書き換え帰納法における向き付け不能な等式の証明

Proving Non-Orientable Conjectures in Rewriting Induction

青戸等人
Takahito Aoto

東北大学 電気通信研究所
RIEC, Tohoku University

概 要

項書き換えシステムに基づく帰納的定理の自動証明法に Reddy(1989) により提案された書き換え帰納法がある．書き換え帰納法では帰納的定理証明の原理に簡約順序に基づくネーター帰納法を用いる．このため，与えられた簡約順序によって等式を向き付けできない場合，その等式の証明に失敗してしまう．また，交換律など，どのような簡約順序によっても向き付けできない等式に対する証明を取り扱うことは出来ない．本論文では，向き付け不能な等式に対する帰納的定理証明も取り扱うことの出来るように書き換え帰納法を拡張する．また，拡張した書き換え帰納法を項書き換えシステムに基づくプログラム変換へ応用する例を示す．

1 はじめに

プログラムの性質のうち，自然数やリストといったデータ構造に関する帰納法で示されるような性質をプログラムの帰納的な性質とよぶ．プログラムの帰納的な性質は，プログラムの検証やプログラム変換など，プログラムの意味の考察を伴う技術において重要な役割を果たす．このため，与えられたプログラムが必要な帰納的性質をもつかの判定，あるいは，必要な帰納的性質をもつことの証明の自動化法を確立することは重要な研究課題の 1 つと考えられる．

項書き換えシステム (term rewriting systems) は等式論理に基づく計算モデルであり，関数型プログラムのモデルとして考えることの可能な計算モデルの 1 つである．項書き換えシステムの帰納的な性質，特に等式として表された性質は帰納的定理 (inductive theorem) とよばれており，その自動証明法も古くから研究されている [3, 7, 8, 9, 10, 11, 12, 15]．書き換え帰納法 (rewriting induction) は，Reddy [12] により提案された帰納的定理の自動証明法であり，停止性を持つ項書き換えシステムを対象とする点やある種の合流性を必要とする潜在帰納法 [7, 8, 9, 11] と対照的なことから，帰納的定理の自動証明法として最もよく知られている方法の 1 つである．

書き換え帰納法では帰納的定理証明の原理に簡約順序に基づくネーター帰納法を用いる．このため，与えられた簡約順序によって等式が向き付けできない場合，その等式の証明に失敗してしまう．向き付けの失敗による証明の失敗は，証明が発散する場合とともに帰納的定理証明の主な失敗原因の 1 つとなっている．また，どのような簡約順序を用いるかに選択の余地はあるものの，交換律など，どのような簡約順序によっても向き付けできない等式の証明については，書き換え帰納法では取り扱うことは出来ない．

この欠点を改善するため，2 通りのアプローチが提案されている．1 つは文献 [12] において示唆されている等式を法とする書き換え (rewriting modulo equations) を用いる方法であり，

もう 1 つは，順序書き換え法 (ordered rewriting technique) を用いる方法 [5] である．定理証明器 SPIKE でも後者に類似した方法を用いている [3]．本論文では，1 番目のアプローチに従って書き換え帰納法を拡張し，向き付け不能な等式に対する帰納的定理証明も取り扱うことの出来るようにする．まず [12] における考察の問題点を指摘し，不充分な点を補って，拡張書き換え帰納法の手続きを与える．また，その手続きが成功する場合には入力された等式が帰納的定理となること (拡張書き換え帰納法の正当性) を示す．最後に，拡張した書き換え帰納法を項書き換えシステムに基づくプログラム変換へ応用する例を示す．

2 準備

本論文では，文献 [10] のアプローチを踏襲し，書き換え帰納法の正当性を抽象書き換えシステム上の性質を応用する形で示す．そのため，抽象書き換えシステムおよび関連する記法についてまず紹介する．

集合 A と A 上の 2 項関係 $\to$ の対 $\mathcal{A} = \langle A, \to \rangle$ を抽象書き換えシステムという．関係 $\to$ の反射推移閉包を $\xrightarrow{*}$，関係 $\to$ の推移閉包を $\xrightarrow{+}$，対称閉包を $\leftrightarrow$，等価閉包を $\overset{*}{\leftrightarrow}$ と記す．関係 $\to$ が整礎である ($a_0 \to a_1 \to \cdots$ なる無限列が存在しない) ことを $\mathrm{SN}(\to)$ と記す．要素 $a \in A$ が，$a \to b$ なる $b \in A$ を持たないとき a を正規形とよぶ．正規形の集合を $\mathrm{NF}(\to)$ と記す．2 つの関係 $\to_1, \to_2$ があるとき，その和 $\to_1 \cup \to_2$ を $\to_{1\cup 2}$ と記す．また，$\to_1/\to_2$ は $\overset{*}{\leftrightarrow}_2 \circ \to_1 \circ \overset{*}{\leftrightarrow}_2$ により定義される関係を表す．簡単のため $\to_1/\to_2$ を $\to_{1/2}$ とも記す．以下では，$/$ は $\cup$ より強い結合力を持つものとし，閉包操作は $/, \cup$ より強い結合力を持つものとする．従って，例えば $\overset{*}{\leftrightarrow}_{1\cup 2}$ は $\to_1 \cup \to_2$ の等価閉包を表す．

$>$ を整礎な半順序とするとき，以下で表されるネーター帰納法が妥当な推論となる [6]．

$$\frac{\forall x \in \mathrm{NF}(>).\ \mathcal{P}(x) \quad \forall x \in A.\ (\forall y \in A.\ (x > y \Rightarrow \mathcal{P}(y)) \Rightarrow \mathcal{P}(x))}{\forall x \in A.\ \mathcal{P}(x)}$$

ここで，$\mathcal{P}(x)$ は x に関する述語を表す．$\mathrm{NF}(>)$ は極小元集合と等しいことに注意する．以下の命題は $>$ に関するネーター帰納法により証明される．

命題 1 (書き換え帰納法の原理 [10]) $\mathcal{A}_1 = \langle A, \to_1 \rangle$，$\mathcal{A}_2 = \langle A, \to_2 \rangle$ を抽象書き換えシステムとし，$>$ を A 上の整礎な半順序とする．

(i) $\to_{1\cup 2} \subseteq >$

(ii) $\to_2 \subseteq \to_1 \circ \overset{*}{\to}_{1\cup 2} \circ \overset{*}{\leftarrow}_{1\cup 2}$

が成立しているならば，$\overset{*}{\leftrightarrow}_1 = \overset{*}{\leftrightarrow}_{1\cup 2}$．

次に本論文で用いる項書き換えシステムの定義および記法について紹介する．項書き換えシステムについての詳細は文献 [2, 14] 等を参照して頂きたい．

関数記号集合と変数集合を $\mathcal{F}$ と V で表す．各関数記号 $f \in \mathcal{F}$ はアリティ $\mathrm{arity}(f) \in \mathbb{N}$ を持つものとする．$\mathcal{F}, V$ 上の項集合は以下により帰納的に定義される：(1) $V \subseteq \mathrm{T}(\mathcal{F}, V)$; (2) $f \in \mathcal{F}$, $\mathrm{arity}(f) = n\ (\geq 0)$, $t_1, \ldots, t_n \in \mathrm{T}(\mathcal{F}, V)$ なら，$f(t_1, \ldots, t_n) \in \mathrm{T}(\mathcal{F}, V)$．項の (構文的) 同一性を $\equiv$ で表す．項 t に現れる変数全体の集合を $V(t)$ と書く．項 t の根記号 $\mathrm{root}(t) \in \mathcal{F} \cup V$ は以下により定義される：(1) $\mathrm{root}(x) = x\ (x \in V)$; (2) $\mathrm{root}(f(t_1, \ldots, t_n)) = f$．

写像 $\sigma : V \to \mathrm{T}(\mathcal{F}, V)$ のうち，有限の定義域を持つものを代入という．代入 σ の定義域を $\mathrm{T}(\mathcal{F}, V)$ へと拡げた準同型拡張も同じく σ で表す．項 $\sigma(t)$ を項 t の代入例とよび，慣例に従って $\sigma(t)$ を $t\sigma$ と記す．項 s, t に同一の代入例が存在するとき s と t は単一化可能であるといい，

$s\sigma \equiv t\sigma$ なる代入 σ を (s と t の) 単一化子という．s と t が単一化可能であるとき，最も一般的な単一化子があることが知られている．これを最汎単一化子とよび，$\mathrm{mgu}(s,t)$ と記す．

$\mathrm{T}(\mathcal{F},V)$ 上の項の対 $\langle l,r\rangle$ のうち，条件 (1) $\mathrm{root}(l)\in\mathcal{F}$; (2) $V(r)\subseteq V(l)$ を満たすものを項書き換え規則といい，項書き換え規則集合を項書き換えシステムともよぶ．以下では慣例に従って項書き換え規則 $\langle l,r\rangle$ を $l\to r$ と記す．どの関数記号集合 $\mathcal{F}$ 上の項を考えているかが自明でないときは，関数記号集合 $\mathcal{F}$ と項書き換え規則集合 $\mathcal{R}$ の対 $\langle\mathcal{F},\mathcal{R}\rangle$ を項書き換えシステムとよぶこともある．$\mathcal{F}$ を省略した場合，暗黙のうちに $\mathcal{F}$ は項書き換え規則集合に現れる関数記号全体の集合であると仮定する．

