TR-0784

等式論理の帰納的定理を証明する手続き

大須賀　昭彦（東芝）、坂井　公（筑波大）

July, 1992

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191～5
Telex ICOT J32964

Institute for New Generation Computer Technology

等式論理の帰納的定理を証明する手続き

大須賀 昭彦
(株) 東芝
システム・ソフトウェア技術研究所
ohsuga@ssel.toshiba.co.jp

坂井 公
筑波大学
電子・情報工学系
sakai@iris.is.tsukuba.ac.jp

概 要

等式論理の始モデルにおいて成立する式を等式論理の帰納的定理と呼ぶ．本稿では，帰納的定理を証明するための推論規則を与える．この推論規則に基づく証明手続きは，潜在帰納法と呼ばれる証明方法の拡張とみることができるが，向きづけのできない公理や定理を一般的に扱うことができ，さらには，公理系に対応する完備な項書換え系が有限の要素で表現できない場合でも手続きを適用できる点が，従来の方法と異なる．推論規則は，反駁的な意味での完全性を持つ．つまり，定理が正しくないときには，証明手続きによって常に反証が得られる．

1 はじめに

抽象データ型を記述するための代数的仕様記述法や関数プログラミング，または制約論理プログラミングなど，計算機科学の比較的新しい分野において，等式に論理的基礎をおく仕様記述言語やプログラミング言語などが多く見受けられる．このような言語系の具体的な記述が持つ性質は，多くの場合，等式論理の帰納的定理として扱うことができる．つまり，帰納的定理を証明する手続きによって，仕様やプログラムの検証，仕様とプログラムの等価性判定などを自動的に行うことが可能となる．

Goguen[6] や Musser[11] らは，等式論理における帰納的定理の成立が，公理系に定理を追加して得られる新たな公理系の無矛盾性と等価なことを示し，Knuth-Bendix の完備化手続き [10] がその無矛盾性の判定に利用できることを示唆した．彼らの方法が全てを一種類の矛盾 ($true = false$) の判定に帰着させていたのに対し，Huet と Hullot は関数記号を構成子と演算子に分割し，種々の矛盾を導入することで，矛盾の検出を容易にした [9]．Fribourg は，Huet の方法を線形に実行しても完全性が失われず，しかも効率が大幅に向上することを示した [5]．Bachmair は Fribourg の方法を拡張して，定理の向きづけを必要としない証明手続きを提案し，その反駁完全性を証明順序法を用いて示した [2]．一般に帰納法を必要とするこの種の定理証明に対し，帰納法を用いないこれらの手続きは，潜在帰納法 (inductionless induction) または無矛盾による証明 (proof by consistency) などと呼ばれている．

潜在帰納法は，帰納的定理を自動証明できる強力な手続きである反面，いくつかの問題点を持つ．E を等式論理の公理系とすると，潜在帰納法を用いて E の帰納的定理を証明する際には，E と等価な項書換え系 R で (1) 有限の書換え規則からなり，(2) 停止性を持つものの存在が前提となる．また，証明手続き中も (3) 定理から導出される等式の向きづけができない，(4) 等式が導出され続けて証明手続きが停止しないなどの原因で，証明が失敗に終ることがある．(3) は文献 [2] においても解決されているが，ここでは (4) を除く (1)〜(3) の問題を統一的な枠組の中で解決する．つまり提案する方法は，得られる項書換え系が停止性を持つように向きづけのできない公理や定理も扱うことができ，さらには，公理系に対応する完備な項書換え系が有限の要素で表現できない場合でも手続きを適用できる点

において，従来の方法と異なっている．

本稿では，2節で帰納的定理の定義を行い，3節で定理証明の考え方を説明する．定理証明で用いる項書換え技術を4節で導入した後，5節で推論規則を与え，これによる証明例を6節で示す．7節では推論規則の完全性を証明し，最後に8節で手続きの実現について考察する．簡単のため，本稿においては単一ソート上で話を展開するが，多ソートへの拡張は容易である．

2 等式論理の帰納的定理

本節では用語と記法を導入した後，帰納的定理の定義を行う．ここで説明しない基本的な事柄については，文献[8]などを参照されたい．

F を**関数記号** (function symbol) の有限集合，V を**変数** (variable) の可算無限集合とする．$T(F,V)$ によって F と V から構成される**項** (term) の集合を表し，$T(F)$ によって F のみから構成される（つまり，変数を持たない）**基礎項** (ground term) の集合を表す．引数の個数は関数記号毎に固定されているものとし，定数は引数をとらない関数と考える．部分項 (subterm) として s を持つ項を $t[s]$ と書き，この s を u へ置き換えた項を $t[u]$ によって表す．項 t に代入 θ を適用した結果を $t\theta$ で表し，t の**例** (instance) と呼ぶ．$t\theta \in T(F)$ となるとき，特に $t\theta$ を t の**基礎例** (ground instance) と呼び，基礎例を与える代入 θ を**基礎代入** (ground substitution) と呼ぶ．

