t\theta^3[s\theta^4]_p]_{p^3} & \overset{\circ}{\to}^*_{\Delta_4} & \cdots
\end{array}$$

ただし，ここで，$i = 0, 1, \ldots$ について，

- $t_i[s\theta^i]_{p^i} \overset{\circ}{\to}^+ t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i}$ は仮定より ($\eta := \theta^i$ として) 得られる書き換え列，かつ，

- Δ_{i+1} は，$t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i}$ の位置 $\not\geq p^{i+1}$ にある最外リデックスの集合

とする．ただし，$t_0 = t$．この書き換え列が実際に存在することを示すために，すべての $i \geq 0$ について次の性質 (A)〜(D) が成立することを示す．

(A) 位置 p^i は項 $t_i[s\theta^i]_{p^i}$ の安定位置，

(B) $t_i[s\theta^i]_{p^i} \xrightarrow{\circ}{}^{+} t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i}$，

(C) 位置 p^{i+1} は項 $t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i}$ の安定位置，かつ，

(D) $t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i} \xrightarrow{\circ}{}^{*}_{\Delta_{i+1}} s_i$ (Δ_{i+1} は，$t_i[t\theta^i[s\theta^{i+1}]_p]_{p^i}$ の位置 $\not\geq p^{i+1}$ にある最外リデックスの集合とする) ならば，$s_i = t_{i+1}[s\theta^{i+1}]_{p^{i+1}}$ かつ位置 p^{i+1} は項 s_i の安定位置．

証明は i に関する帰納法で行う．

(基本段階) $i = 0$ とする．

性質 (A) $p^0 = \epsilon$ より自明．

性質 (B) 仮定 $s\eta \xrightarrow{\circ}{}^{+}_{\mathcal{R}} t\eta[s\theta\eta]_p$ ($\eta := \emptyset$) より，$s \xrightarrow{\circ}{}^{+} t[s\theta]_p$．

性質 (C) 位置 $p \in \mathrm{Pos}(t)$ が項 t の安定位置であることと仮定 $s(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$ より，補題 41 から位置 p は項 $t[s\theta]_p$ の安定位置．

性質 (D) $t[s\theta]_p \xrightarrow{\circ}{}^{*}_{\Delta_1} s_0$ とする．このとき，Δ_1 のリデックスの位置は $\not\geq p$ で，$t[s\theta]_p(p) = s(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$ であることから，補題 42 より，$s_0 = t_1[s\theta]_p$ とおけ，かつ，位置 p は項 s_0 の安定位置である．

(帰納段階) $i = k$ の場合について性質 (A)〜(D) が成立するものとする．$i = k + 1$ の場合について性質 (A)〜(D) が成立することを示す．

性質 (A) 無限書き換え列の構成より，$t_k[t\theta^k[s\theta^{k+1}]_p]_{p^k} \xrightarrow{\circ}{}^{*}_{\Delta_{k+1}} t_{k+1}[s\theta^{k+1}]_{p^{k+1}}$．よって，帰納法の仮定 (D) より，位置 p^{k+1} は項 $t_{k+1}[s\theta^{k+1}]_{p^{k+1}}$ の安定位置であることが成り立つ．

性質 (B) 仮定 $s\eta \xrightarrow{\circ}{}^{+}_{\mathcal{R}} t\eta[s\theta\eta]_p$ ($\eta := \theta^{k+1}$ とる) より $s\theta^{k+1} \xrightarrow{\circ}{}^{+} t\theta^{k+1}[s\theta^{k+2}]_p$．また，$t\theta^{k+1}[s\theta^{k+2}]_p(\epsilon) = t(\epsilon) \neq \mathsf{cons}$．さらに，性質 (A) より，位置 p^{k+1} は項 $t_{k+1}[s\theta^{k+1}]_{p^{k+1}}$ の安定位置．よって，補題 46 より，$t_{k+1}[s\theta^{k+1}]_{p^{k+1}} \xrightarrow{\circ}{}^{+} t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}$ かつ位置 p^{k+1} は項 $t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}$ の安定位置．

性質 (C) 仮定より，位置 p が項 t の安定位置．よって，補題 39 から，位置 p は項 $t\theta^{k+1}[s\theta^{k+2}]_p$ の安定位置．また，性質 (B) より，位置 p^{k+1} は項 $t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}$ の安定位置．よって，$t\theta^{k+1}[s\theta^{k+2}]_p(\epsilon) = t(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$ より，補題 41 から，位置 $p^{k+2} (= p^{k+1}.p)$ は項 $t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}$ の安定位置．

性質 (D) $t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}} \xrightarrow{\circ}{}^{*}_{\Delta_{k+2}} s_{k+1}$ とする．このとき，Δ_{k+2} に含まれるリデックスの位置は $\not\geq p^{k+2}$ とする．また，性質 (C) より位置 p^{k+2} は項 $t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}$ の安定位置．よって，$t_{k+1}[t\theta^{k+1}[s\theta^{k+2}]_p]_{p^{k+1}}(p^{k+2}) = s(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$ であることから，補題 42 より，$s_{k+1} = t_{k+2}[s\theta^{k+2}]_{p^{k+2}}$ とおけ，かつ，位置 p^{k+2} は項 s_{k+1} の安定位置．