ある項書き換え規則の左辺の根記号となっている関数記号を定義記号とよぶ．($\mathcal{R}$ の) 定義記号集合 $\{\mathrm{root}(l)\mid l\to r\in\mathcal{R}\}$ を $\mathcal{D}_{\mathcal{R}}$ と記す ($\mathcal{R}$ が文脈から自明なときは添字 $_{\mathcal{R}}$ は略す，以下同様)．項 t に現れる定義記号の集合を $\mathcal{D}(t)$ と記す．集合 $\mathcal{F}\setminus\mathcal{D}$ の要素を構成子記号とよび，構成子記号集合を $\mathcal{C}$ と記す．項 $t\in\mathrm{T}(\mathcal{C},V)$ を構成子項とよぶ．任意の定義域上の変数 x について $\sigma(x)\in\mathrm{T}(\mathcal{C},V)$ となる代入 σ を構成子代入とよぶ．定義記号 $f\in\mathcal{D}$ と構成子項 $c_1,\ldots,c_n\in\mathrm{T}(\mathcal{C},V)$ について $f(c_1,\ldots,c_n)$ となる項を基本項とよぶ．全ての項書き換え規則 $l\to r\in\mathcal{R}$ について l が基本項であるとき，項書き換えシステム $\mathcal{R}$ を構成子システムとよぶ．

$\Box$ (ホール) を $\mathcal{F}$ に出現しないある特別な定数とする．$\mathrm{T}(\mathcal{F}\cup\{\Box\},V)$ の要素を文脈とよぶ．文脈 C に n 個のホールが出現するとき，C のホールを左から順に項 $t_1,\ldots,t_n$ で置きかえて得られる項を $C[t_1,\ldots,t_n]$ と記す．$s\equiv C[u]$ となるとき，項 u を項 s の部分項とよび，$u\trianglelefteq s$ と記す．部分項のうち基本項であるものを基本部分項とよび，項 s の基本部分項集合 $\{u\trianglelefteq s\mid\exists f\in\mathcal{D}.\,\exists c_1,\ldots,c_n\in\mathrm{T}(\mathcal{C},V).\,u\equiv f(c_1,\ldots,c_n)\}$ を $\mathcal{B}(s)$ と記す．

$\mathcal{R}$ を項書き換えシステムとする．ある文脈 C，代入 σ，項書き換え規則 $l\to r\in\mathcal{R}$ が存在して，$s\equiv C[l\sigma]$ かつ $t\equiv C[r\sigma]$ となるとき，$s\to_{\mathcal{R}}t$ と記す．項集合 $\mathrm{T}(\mathcal{F},V)$ 上の関係 $\to_{\mathcal{R}}$ を項書き換えシステム $\mathcal{R}$ の簡約関係とよぶ．抽象書き換えシステム $\langle\mathrm{T}(\mathcal{F},V),\to_{\mathcal{R}}\rangle$ 上に定義される各種の概念や記法を踏襲する．等式 $l\doteq r$ とは $\mathrm{T}(\mathcal{F},V)$ 上の項の対 $\langle l,r\rangle$ である．特に等式 $l\doteq r$ という場合，$\langle l,r\rangle$ と $\langle r,l\rangle$ を区別しないで用いる．等式集合 E による簡約関係も項書き換えシステムの簡約関係と同様に定義される．すなわち，ある文脈 C，代入 σ，等式 $l\doteq r\in E$ が存在して，$s\equiv C[l\sigma]$ かつ $t\equiv C[r\sigma]$ となるか，$t\equiv C[l\sigma]$ かつ $s\equiv C[r\sigma]$ となるとき，$s\leftrightarrow_E t$ と記す．

$V(t)=\emptyset$ となる項 t を基底項とよぶ．関数記号集合 $\mathcal{F}$ 上の基底項全体の集合を $\mathrm{T}(\mathcal{F})$ と記す．項 t の代入例が基底項であるとき，すなわち，$t\sigma\in\mathrm{T}(\mathcal{F})$ のとき $t\sigma$ を t の基底代入例とよぶ．項書き換え規則や等式の基底代入例も同様に定める．任意の定義域上の変数 x について $\sigma_g(x)\in\mathrm{T}(\mathcal{F})$ となる代入 σ_g を基底代入とよぶ．任意の基本基底項が $\mathcal{R}$ の正規形でないとき，項書き換えシステム $\mathcal{R}$ を擬簡約であるという．本論文では，基底代入 σ_g を項 t に適用した項 $t\sigma_g$ を取り上げるとき，一般性を失うことなく，$t\sigma_g$ は基底代入例である (つまり，σ_g の定義域は $V(t)$ を含む) と仮定する．項書き換えシステム $\mathcal{R}$ の帰納的定理とは基底項集合上で成立する等式，つまり，どのような基底代入例 $s\sigma_g\doteq t\sigma_g$ についても，$s\sigma_g\overset{*}{\leftrightarrow}_{\mathcal{R}}t\sigma_g$ が成立するような等式 $s\doteq t$ のことをいう．

例 **2** (帰納的定理) 以下は自然数の加算を計算する項書き換えシステムである．

$$\mathcal{R}=\left\{\begin{array}{lcl}\mathsf{plus}(\mathsf{0},y) & \to & y\\ \mathsf{plus}(\mathsf{s}(x),y) & \to & \mathsf{s}(\mathsf{plus}(x,y))\end{array}\right\}.$$

このとき，加算の結合律を表す以下の等式

$$\mathsf{plus}(\mathsf{plus}(x,y),z)\quad\doteq\quad\mathsf{plus}(x,\mathsf{plus}(y,z))$$

Simplify

$$\frac{\langle E \uplus \{s \doteq t\},\ H\rangle}{\langle E \cup \{s' \doteq t\},\ H\rangle} \quad s \to_{\mathcal{R}\cup H} s'$$

Delete

$$\frac{\langle E \uplus \{s \doteq s\},\ H\rangle}{\langle E,\ H\rangle}$$

Expand

$$\frac{\langle E \uplus \{s \doteq t\},\ H\rangle}{\langle E \cup \mathrm{Expd}_u(s,t),\ H \cup \{s \to t\}\rangle}\ u \in \mathcal{B}(s), s > t$$

ここで

$$\mathrm{Expd}_u(s,t) \quad = \quad \{C[r]\sigma \doteq t\sigma \mid s \equiv C[u],\ \sigma = \mathrm{mgu}(u,l),\ l \to r \in \mathcal{R}\}$$

図 1: 書き換え帰納法の推論規則

は $\mathcal{R}$ の帰納的定理となっている．一方，$\mathsf{plus}(\mathsf{plus}(x,y),z) \overset{*}{\leftrightarrow}_{\mathcal{R}} \mathsf{plus}(x,\mathsf{plus}(y,z))$ は成立していない．なぜなら，$\mathsf{plus}(\mathsf{plus}(x,y),z)$ も $\mathsf{plus}(x,\mathsf{plus}(y,z))$ も，変数 x,y,z が具体化されておらず項書き換え規則が適用できない (つまり正規形となっている) からである．この例は帰納的定理は必ずしも等式論理の定理とはならないことを示している． □

簡約関係 $\to_{\mathcal{R}}$ が整礎であるとき，項書き換えシステム $\mathcal{R}$ は停止性をもつという．項集合上の関係 R が代入に関して閉じているとは任意の代入 σ について $s\,R\,t \Rightarrow s\sigma\,R\,t\sigma$ となることをいい，文脈に関して閉じているとは，任意の文脈 C について $s\,R\,t \Rightarrow C[s]\,R\,C[t]$ となることをいう．代入および文脈に関して閉じている整礎な半順序を簡約順序とよぶ．また，代入および文脈に関して閉じている擬順序 $\succsim$ で $\succ = \succsim \setminus \precsim$ が簡約順序となるものを簡約擬順序とよぶ．関係 $\succsim \cap \precsim$ を $\approx$ と記す．項書き換えシステム $\mathcal{R}$ が停止性をもつことと $\forall l \to r \in R.\ (l > r)$ となる簡約順序 $>$ が存在することが同値であることは容易に示せる．辞書式経路順序および再帰経路順序は特によく知られた簡約 (擬) 順序である．

3 書き換え帰納法

書き換え帰納法は Reddy [12] により提案された帰納的定理の自動証明法である．書き換え帰納法では，等式集合 E と項書き換え規則集合 H の対に関する推論を適用することで導出を行う．直観的には，E はこれから証明しようとする等式を，H は帰納法の仮定および証明済みの定理 (として使える項書き換え規則) を表す．図 1 に書き換え帰納法の (下向きの) 推論規則を示す．$\uplus$ は直和 (disjoint union) を表す．等式の向きは区別しないことに注意する．$\mathcal{R}$ および $>$ は入力として与えられる項書き換えシステムと簡約順序である．

次の Expd の性質は後の証明に用いる．

補題 3 (Expd の性質) $\mathcal{R}$ を擬簡約な項書き換えシステムとし，$u \in \mathcal{B}(s)$ とする．
(1) 任意の基底構成子代入 σ_g について $s\sigma_g \to_{\mathcal{R}} \circ \leftrightarrow_{\mathrm{Expd}_u(s,t)} t\sigma_g$．
(2) $v \leftrightarrow_{\mathrm{Expd}_u(s,t)} w \Rightarrow v \overset{*}{\leftrightarrow}_{\mathcal{R}\cup\{s\doteq t\}} w$

証明．(1) u が基本項，σ_g が基底構成子代入であることから，$u\sigma_g$ は基本基底項．よって，$\mathcal{R}$ の擬簡約性から，ある $l \to r \in \mathcal{R}$ が存在して，$u\sigma_g$ は l の代入例になっている．今，一般性

を失うことなく $V(l) \cap V(s) = \emptyset$ としてよいから，σ_g を拡張して $u\sigma_g \equiv l\sigma_g$ とする．このとき σ_g は u と l の単一化子であるから，$\sigma = \mathrm{mgu}(u, l)$ とおくと $\sigma_g = \theta_g \circ \sigma$ なる基底代入 θ_g が存在する．ゆえに，$s \equiv C[u]$ とおいて，$s\sigma_g \equiv C\sigma_g[u\sigma_g] \equiv C\sigma_g[u\sigma\theta_g] \equiv C\sigma_g[l\sigma\theta_g] \to_{\mathcal{R}} C\sigma_g[r\sigma\theta_g] \equiv C[r]\sigma\theta_g \leftrightarrow_{\mathrm{Expd}_u(s,t)} t\sigma\theta_g \equiv t\sigma_g$．(2) $v \leftrightarrow_{\mathrm{Expd}_u(s,t)} w$ とする．すると，ある文脈 $\hat{C}$ および代入 $\hat{\sigma}$ に対して $v \equiv \hat{C}[C[r]\sigma\hat{\sigma}]$, $w \equiv \hat{C}[t\sigma\hat{\sigma}]$ (または $w \equiv \hat{C}[C[r]\sigma\hat{\sigma}]$, $v \equiv \hat{C}[t\sigma\hat{\sigma}]$)．ただし，ここで $\sigma = \mathrm{mgu}(u, l)$, $s \equiv C[u]$, $l \to r \in \mathcal{R}$ とする．すると，$v \equiv \hat{C}[C[r]\sigma\hat{\sigma}] \leftarrow_{\mathcal{R}} \hat{C}[C[l]\sigma\hat{\sigma}] \equiv \hat{C}[C\sigma[l\sigma]\hat{\sigma}] \equiv \hat{C}[C\sigma[u\sigma]\hat{\sigma}] \equiv \hat{C}[C[u]\sigma\hat{\sigma}] \equiv \hat{C}[s\sigma\hat{\sigma}] \leftrightarrow_{\{s \doteq t\}} \hat{C}[t\sigma\hat{\sigma}] \equiv w$. □

定義 4 (書き換え帰納法)

1. 等式集合と項書き換え規則集合の対について，図 1 の推論規則を適用して，$\langle E, H\rangle$ から $\langle E', H'\rangle$ が得られることを $\langle E, H\rangle \rightsquigarrow \langle E', H'\rangle$ と記す．$\rightsquigarrow$ の反射推移閉包を $\overset{*}{\rightsquigarrow}$ と記す．必要に応じて，$\rightsquigarrow$ に添字をつけ *Simplify* 規則 (*Delete* 規則，*Expand* 規則) による導出を $\rightsquigarrow^s$ ($\rightsquigarrow^d$, $\rightsquigarrow^e$) と表す．

2. 書き換え帰納法の手続き
   入力: 擬簡約な構成子システム $\mathcal{R}$，
   等式集合 E，
   $\mathcal{R} \subseteq >$ なる簡約順序 $>$
   出力: 証明結果 ("成功" または "失敗")

   1. $\langle E, \emptyset\rangle$ から始めて，図 1 の推論規則を可能な限り繰り返し適用する．
   2. その結果，ある H に対して $\langle E, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H\rangle$ となれば "成功" を，ならなければ "失敗" を返す． □

書き換え帰納法は半アルゴリズムである．すなわち，等式集合が空にならないまま無限に導出を繰り返して手続きが停止しないことがあり得る．この場合を手続きが発散するともいう．また，規則を適用する等式の選び方や適用する推論規則に任意性があるので上記の手続きは非決定的である．

以下の命題は，書き換え帰納法の手続きが "成功" を返した場合，入力等式集合 E が入力項書き換えシステム $\mathcal{R}$ の帰納的定理であることを保証する．証明は書き換え帰納法の原理 (命題 1) を抽象書き換えシステム $\langle \mathrm{T}(\mathcal{F}), \to_{\mathcal{R}}\rangle$ と $\langle \mathrm{T}(\mathcal{F}), \to_H\rangle$ に対して適用する．

命題 5 (書き換え帰納法の正当性 [12]) $\mathcal{R}$ を擬簡約な構成子システム，E を等式集合，$>$ を $\mathcal{R} \subseteq >$ なる簡約順序とする．ある項書き換え規則集合 H が存在して，$\langle E, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H\rangle$ となるならば，E 中の等式は $\mathcal{R}$ の帰納的定理である．

例 6 (書き換え帰納法の実行例) 入力項書き換えシステム $\mathcal{R}$ と証明する等式集合 E として以下が与えられたとする．

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{plus}(0, y) & \to & y \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

$$E = \left\{ \begin{array}{lcl} \mathsf{plus}(\mathsf{plus}(x, y), z) & \doteq & \mathsf{plus}(x, \mathsf{plus}(y, z)) \end{array} \right\}$$

とする．以下は，$>$ を優位順序 $\mathsf{plus} > \mathsf{s} > 0$ に基づく辞書式経路順序としたときの書き換え帰納法の実行過程の例である．

$$\left\langle \left\{ \begin{array}{l} \mathsf{plus}(\mathsf{plus}(x, y), z) \doteq \mathsf{plus}(x, \mathsf{plus}(y, z)) \end{array} \right\}, \left\{ \right\} \right\rangle$$

$$\rightsquigarrow^{e} \qquad \left\langle \left\{ \begin{array}{l} \mathsf{plus}(y_0, z) \doteq \mathsf{plus}(0, \mathsf{plus}(y_0, z)) \\ \mathsf{plus}(\mathsf{s}(\mathsf{plus}(x_1, y_1)), z) \doteq \mathsf{plus}(\mathsf{s}(x_1), \mathsf{plus}(y_1, z)) \end{array} \right\}, \left\{ \mathsf{plus}(\mathsf{plus}(x, y), z) \to \mathsf{plus}(x, \mathsf{plus}(y, z)) \right\} \right\rangle$$

$$\rightsquigarrow^{s}\rightsquigarrow^{s}\rightsquigarrow^{s} \quad \left\langle \left\{ \begin{array}{l} \mathsf{plus}(y_0, z) \doteq \mathsf{plus}(y_0, z) \\ \mathsf{s}(\mathsf{plus}(x_1, \mathsf{plus}(y_1, z)) \doteq \mathsf{s}(\mathsf{plus}(x_1, \mathsf{plus}(y_1, z)) \end{array} \right\}, \left\{ \mathsf{plus}(\mathsf{plus}(x, y), z) \to \mathsf{plus}(x, \mathsf{plus}(y, z)) \right\} \right\rangle$$

$$\rightsquigarrow^{d}\rightsquigarrow^{d} \qquad \left\langle \left\{ \right\}, \left\{ \mathsf{plus}(\mathsf{plus}(x, y), z) \to \mathsf{plus}(x, \mathsf{plus}(y, z)) \right\} \right\rangle$$

□

4 向き付け不能な等式に対する書き換え帰納法

書き換え帰納法の鍵となるのは $Expand$ 規則であるが，$Expand$ 規則は入力簡約順序によって向き付けが可能となるような等式にしか適用できない．このため，入力された簡約順序によって帰納的定理が向き付けできない場合，その帰納的定理の証明は失敗してしまう．

例 7 (書き換え帰納法の失敗) 加算の項書き換えシステム

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{plus}(0, y) & \to & y \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

において，加法の交換律

$$\mathsf{plus}(x, y) \qu\doteq \quad \mathsf{plus}(y, x)$$

は帰納的定理となっている．しかしながら，$\mathsf{plus}(x, y) > \mathsf{plus}(y, x)$ も $\mathsf{plus}(y, x) > \mathsf{plus}(x, y)$ も成立しない[1]ため，書き換え帰納法で自動証明を試みると，どの規則も適用できず，$E \neq \emptyset$ のまま停止し証明に失敗する． □

向き付け不能な等式を取り扱うための解決法として $\to_{\mathcal{R}} \cup \to_H$ を $\to_{\mathcal{R}}/\to_H$ に改めることが文献 [12] 注 14 で提案されている．しかしながら，同文献命題 13 に対応する抽象書き換えシステムにおける以下の命題は成立しない．

予想 8 (抽象書き換えシステムの正しくない性質) $\mathcal{A}_1 = \langle A, \to_1 \rangle$，$\mathcal{A}_2 = \langle A, \to_2 \rangle$ を抽象書き換えシステムとし，$\succsim$ を整礎な擬順序とする．

(i) $\to_1 \subseteq \succ$

(ii) $\to_2 = \leftarrow_2 \subseteq \approx$

(iii) $\to_2 \subseteq \to_1 \circ \overset{*}{\to}_{1/2} \circ \overset{*}{\leftarrow}_{1/2}$