以上で，性質 (A)〜(D) が成立することが示された．

各 Δ_i のとり方および $p \neq \epsilon$ であることから，この無限書き換え列は最外公平書き換え列である．よって，項 s が $\mathcal{R}$ において生成的であるとすると，この書き換え列の極限は項 s の ω-正規形である．一方，ある位置 $q < p$ について $t(q) \in \mathcal{D}$ より任意の $i \geq 0$ について $t_i(q) \in \mathcal{D}$．よって，この書き換え列の ω-正規形は構成子項ではない．したがって，項 s は $\mathcal{R}$ において生成的でない． □

(定理 35 の証明) `step` xs がよびだされるとき，任意の 4 つ組 $(x, s_0, s, \sigma) \in xs$ について，t_0 ($\overset{\circ}{\rightsquigarrow}{}^{+} \circ$ `tail`$^{+}$)* $s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} s$，$s_0\sigma(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$，および，$s(\epsilon) \neq \mathsf{cons}$ が成立することは容易に確認できる．また，$x = \mathtt{true}$ ならば $s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} s$ は根ナローイングを含む．ここで，cons が構成子であることに注意すると，代入 ρ と項 $h_0, \ldots, h_n$ が存在して，$t_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\rho} h_0 :: h_1 :: \cdots :: h_n :: s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} h_0 :: h_1 :: \cdots :: h_n :: s$ となることがわかる．よって，補題 45 より任意の代入 η について，$t_0\rho\sigma\eta \xrightarrow{\circ}{}^{+} h_0\sigma\eta :: h_1\sigma\eta :: \cdots :: h_n\sigma\eta :: s_0\sigma\eta \xrightarrow{\circ}{}^{+} h_0\eta :: h_1\eta :: \cdots :: h_n\eta :: s\eta$ となる．手続きが成功するときは，ある $(x, s_0, s, \sigma) \in xs$ について，`check` (x, s_0, s, σ) が成功したときである．よって，項 s の安定位置 p および $\tau = \mathrm{match}_{inf}(s_0\sigma\delta^{\star}, s|_p\delta^{\star})$ となる代入 δ, τ が存在する．すなわち，$s_0\sigma\delta^{\star}\tau = s|_p\delta^{\star}$ が成立する．したがって，$s_0 \overset{\circ}{\rightsquigarrow}{}^{+}_{\sigma} s$ かつ補題 45 よ

り，任意の代入 η について，$s_0\sigma\delta^\star\eta \overset{\circ}{\to}^+ s\delta^\star\eta = s\delta^\star\eta[s_0\sigma\delta^\star\tau\eta]_p$ が成立する．以下の 2 通りに場合分けを行う．

- $p = \epsilon$ の場合．$\eta = \emptyset$ とする．このとき，$s_0\sigma\delta^\star \overset{\circ}{\to}^+ s_0\sigma\delta^\star\tau \overset{\circ}{\to}^+ s_0\sigma\delta^\star\tau\tau \overset{\circ}{\to}^+ \cdots$ なる無限書き換え列が存在する．このとき，$x = \mathtt{true}$ であるから，$s_0\sigma\delta^\star\tau^i \overset{\circ}{\to}^+ s_0\sigma\delta^\star\tau^{i+1}$ $(i = 0, 1, \ldots)$ は根書き換えを含む．よって，この無限書き換え列は最外公平書き換え列である．したがって，命題 43 より $s_0\sigma\delta^\star$ が ω-正規形をもつならば，この書き換え列によって到達可能になるはずだが，この書き換え列は根書き換えを無限回含むため ω-正規形に到達しない．つまり，$s_0\sigma\delta^\star$ は ω-正規形をもたない．よって，$s_0\sigma\delta^\star$ は $\mathcal{R}$ において生成的でない．さらに，補題 44 より，$t_0\rho\sigma\delta^\star$ は $\mathcal{R}$ において生成的でない．したがって，定義 33 から項 t_0 は $\mathcal{R}$ において一般生成的でない．
- $p \neq \epsilon$ の場合．$s_0\sigma(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$，$s(\epsilon) \neq \mathsf{cons}$ かつ条件 $\exists q < p.\ s(q) \in \mathcal{D}$ より，$s_0\sigma\delta^\star(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$，$s\delta^\star(\epsilon) \in \mathcal{F} \setminus \{\mathsf{cons}\}$，かつ，ある位置 $q < p$ について $s\delta^\star(q) = s(q) \in \mathcal{D}$．位置 p は項 s の安定位置より，p は $s\delta^\star$ の安定位置である．よって，補題 32 より，$s_0\sigma\delta^\star$ は $\mathcal{R}$ において生成的でない．さらに，補題 44 より，$t_0\rho\sigma\delta^\star$ は $\mathcal{R}$ において生成的でない．したがって，定義 33 から項 t_0 は $\mathcal{R}$ において一般生成的でない． □