が成立しているならば，$\overset{*}{\leftrightarrow}_1 = \overset{*}{\leftrightarrow}_{1\cup 2}$．

例 9 (予想 8 に対する反例) 集合 $A = \{a, b, c\}$ および A 上の 2 つの関係 $\to_1 = \{\langle a, b\rangle\}$, $\to_2 = \{\langle a, c\rangle\}$，および $c \approx a \succ b$ なる擬順序 $\succsim$ を考える．明らかに条件 (i),(ii) は成立する．また，$a \to_1 b \leftarrow_1 a \leftrightarrow_2 c$ であるから条件 (iii) も成立する．しかしながら，$c \overset{*}{\leftrightarrow}_{1\cup 2} b$ だが $c \overset{*}{\leftrightarrow}_1 b$ でない． □

[1] $\mathsf{plus}(x, y) > \mathsf{plus}(y, x)$ が成立するとすると，簡約順序は代入に閉じているため $\mathsf{plus}(0, 0) > \mathsf{plus}(0, 0)$ が成立することになってしまい，これは $>$ が半順序であることに矛盾する．従って，どのような簡約順序を用いてもこれらの等式は向き付け出来ない．

Simplify

$$\frac{\langle E \uplus \{s \doteq t\},\ H\rangle}{\langle E \cup \{s' \doteq t\},\ H\rangle} \quad s \to_{\mathcal{R}/H} s'$$

Delete

$$\frac{\langle E \uplus \{s \doteq t\},\ H\rangle}{\langle E,\ H\rangle} \quad s \overset{*}{\leftrightarrow}_H t$$

Expand

$$\frac{\langle E \uplus \{s \doteq t\},\ H\rangle}{\langle E \cup \mathrm{Expd}_u(s,t),\ H \cup \{s \doteq t\}\rangle}\ u \in \mathcal{B}(s), s \approx t$$

図 2: 非健全な推論規則

また，同様に書き換え帰納法の推論規則における $\mathcal{R} \cup H$ を $\mathcal{R}/H$ に改めて得られる推論規則を図 2 に示す．この推論規則もまた次に示すように健全ではない．

例 **10** (非健全な推論規則による書き換え帰納法の実行) リストの連結を計算する app に関する以下の項書き換えシステム $\mathcal{R}$ を考える．

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{app}(\mathsf{nil}, ys) & \to & ys \\ \mathsf{app}(\mathsf{cons}(x, xs), ys) & \to & \mathsf{cons}(x, \mathsf{app}(xs, ys)) \end{array} \right\}$$

このとき，リストの連結は交換律を満たさないから，

$$\mathsf{app}(xs, ys) \doteq \mathsf{app}(ys, xs)$$

は $\mathcal{R}$ の帰納的定理ではない．しかしながら，$\gtrsim$ を優位順序 app $\succ$ cons $\succ$ nil に基づく再帰経路順序としたとき，以下に示すように書き換え帰納法が成功してしまう．

$$\left\langle \left\{ \mathsf{app}(xs, ys) \doteq \mathsf{app}(ys, xs) \right\}, \left\{ \right\} \right\rangle$$

$$\rightsquigarrow^{e} \quad \left\langle \begin{array}{l} \left\{ \begin{array}{l} ys \doteq \mathsf{app}(ys, \mathsf{nil}) \\ \mathsf{cons}(x_1, \mathsf{app}(xs_1, ys)) \doteq \mathsf{app}(ys, \mathsf{cons}(x_1, xs_1)) \end{array} \right\} \\ \left\{ \mathsf{app}(xs, ys) \doteq \mathsf{app}(ys, xs) \right\} \end{array} \right\rangle$$

$$\rightsquigarrow^{s}\rightsquigarrow^{s} \quad \left\langle \begin{array}{l} \left\{ \begin{array}{l} ys \doteq ys \\ \mathsf{cons}(x_1, \mathsf{app}(xs_1, ys)) \doteq \mathsf{cons}(x_1, \mathsf{app}(xs_1, ys)) \end{array} \right\} \\ \left\{ \mathsf{app}(xs, ys) \doteq \mathsf{app}(ys, xs) \right\} \end{array} \right\rangle$$

$$\rightsquigarrow^{d}\rightsquigarrow^{d} \quad \left\langle \left\{ \right\}, \left\{ \mathsf{app}(xs, ys) \doteq \mathsf{app}(ys, xs) \right\} \right\rangle$$

ここで，*Simplify* 規則の適用では，

$$\begin{array}{rcl} \mathsf{app}(ys, \mathsf{nil}) & \leftrightarrow_H & \mathsf{app}(\mathsf{nil}, ys) \\ & \to_{\mathcal{R}} & ys \end{array}$$

および

$$\begin{array}{rcl} \mathsf{app}(ys, \mathsf{cons}(x_1, xs_1)) & \leftrightarrow_H & \mathsf{app}(\mathsf{cons}(x_1, xs_1), ys) \\ & \to_{\mathcal{R}} & \mathsf{cons}(x_1, \mathsf{app}(xs_1, ys)) \end{array}$$

を用いている． □

次に，我々の拡張書き換え帰納法の手続きの原理となる抽象書き換えシステムに関する性質を与える．

補題 11 (拡張書き換え帰納法の原理) $\mathcal{A}_1 = \langle A, \to_1 \rangle$，$\mathcal{A}_2 = \langle A, \to_2 \rangle$, $\mathcal{A}_3 = \langle A, \to_3 \rangle$ を抽象書き換えシステムとし，$\succsim$ を整礎な擬順序とする．

(i) $\to_{1\cup 2} \subseteq \succ$
(ii) $\to_3 = \leftarrow_3 \subseteq \approx$
(iii) $\to_2 \subseteq \to_1 \circ \overset{*}{\to}_{(1\cup 2)/3} \circ \overset{*}{\leftrightarrow}_3 \circ \overset{*}{\leftarrow}_{(1\cup 2)/3}$
(iv) $\to_3 \subseteq \to_1 \circ \overset{*}{\to}_{(1\cup 2)/3} \circ \overset{*}{\leftrightarrow}_3 \circ \overset{*}{\leftarrow}_{(1\cup 2)/3} \circ \leftarrow_1$
(v) $\forall x, y \in \mathrm{NF}(\to_{(1\cup 2)/3}).\ (x \overset{*}{\leftrightarrow}_3 y \Rightarrow x = y)$
が成立しているならば，$\overset{*}{\leftrightarrow}_1 = \overset{*}{\leftrightarrow}_{1\cup 2\cup 3}$．

証明．$\subseteq$ は明らかなので $\supseteq$ を示す．まず，$\succ$ に関するネーター帰納法で，任意の $x \in A$ について

$$\forall y \in A.\ (x \overset{*}{\to}_{(1\cup 2)/3} y \Rightarrow x \overset{*}{\leftrightarrow}_1 y) \tag{1}$$

が成立することを示す．

(Base Step) このとき $x = y$ であるから，$x \overset{*}{\leftrightarrow}_1 y$ は明らか．

(Induction Step) $x \overset{*}{\to}_{(1\cup 2)/3} y$ とする．$x = y$ の場合は明らかなので，$x \overset{*}{\leftrightarrow}_3 u \to_{1\cup 2} v \overset{*}{\leftrightarrow}_3 z \overset{*}{\to}_{(1\cup 2)/3} y$ の場合を考える．$x \succ z$ だから z に対して帰納法の仮定が適用でき，$z \overset{*}{\leftrightarrow}_1 y$ が得られる．

ここでまず，

$$\forall a, b \precsim x\ (a \to_3 b \Rightarrow a \overset{*}{\leftrightarrow}_1 b) \tag{2}$$

を示す．$a \to_3 b$ と条件 (iv) から，$a \to_1 c \overset{*}{\to}_{(1\cup 2)/3} c' \overset{*}{\leftrightarrow}_3 d' \overset{*}{\leftarrow}_{(1\cup 2)/3} d \leftarrow_1 b$ なる c, c', d, d' が存在する．このとき $x \succ c, d$ より帰納法の仮定より $c \overset{*}{\leftrightarrow}_1 c',\ d \overset{*}{\leftrightarrow}_1 d'$ が成立する．従って，$c', d' \in \mathrm{NF}(\to_{(1\cup 2)/3})$ ならば条件 (v) より性質 (2) が成立する．今，$c' \notin \mathrm{NF}(\to_{(1\cup 2)/3})$ の場合を考える．すると，条件 (i),(ii) より $c' \overset{+}{\to}_{(1\cup 2)/3} n \in \mathrm{NF}(\to_{(1\cup 2)/3})$ なる n が存在する．このとき $d' \overset{*}{\leftrightarrow}_3 c' \overset{+}{\to}_{(1\cup 2)/3} n$ より $d' \overset{*}{\to}_{(1\cup 2)/3} n$．よって，$a \to_1 c \overset{*}{\to}_{(1\cup 2)/3} n \overset{*}{\leftarrow}_{(1\cup 2)/3} d' \overset{*}{\leftarrow}_{(1\cup 2)/3} d \leftarrow_1 b$．すると帰納法の仮定より $a \to_1 c \overset{*}{\leftrightarrow}_1 n \overset{*}{\leftrightarrow}_1 d \leftarrow_1 b$．また，$d' \notin \mathrm{NF}(\to_{(1\cup 2)/3})$ の場合も同様なことが成立するので，性質 (2) が成立する．

性質 (2) より $x \overset{*}{\leftrightarrow}_1 u$, $v \overset{*}{\leftrightarrow}_1 z$ が成立するので，あとは $u \overset{*}{\leftrightarrow}_1 v$ を示せばよい．$u \to_1 v$ のときは明らかなので $u \to_2 v$ の場合を考える．このとき条件 (iii) より，$u \to_1 w \overset{*}{\to}_{(1\cup 2)/3} w' \overset{*}{\leftrightarrow}_3 v' \overset{*}{\leftarrow}_{(1\cup 2)/3} v$ なる w, w', v' が存在する．このとき，上と同様に，$w', v' \in \mathrm{NF}(\to_{(1\cup 2)/3})$ としてよい．すると条件 (v) および w, v に対する帰納法の仮定より $u \to_1 w \overset{*}{\leftrightarrow}_1 w' \overset{*}{\leftrightarrow}_1 v' \overset{*}{\leftrightarrow}_1 v$．以上で任意の $x \in A$ が性質 (1) を満たすことが示された．

次に，$\to_3 \subseteq \overset{*}{\leftrightarrow}_1$ を示す．これは上の性質 (2) の証明と同様に，ただし，帰納法の仮定の代わりにすでに示した性質 (1) を用いることにより示されるので，ここでは略す．

最後に，$\overset{*}{\to}_{1\cup 2\cup 3} \subseteq \overset{*}{\to}_{(1\cup 2)/3} \cup \overset{*}{\leftrightarrow}_3$ であることから題意が導かれる． □

この抽象書き換えシステムの原理に基づいて設計した拡張書き換え帰納法の推論規則を図 3 に示す．拡張書き換え帰納法の手続きは次のようになる．

Simplify

$$\frac{\langle E \uplus \{s \doteq t\},\ H,\ G\rangle}{\langle E \cup \{s' \doteq t\},\ H,\ G\rangle}\ \ s \to_{(\mathcal{R}\cup H)/G} s'$$

Delete

$$\frac{\langle E \uplus \{s \doteq t\},\ H,\ G\rangle}{\langle E,\ H,\ G\rangle}\ \ s \overset{*}{\leftrightarrow}_G t$$

Expand

$$\frac{\langle E \uplus \{s \doteq t\},\ H,\ G\rangle}{\langle E \cup \mathrm{Expd}_u(s,t),\ H \cup \{s \to t\},\ G\rangle}\ u \in \mathcal{B}(s), s \succ t$$

Expand2

$$\frac{\langle E \uplus \{s \doteq t\}, H, G\rangle}{\langle E \cup \mathrm{Expd2}_{u,v}(s,t), H, G \cup \{s \doteq t\}\rangle}\ u \in \mathcal{B}(s), v \in \mathcal{B}(t), s \approx t$$

ここで，

$$\mathrm{Expd2}_{u,v}(s,t) \quad = \quad \bigcup \Big\{ \mathrm{Expd}_{v\sigma}(t\sigma, s') \mid \langle s', t\sigma\rangle \in \mathrm{Expd}_u(s,t) \Big\}$$

図 3: 拡張書き換え帰納法の推論規則

定義 12 (拡張書き換え帰納法)

1. 等式集合，項書き換え規則集合，等式集合の 3 つ組について，図 3 の推論規則を適用して，$\langle E, H, G\rangle$ から $\langle E', H', G'\rangle$ が得られることを $\langle E, H, G\rangle \rightsquigarrow \langle E', H', G'\rangle$ と記す．$\rightsquigarrow$ の反射推移閉包を $\overset{*}{\rightsquigarrow}$ と記す．必要に応じて，$\rightsquigarrow$ に添字をつけ *Simplify* 規則 (*Delete* 規則，*Expand* 規則，*Expand2* 規則) による導出を $\rightsquigarrow^s$ ($\rightsquigarrow^d$, $\rightsquigarrow^e$, $\rightsquigarrow^{e2}$) と表す．

2. 拡張書き換え帰納法の手続き
   入力: 擬簡約な構成子システム $\mathcal{R}$，
   等式集合 E，
   $\mathcal{R} \subseteq \succ$ なる簡約擬順序 $\succsim$
   出力: 証明結果 ("成功" または "失敗")
   1. $\langle E, \emptyset, \emptyset\rangle$ から始めて，図 3 の推論規則を可能な限り繰り返し適用する．
   2. その結果，ある H, G に対して $\langle E, \emptyset, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H, G\rangle$ となれば "成功" を，ならなければ "失敗" を返す． □

拡張書き換え帰納法の推論規則のうち，*Expand* 規則は書き換え帰納法の推論規則をそのまま継承したものとなっている．*Simplify* 規則では $\to_{\mathcal{R}\cup H}$ による簡約を一般化した $\to_{(\mathcal{R}\cup H)/G}$ による簡約が可能となっている．*Delete* 規則は，書き換え帰納法では $s \equiv t$ の場合にしか除去できなかったが，拡張書き換え帰納法では $s \overset{*}{\leftrightarrow}_G t$ であれば除去できるように拡張されている．*Expand2* 規則は，*Expand* 規則では展開するのは一方の辺のみだったのを，両辺を展開するように変更した規則である．また，*Expand* 規則では $s \succ t$ ($\Leftrightarrow s \succsim t \wedge s \not\precsim t$) だった条件が *Expand2* 規則では $s \approx t$ ($\Leftrightarrow s \succsim t \wedge s \precsim t$) となっている．また，Expd2 の定義では，項書き換えシステム $\mathcal{R}$ が構成子システムであることより，基本部分項 u について $\sigma = \mathrm{mgu}(u, l)$ は構成子代入であり，従って $v\sigma \in \mathcal{B}(t\sigma)$ であることに注意する．

例 13 (*Expand2* 規則の適用例)

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{plus}(0, y) & \to & y \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\},$$

$s \equiv \mathsf{plus}(x,y)$, $t \equiv \mathsf{plus}(y,x)$ とする．このとき，

$$\mathrm{Expd2}_{s,t}(s,t) \quad = \quad \left\{ \begin{array}{l} \mathsf{0} \doteq \mathsf{0} \\ \mathsf{s}(x_1) \doteq \mathsf{s}(\mathsf{plus}(x_1,\mathsf{0})) \\ \mathsf{s}(\mathsf{plus}(x_3,\mathsf{s}(y_3))) \doteq \mathsf{s}(\mathsf{plus}(y_3,\mathsf{s}(x_3))) \end{array} \right\}.$$

ここで，等式 $\mathsf{s}(\mathsf{plus}(x_2,\mathsf{0})) \doteq \mathsf{s}(x_2)$ も $\mathrm{Expd2}_{s,t}(s,t)$ に含まれるが，これは 2 番目の等式と (等式として) 同一なので省かれていることに注意する． □

補題 14 (Expd2 の性質) $\mathcal{R}$ を擬簡約な構成子システムとし，$u \in \mathcal{B}(s)$, $v \in \mathcal{B}(t)$ とする．
(1) 任意の基底構成子代入 σ_g について $s\sigma_g \to_{\mathcal{R}} \circ \leftrightarrow_{\mathrm{Expd2}_{u,v}(s,t)} \circ \leftarrow_{\mathcal{R}} t\sigma_g$ ．
(2) $w \leftrightarrow_{\mathrm{Expd2}_{u,v}(s,t)} w' \Rightarrow w \overset{*}{\leftrightarrow}_{\mathcal{R}\cup\{s\doteq t\}} w'$ ．

証明． (1) $l \to r \in \mathcal{R}$, $\sigma = \mathrm{mgu}(u,l)$, $s \equiv C[u]$ とおく．このとき，補題 3 (1) より $s\sigma_g \to_{\mathcal{R}} C[r]\sigma\theta_g \leftrightarrow_{\{C[r]\sigma\doteq t\sigma\}} t\sigma\theta_g \equiv t\sigma_g$ なる $\langle C[r]\sigma, t\sigma\rangle \in \mathrm{Expd}_u(s,t)$, 基底項代入 θ_g が存在する．また，l,u が基本項であることより σ は構成子代入．ゆえに，$v \in \mathcal{B}(t)$ より $v\sigma \in \mathcal{B}(t\sigma)$．よって，もう 1 度補題 3 (1) を用いると，$t\sigma\theta_g \to_{\mathcal{R}} \circ \leftrightarrow_{\mathrm{Expd}_{v\sigma}(t\sigma,C[r]\sigma)} C[r]\sigma\theta_g$．ゆえに $s\sigma_g \to_{\mathcal{R}} C[r]\sigma\theta_g \leftrightarrow_{\mathrm{Expd2}_{u,v}(s,t)} \circ \leftarrow_{\mathcal{R}} t\sigma\theta_g \equiv t\sigma_g$．(2) $w \leftrightarrow_{\mathrm{Expd2}_{u,v}(s,t)} w'$ とする．このとき定義より，$w \leftrightarrow_{\mathrm{Expd}_{v\sigma}(t\sigma,s')} w'$ なる $\langle s', t\sigma\rangle \in \mathrm{Expd}_u(s,t)$ が存在する．ただし，ここで $\sigma = \mathrm{mgu}(l,u)$, $l \to r \in \mathcal{R}$ とする．仮定より σ は構成子代入であるから，$v\sigma \in \mathcal{B}(t\sigma)$ が成立する．すると補題 3 (2) より $w \overset{*}{\leftrightarrow}_{\mathcal{R}\cup\{s'\doteq t\sigma\}} w'$. 従って，$w \overset{*}{\leftrightarrow}_{\mathcal{R}\cup\mathrm{Expd}_u(s,t)} w'$. ここで補題 3 (2) をもう 1 度適用することにより $w \overset{*}{\leftrightarrow}_{\mathcal{R}\cup\{s\doteq t\}} w'$ ． □

以下では拡張書き換え帰納法の正当性を示す．以降，本節の補題では拡張書き換え帰納法のステップは擬簡約な構成子システム $\mathcal{R}$ と $\mathcal{R} \subseteq \succ$ なる簡約擬順序 $\succsim$ に対して実行されていると仮定する．

補題 15 (推論ステップの不変量) $\langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ とする．基底項集合上で，$\overset{*}{\leftrightarrow}_{\mathcal{R}\cup E_n\cup H_n\cup G_n} = \overset{*}{\leftrightarrow}_{\mathcal{R}\cup E_{n+1}\cup H_{n+1}\cup G_{n+1}}$ ．

証明. $\langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ において適用された推論規則の種類により場合分けを行う．$\langle E_n, H_n, G_n\rangle \rightsquigarrow^{s,d} \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ のときは明らか．$\langle E_n, H_n, G_n\rangle \rightsquigarrow^{e} \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ のとき．このとき，$E_n = E \uplus \{s = t\}$, $E_{n+1} = E \cup \mathrm{Expd}_u(s,t)$, $H = H_n$, $H_{n+1} = H \cup \{s \to t\}$, $s \succ t$, $u \in \mathcal{B}(s)$, $G = G_n = G_{n+1}$ とおける．$\overset{*}{\leftrightarrow}_{\mathcal{R}\cup E_n\cup H_n\cup G_n} \subseteq \overset{*}{\leftrightarrow}_{\mathcal{R}\cup E_{n+1}\cup H_{n+1}\cup G_{n+1}}$ は明らか．逆の包含関係を示すには，$u_g \leftrightarrow_{\mathrm{Expd}_u(s,t)} v_g \Rightarrow u_g \overset{*}{\leftrightarrow}_{\mathcal{R}\cup E_n\cup H_n} v_g$ を示せば充分．これは $s \doteq t \in E_n$ より補題 3 (2) から導かれる．$\langle E_n, H_n, G_n\rangle \rightsquigarrow^{e2} \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ のときも補題 14 (2) を用いて同様に示される． □

補題 16 (E_n の性質) $\langle E_n, H_n, G_n\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ とする． 基底項集合上で， $\leftrightarrow_{E_n} \subseteq \overset{*}{\to}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \overset{*}{\leftarrow}_{(\mathcal{R}\cup H^\sharp)/G^\sharp}$ ．

証明. $\langle E_n, H_n, G_n\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ の長さ k に関する帰納法．$k = 0$ のときは $E_n = \emptyset$ より明らか．$k > 0$ のとき，$\langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ とおける．$\langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ において適用された推論規則による場合分けにより，任意の基底項 s_g, t_g について，$s_g \leftrightarrow_{E_n} t_g \Rightarrow s_g \overset{*}{\to}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \overset{*}{\leftarrow}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} t_g$ を示す．

1. $\langle E_n, H_n, G_n\rangle \rightsquigarrow^{s} \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ の場合．このとき，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E \cup \{s' \doteq t\}$, $H = H_n = H_{n+1}$, $G = G_n = G_{n+1}$, $s \to_{(\mathcal{R}\cup H)/G} s'$ とおける．

$s_g \leftrightarrow_E t_g$ のときは帰納法の仮定から明らか．$s_g \leftrightarrow_{\{s \doteq t\}} t_g$ とする．このとき，$s_g \equiv C_g[s\sigma_g]$, $t_g \equiv C_g[t\sigma_g]$ (または $s_g \equiv C_g[t\sigma_g]$, $t_g \equiv C_g[s\sigma_g]$) とおける．帰納法の仮定より $C_g[s'\sigma_g] \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} C_g[t\sigma_g] \equiv t_g$ ．ゆえに，$s_g \equiv C_g[s\sigma_g] \rightarrow_{(\mathcal{R}\cup H)/G} C_g[s'\sigma_g]$, $H \subseteq H^\sharp$, $G \subseteq G^\sharp$ より，$s_g \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} t_g$ ．($s_g \equiv C_g[t\sigma_g]$, $t_g \equiv C_g[s\sigma_g]$ の場合も同様．)

2. $\langle E_n, H_n, G_n\rangle \rightsquigarrow^d \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ の場合．このとき，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E$, $H = H_n = H_{n+1}$, $G = G_n = G_{n+1}$, $s \overset{*}{\leftrightarrow}_G t$ とおける．$s_g \leftrightarrow_E t_g$ のときは帰納法の仮定から明らか．$s_g \leftrightarrow_{\{s \doteq t\}} t_g$ のときは，$s_g \equiv C_g[s\sigma_g] \overset{*}{\leftrightarrow}_G C_g[t\sigma_g] \equiv t_g$ より明らか．

3. $\langle E_n, H_n, G_n\rangle \rightsquigarrow^e \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ の場合．このとき，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E \cup \mathrm{Expd}_u(s,t)$, $H = H_n$, $H_{n+1} = H \cup \{s \rightarrow t\}$, $u \in \mathcal{B}(s)$, $s \succ t$, $G = G_n = G_{n+1}$ とおける．$s_g \leftrightarrow_E t_g$ のときは帰納法の仮定から明らか．$s_g \leftrightarrow_{\{s \doteq t\}} t_g$ のときは，$s \rightarrow t \in H_{n+1} \subseteq H^\sharp$ より明らか．

4. $\langle E_n, H_n, G_n\rangle \rightsquigarrow^{e2} \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle$ の場合．このとき，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E \cup \mathrm{Expd2}_{u,v}(s,t)$, $H = H_n = H_{n+1}$, $G = G_n$, $G_{n+1} = G \cup \{s \doteq t\}$, $u \in \mathcal{B}(s)$, $v \in \mathcal{B}(t)$, $s \approx t$ とおける．$s_g \leftrightarrow_E t_g$ のときは帰納法の仮定から明らか．$s_g \leftrightarrow_{\{s \doteq t\}} t_g$ のときは，$s \doteq t \in G_{n+1} \subseteq G^\sharp$ より明らか． □

補題 17 ($H^\sharp$ の性質) $\langle E_n, H_n, G_n\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ とする．基底項集合上で，$\rightarrow_{H^\sharp} \subseteq \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp}$ ．

証明. $s \rightarrow t \in H^\sharp$, $s_g \rightarrow_{\{s \rightarrow t\}} t_g$ とする．また，$s_g \equiv C_g[s\sigma_g]$, $t_g \equiv C_g[t\sigma_g]$ とおく．

このとき，任意の $x \in V(s)$ について，$\sigma_g(x) \in \mathrm{NF}(\rightarrow_{\mathcal{R}})$ としてよい．なぜなら，$\mathrm{SN}(\rightarrow_{\mathcal{R}})$ より，任意の $x \in V(s)$ に対して $\sigma_g(x) \xrightarrow{*}_{\mathcal{R}} \hat{\sigma}_g(x) \in \mathrm{NF}(\rightarrow_{\mathcal{R}})$ となる代入 $\hat{\sigma}_g$ が存在し，$C_g[s\hat{\sigma}_g] \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} C_g[t\hat{\sigma}_g]$ が言えれば，$C_g[s\sigma_g] \xrightarrow{*}_{\mathcal{R}} C_g[s\hat{\sigma}_g]$，$C_g[t\sigma_g] \xrightarrow{*}_{\mathcal{R}} C_g[t\hat{\sigma}_g]$ より $C_g[s\sigma_g] \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} C_g[t\sigma_g]$ が言えるからである．

そこで，任意の $x \in V(s)$ について，$\sigma_g(x) \in \mathrm{NF}(\rightarrow_{\mathcal{R}})$ とする．すると，$\mathcal{R}$ の擬簡約性より σ_g は構成子代入．また，推論規則の形より $\langle E_0, \emptyset, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$, $H_{n+1} = H_n \cup \{s \rightarrow t\}$ となるある n が存在し，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E \cup \mathrm{Expd}_u(s,t)$, $u \in \mathcal{B}(s)$, $s \succ t$ とおける．すると，補題 3 (1) より $s\sigma_g \rightarrow_{\mathcal{R}} \circ \leftrightarrow_{\mathrm{Expd}_u(s,t)} t\sigma_g$ ．ゆえに $s_g \equiv C_g[s\sigma_g] \rightarrow_{\mathcal{R}} \circ \leftrightarrow_{E_{n+1}} C_g[t\sigma_g] \equiv t_g$ ．従って，補題 16 より $s_g \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} t_g$ ． □

補題 18 ($G^\sharp$ の性質) $\langle E_n, H_n, G_n\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ とする．基底項集合上で，$\leftrightarrow_{G^\sharp} \subseteq \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \leftarrow_{\mathcal{R}}$ ．

証明. $s \doteq t \in G^\sharp$, $s_g \leftrightarrow_{\{s \doteq t\}} t_g$ とする．また，$s_g \equiv C_g[s\sigma_g]$, $t_g \equiv C_g[t\sigma_g]$ とおく．

補題 17 の場合と同様にして任意の $x \in V(s) \cup V(t)$ について $\sigma_g(x) \in \mathrm{NF}(\rightarrow_{\mathcal{R}})$ としてよい．推論規則の形より $\langle E_0, \emptyset, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle E_n, H_n, G_n\rangle \rightsquigarrow \langle E_{n+1}, H_{n+1}, G_{n+1}\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$, $G_{n+1} = G_n \cup \{s \doteq t\}$ となるある n が存在し，$E_n = E \uplus \{s \doteq t\}$, $E_{n+1} = E \cup \mathrm{Expd2}_{u,v}(s,t)$, $u \in \mathcal{B}(s)$, $v \in \mathcal{B}(t)$, $s \approx t$ とおける．すると，補題 14 (1) より $s\sigma_g \rightarrow_{\mathcal{R}} \circ \leftrightarrow_{\mathrm{Expd2}_{u,v}(s,t)} \circ \leftarrow_{\mathcal{R}} t\sigma_g$ ．ゆえに $s_g \equiv C_g[s\sigma_g] \rightarrow_{\mathcal{R}} \circ \leftrightarrow_{E_{n+1}} \circ \leftarrow_{\mathcal{R}} C_g[t\sigma_g] \equiv t_g$ ．従って，補題 16 より $s_g \rightarrow_{\mathcal{R}} \circ \xrightarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \overset{*}{\leftrightarrow}_{G^\sharp} \circ \xleftarrow{*}_{(\mathcal{R}\cup H^\sharp)/G^\sharp} \circ \leftarrow_{\mathcal{R}} t_g$ ． □

定理 19 (拡張書き換え帰納法の正当性) $\mathcal{R}$ を擬簡約な構成子システム，E を等式集合，$\succsim$ を $\mathcal{R} \subseteq \succ$ なる簡約擬順序とする．ある項書き換え規則集合 $H^\sharp$，等式集合 $G^\sharp$ が存在して，$\langle E, \emptyset, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ となるならば，E 中の等式は $\mathcal{R}$ の帰納的定理である．

証明. 補題 15 を繰り返し適用すると，$\langle E, \emptyset, \emptyset\rangle \overset{*}{\rightsquigarrow} \langle \emptyset, H^\sharp, G^\sharp\rangle$ より，基底項集合上で $\overset{*}{\leftrightarrow}_{E\cup\mathcal{R}} = \overset{*}{\leftrightarrow}_{\mathcal{R}\cup H^\sharp\cup G^\sharp}$．ゆえに，基底項集合上で $\overset{*}{\leftrightarrow}_{\mathcal{R}\cup H^\sharp\cup G^\sharp} = \overset{*}{\leftrightarrow}_{\mathcal{R}}$ であることを示せばよい．抽象書き換えシステム $\langle \mathrm{T}(\mathcal{F}), \rightarrow_{\mathcal{R}}\rangle$, $\langle \mathrm{T}(\mathcal{F}), \rightarrow_{H^\sharp}\rangle$, $\langle \mathrm{T}(\mathcal{F}), \rightarrow_{G^\sharp}\rangle$，擬順序 $\succsim$ に対して，補題 11 を適用することを考える．$\mathcal{R}\cup H^\sharp \subseteq \succ$, $G^\sharp \subseteq \approx$ より補題 11 の条件 (i),(ii) が成立する．また，補題 17 より条件 (iii)，補題 18 より条件 (iv) が成立する．また，推論規則 $Expand2$ の適用条件から，任意の $s \doteq t \in G^\sharp$ に対して $\mathcal{B}(s) \neq \emptyset$ かつ $\mathcal{B}(t) \neq \emptyset$．よって，$\mathcal{R}$ の擬簡約性より任意の s_g, t_g に対して $s_g \overset{+}{\leftrightarrow}_{G^\sharp} t_g \Rightarrow s_g, t_g \notin \mathrm{NF}(\rightarrow_{\mathcal{R}})$．ゆえに条件 (v) も成立する．以上より補題 11 から $\overset{*}{\leftrightarrow}_{\mathcal{R}} = \overset{*}{\leftrightarrow}_{\mathcal{R}\cup H^\sharp\cup G^\sharp}$． □

例 20 (拡張書き換え帰納法の実行例) 入力項書き換えシステム $\mathcal{R}$ と証明する等式集合 E として以下が与えられたとする．

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{plus}(\mathsf{0}, y) & \rightarrow & y \\ \mathsf{plus}(\mathsf{s}(x), y) & \rightarrow & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

$$E = \left\{ \begin{array}{lcl} \mathsf{plus}(x, y) & \doteq & \mathsf{plus}(y, x) \end{array} \right\}$$

以下は，$\succsim$ を優位順序 $\mathsf{plus} \succ \mathsf{s} \succ \mathsf{0}$ に基づく再帰経路順序としたときの拡張書き換え帰納法の実行過程の例である．

$$\left\langle \left\{ \begin{array}{l} \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \end{array} \right\}, \left\{\right\}, \left\{\right\} \right\rangle$$

$$\rightsquigarrow^{e} \quad \left\langle \begin{array}{l} \left\{ \begin{array}{l} \mathsf{0} \doteq \mathsf{0},\ \mathsf{s}(x_1) \doteq \mathsf{s}(\mathsf{plus}(x_1, \mathsf{0})) \\ \mathsf{s}(\mathsf{plus}(x_2, \mathsf{s}(y_2))) \doteq \mathsf{s}(\mathsf{plus}(y_2, \mathsf{s}(x_2))) \end{array} \right\} \\ \left\{\right\}, \left\{ \begin{array}{l} \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \end{array} \right\} \end{array} \right\rangle$$

$$\rightsquigarrow^{d}\rightsquigarrow^{s}\rightsquigarrow^{d} \quad \left\langle \begin{array}{l} \left\{ \begin{array}{l} \mathsf{s}(\mathsf{plus}(x_2, \mathsf{s}(y_2))) \doteq \mathsf{s}(\mathsf{plus}(y_2, \mathsf{s}(x_2))) \end{array} \right\} \\ \left\{\right\}, \left\{ \begin{array}{l} \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \end{array} \right\} \end{array} \right\rangle$$

$$\rightsquigarrow^{s}\rightsquigarrow^{s}\rightsquigarrow^{d} \quad \left\langle \left\{\right\}, \left\{\right\}, \left\{ \begin{array}{l} \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \end{array} \right\} \right\rangle$$

このとき，定理 19 より E の等式は $\mathcal{R}$ の帰納的定理である． □

5 プログラム変換への応用例

関数定義を等式により記述したものを E に入れ，書き換え帰納法を用いて帰納的定理証明を行うと，あるところで書き換え帰納法の手続きが停止して証明に失敗する．このとき，E に残っている等式をもとの関数定義をプログラム変換したものをみなすこととする．これを書き換え帰納法によるプログラム変換とよぶ．書き換え帰納法によるプログラム変換は，プログラム融合変換 [17] と同様な結果を得ることが出来る場合があるばかりでなく，途中で変換に必要な補題の証明を行いながら変換を進めることが出来るため，より強力な変換にも成功する場合がある [13]．

以下に拡張した書き換え帰納法に基づいてプログラム変換を行った例を示す．

例 **21** (書き換え帰納法によるプログラム融合変換)

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{plus}(\mathsf{0}, y) & \to & y \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

$$E = \left\{ \begin{array}{lcl} \mathsf{plus}(x, y) & \doteq & \mathsf{plus}(y, x) \\ \mathsf{h}(x) & \doteq & \mathsf{plus}(x, x) \end{array} \right\}$$

とする．以下は，$\succsim$ を優位順序 $\mathsf{plus} \succ \mathsf{h} \succ \mathsf{s} \succ \mathsf{0}$ に基づく再帰経路順序としたときの拡張書き換え帰納法の実行過程の例である．

$$\left\langle \left\{ \begin{array}{l} \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \\ \mathsf{h}(x) \doteq \mathsf{plus}(x, x) \end{array} \right\}, \{\}, \{\} \right\rangle$$

$$\overset{*}{\rightsquigarrow} \quad \left\langle \left\{ \mathsf{h}(x) \doteq \mathsf{plus}(x, x) \right\}, \left\{ \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \right\} \right\rangle$$

$$\rightsquigarrow^{e} \quad \left\langle \left\{ \begin{array}{l} \mathsf{h}(\mathsf{0}) \doteq \mathsf{0} \\ \mathsf{h}(\mathsf{s}(x_1)) \doteq \mathsf{s}(\mathsf{plus}(x_1, \mathsf{s}(x_1))) \end{array} \right\}, \left\{ \begin{array}{l} \mathsf{plus}(x, x) \to \mathsf{h}(x) \end{array} \right\}, \left\{ \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \right\} \right\rangle$$

$$\rightsquigarrow^{s}\rightsquigarrow^{s} \quad \left\langle \left\{ \begin{array}{l} \mathsf{h}(\mathsf{0}) \doteq \mathsf{0} \\ \mathsf{h}(\mathsf{s}(x_1)) \doteq \mathsf{s}(\mathsf{s}(\mathsf{h}(x_1))) \end{array} \right\}, \left\{ \begin{array}{l} \mathsf{plus}(x, x) \to \mathsf{h}(x) \end{array} \right\}, \left\{ \mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x) \right\} \right\rangle$$

このとき，残った等式集合から得られる項書き換えシステム

$$\left\{ \begin{array}{lcl} \mathsf{h}(\mathsf{0}) & \to & \mathsf{0} \\ \mathsf{h}(\mathsf{s}(x_1)) & \to & \mathsf{s}(\mathsf{s}(\mathsf{h}(x_1))) \end{array} \right\}$$

は，最初に入力した等式から得られる項書き換えシステム $\{\mathsf{h}(x) \to \mathsf{plus}(x, x)\}$ と等価な項書き換えシステムになっている． □

次に，別の応用例として，パターンに基づくプログラム変換 [4] による例を示す．パターンに基づくプログラム変換では，与えられた変換パターンに基づいて項書き換えシステムを変換する．このとき，変換の正当性を保証するために，変換パターンを構成する仮定集合の具体化が入力項書き換えシステムの帰納的定理となる必要がある．

例 **22** (パターンに基づくプログラム変換) 変換パターン $\langle \mathcal{P}, \mathcal{P}', \mathcal{H} \rangle$ を以下で与える．

$$\mathcal{P} = \left\{ \begin{array}{lcl} \mathsf{f}(\mathsf{a}) & \to & \mathsf{b} \\ \mathsf{f}(\mathsf{c}(u, v)) & \to & \mathsf{g}(\mathsf{f}(v), u) \\ \mathsf{g}(\mathsf{b}, u) & \to & u \\ \mathsf{g}(\mathsf{d}(u, v), w) & \to & \mathsf{d}(u, \mathsf{g}(v, w)) \end{array} \right\}, \mathcal{P}' = \left\{ \begin{array}{lcl} \mathsf{f}(u) & \to & \mathsf{f}_1(u, \mathsf{b}) \\ \mathsf{f}_1(\mathsf{a}, u) & \to & u \\ \mathsf{f}_1(\mathsf{c}(u, v), w) & \to & \mathsf{f}_1(v, \mathsf{g}(w, u)) \\ \mathsf{g}(\mathsf{b}, u) & \to & u \\ \mathsf{g}(\mathsf{d}(u, v), w) & \to & \mathsf{d}(u, \mathsf{g}(v, w)) \end{array} \right\},$$

$$\mathcal{H} = \left\{ \begin{array}{lcl} \mathsf{g}(u, v) & \doteq & \mathsf{g}(v, u) \\ \mathsf{g}(\mathsf{g}(u, v), w) & \doteq & \mathsf{g}(u, \mathsf{g}(v, w)) \end{array} \right\}.$$

このとき，この変換パターンにより，次の項書き換えシステム $\mathcal{R}$ の変換を行う．

$$\mathcal{R} = \left\{ \begin{array}{lcl} \mathsf{sum}(\mathsf{nil}) & \to & \mathsf{0} \\ \mathsf{sum}(\mathsf{cons}(x, xs)) & \to & \mathsf{plus}(\mathsf{sum}(xs), x) \\ \mathsf{plus}(\mathsf{0}, x) & \to & x \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

このとき，変換出力として，

$$\mathcal{R}' = \left\{ \begin{array}{lcl} \mathsf{sum}(xs) & \to & \mathsf{sum1}(xs, \mathsf{0}) \\ \mathsf{sum1}(\mathsf{nil}, x) & \to & x \\ \mathsf{sum1}(\mathsf{cons}(x, xs), y) & \to & \mathsf{sum1}(xs, \mathsf{plus}(y, x)) \\ \mathsf{plus}(\mathsf{0}, x) & \to & x \\ \mathsf{plus}(\mathsf{s}(x), y) & \to & \mathsf{s}(\mathsf{plus}(x, y)) \end{array} \right\}$$

が得られる．このプログラム変換の正当性を保証するために，$\mathcal{H}$ の具体化

$$E = \left\{ \begin{array}{lcl} \mathsf{plus}(x, y) & \doteq & \mathsf{plus}(y, x) \\ \mathsf{plus}(\mathsf{plus}(x, y), z) & \doteq & \mathsf{plus}(x, \mathsf{plus}(y, z)) \end{array} \right\}$$

が $\mathcal{R}$ の帰納的定理となっていることが必要となる．$\mathsf{plus}(x, y) \doteq \mathsf{plus}(y, x)$ の方は再帰経路順序に基づく拡張書き換え帰納法 (例 20) で，$\mathsf{plus}(\mathsf{plus}(x, y), z) \doteq \mathsf{plus}(x, \mathsf{plus}(y, z))$ の方は辞書式経路順序に基づく (拡張) 書き換え帰納法 (例 6) で証明できる． □

6 まとめ

本論文では，書き換え帰納法を拡張し，簡約順序によって向き付けできないような帰納的定理を扱えるようにした拡張書き換え帰納法を提案した．まず，文献 [10] により定式化された抽象書き換えシステムにおける書き換え帰納法の原理を拡張した．そして，拡張した書き換え帰納法の原理に基づく拡張書き換え帰納法の手続きを与え，その正当性を示した．最後に，拡張した書き換え帰納法を項書き換えシステムに基づくプログラム変換へ応用する例を示した．本論文で提案する拡張書き換え帰納法を文献 [1] のように単純型付き項書き換えシステム [18] (S 式書き換えシステム [16]) といった高階の項書き換えシステムへ拡張することも容易に可能である．

現在，本論文で提案した拡張書き換え帰納法の実装および実験を進めている．本手法の有効性および問題点を実験により調べることは今後の課題である．また，本論文で提案した拡張書き換え帰納法では，簡約擬順序が両方向とも成立しない，つまり，両辺が比較不能となるような等式は取り扱うことが出来ない．このような場合が取り扱えるように書き換え帰納法を拡張することも今後の研究課題である．

7 謝辞

丁寧なコメントを下さった査読者の方々および関連研究を御指摘下さった外山芳人教授に感謝致します．この研究は一部日本学術振興会科学研究費 14580357 および文部科学省科学研究費 17700002,16016202 の補助を受けて行われた．

参考文献

[1] T. Aoto, T. Yamada, and Y. Toyama. Proving inductive theorems of higher-order functional programs. In *Proceedings of the 14th International Conference on Rewriting Techniques and Applications*, Vol. 3901 of *LNCS*, pp. 269–284. Springer-Verlag, 2004.

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

[3] A. Bouhoula, E. Kounalis, and M. Rusinowitch. Automated mathematical induction. *Journal of Logic and Computation*, 5(5):631–668, 1995.

[4] Y. Chiba, T. Aoto, and Y. Toyama. Program transformation by templates based on term rewriting. In *Proceedings of the 7th ACM-SIGPLAN International Conference on Principles and Practice of Declarative Programming*, pp. 59–69. ACM Press, 2005.

[5] N. Dershowitz and U. S. Reddy. Deductive and inductive synthesis of equational programs. *Journal of Symbolic Computation*, 15:467–494, 1993.

[6] G. Huet. Confluent reductions: abstract properties and applications to term rewriting systems. *Journal of the Association for Computing Machinery*, 27(4):797–821, 1980.

[7] G. Huet and J.-M. Hullot. Proof by induction in equational theories with constructors. *Journal of Computer and System Sciences*, 25(2):239–266, 1982.

[8] J.-P. Jouannaud and E. Kounalis. Automatic proofs by induction in theories without constructors. *Information and Computation*, 82:1–33, 1989.

[9] D. Kapur, P. Narendran, and H. Zhang. Automating inductionless induction using test sets. *Journal of Symbolic Computation*, 11(1–2):81–111, 1991.

[10] 小池宏高, 外山芳人. 潜在帰納法と書き換え帰納法. コンピュータソフトウェア, 17(6):1–12, 2000.

[11] D. R. Musser. On proving inductive properties of abstract data types. In *Proceedings of the 7th Annual ACM Symposium on Principles of Programming Languages*, pp. 154–162. ACM Press, 1980.

[12] U. S. Reddy. Term rewriting induction. In *Proceedings of the 10th International Conference on Automated Deduction*, Vol. 449 of *LNAI*, pp. 162–177. Springer-Verlag, 1990.

[13] 坂本邦彦, 青戸等人, 外山芳人. 書き換え帰納法に基づくプログラム融合変換. 日本ソフトウェア科学会第 21 回大会論文集, 2B-3, 2004.

[14] Terese. *Term Rewriting Systems*. Cambridge University Press, 2003.

[15] Y. Toyama. How to prove equivalence of term rewriting systems without induction. *Theoretical Computer Science*, 90(2):369–390, 1991.

[16] Y. Toyama. Termination of S-expression rewriting systems: Lexicographic path ordering for higher-order terms. In *Proceedings of the 15th International Conference on Rewriting Techniques and Applications*, Vol. 3091 of *LNCS*, pp. 40–54. Springer-Verlag, 2004.

[17] P. Wadler. Deforestation: transforming programs to eliminate trees. *Theoretical Computer Science*, 73:231–248, 1990.

[18] T. Yamada. Confluence and termination of simply typed term rewriting systems. In *Proceedings of the 12th International Conference on Rewriting Techniques and Applications*, Vol. 2051 of *LNCS*, pp. 338–352. Springer-Verlag, 2001.