$l = r$ で表される2項の対を**等式** (equation) と呼び，等式の集合を**等号系** (equational system) と呼ぶ．E を等号系とするとき，$\Leftrightarrow_E$ によって E の対称かつ安定な閉包を表す．つまり，$t \Leftrightarrow_E u$ であるのは，E の等式 $l \cong r$，代入 θ，項 c が存在して，$t \equiv c[l\theta]$ かつ $u \equiv c[r\theta]$ となるとき，かつ，このときに限られる．ここで，$l \cong r$ は $l = r$ または $r = l$ の略記で，$l \equiv r$ は l と r の構文的な一致を表す．$\Leftrightarrow_E$ の反射推移閉包を $\dot{\Leftrightarrow}_E$ で表すと，$\dot{\Leftrightarrow}_E$ は $T(F,V)$ 上の合同関係である．E の**始モデル** (initial model) とは，$T(F)$ の $\dot{\Leftrightarrow}_E$ による商代数 $T(F) / \dot{\Leftrightarrow}_E$ をいう．

定義 2.1 (帰納的定理)　E を等号系とする．E の始モデル上で恒に成立する等式を E の**帰納的定理** (inductive theorem) と呼ぶ．つまり，等式 $l = r$ が E の帰納的定理であるとは，任意の基礎代入 θ について $l\theta \dot{\Leftrightarrow}_E r\theta$ となることをいう■

本稿では，等式に現われる変数はすべて全称束縛されていると考えるので，異なる等式の間に変数の共有はないものと仮定する (共有がある場合，暗黙のうちに新しい変数で置き換える)．

3 矛盾による帰納的定理の証明

本節では帰納的定理証明の原理を説明する．

E を等号系とする．等式 $l = r$ が基礎代入 θ によって E と**帰納的に矛盾する** (inductively inconsistent) とは，$l\theta \not\dot{\Leftrightarrow}_E r\theta$ となることをいう．定義より明らかに，$l = r$ が帰納的に矛盾するとき，かつ，このときに限り，$l = r$ は E の帰納的定理でない．つまり，帰納的な矛盾性を判定できれば帰納的定理の証明が可能となる．

帰納的な矛盾を検出するために，文献[9]と同様の(自由)構成子を導入する．F は**演算子** (defined symbol) の集合 D と，**構成子** (constructor) の集合 C に分割されているものとする．演算子を含まない項を**構成子項** (constructor term) と呼び，少なくとも1つの演算子を含む項を**非構成子項** (non-constructor term) と呼ぶ．定義より，変数そのものも構成子項である．本稿では，変数を持たない構成子項(基礎構成子項と呼ぶ)が常に存在し，C には少なくとも2つの構成子が含まれることを仮定する．

文献[9]の定義原理 (principle of definition) に対応する概念として，帰納的完全性を定義する．等号系 E は以下の条件を満たすとき，**帰納的に完全** (inductively complete) である．

(C1) 任意の基礎項 t について $t \dot{\Leftrightarrow}_E u$ なる基礎構成子項 u が存在する．かつ，

(C2) 任意の基礎構成子項 t, u について $t \dot{\Leftrightarrow}_E u$ となるのは $t \equiv u$ のときに限られる．

帰納的な完全性は，基礎構成子項の表現が始モデルに対する冗長性や曖昧性を持たないことを保証する自然な条件である．この判定方法については後述する．帰納的に完全な E の下で，2項 l, r が異なる基礎構成子項であれば，明らかに等式 $l = r$ は E と帰納的に矛盾する．この考え方を変数を含む等式へ持ち上げる．等式 $l = r$ が**構文的に矛盾する** (syntactically inconsistent) とは，l, r が共に構成子項で，$l \not\equiv r$ であることをいう．

補題 3.1 E を帰納的に完全な等号系とする．等式 $l = r$ が構文的に矛盾するならば，$l = r$ は E と帰納的に矛盾する▌

(証明) 基礎構成子項は常に存在し，かつ C には少なくとも 2 つの構成子が含まれるので，項の構造に関する帰納法を用いて，$l\theta, r\theta$ を異なる基礎構成子項とする代入 θ の存在を示すことができる． □

4 項書換え技術

本節では，帰納的定理証明に用いる項書換え技術の導入を行う．

$\succ$ を $T(F,V)$ 上の**単純化順序** [4] (simplification ordering) とする．単純化順序とは，(1) 任意の項 l, r, c について $l \succ r$ ならば $c[l] \succ c[r]$ (項の構造に関して安定)，(2) 任意の項 l, r と代入 θ について $l \succ r$ ならば $l\theta \succ r\theta$ (代入に関して安定)，(3) 任意の項 t, s について $t[s] \succ s$ を満たす順序である．単純化順序の下では，$T(F,V)$ は整礎集合となる．つまり，任意の項 $t_i \in T(F,V)$ について $t_1 \succ t_2 \succ \cdots$ なる無限減少列は存在しない [4]．さらに，$\succ$ は以下の条件を満たすとする．

(4) $\succ$ は $T(F)$ 上の全順序である．かつ，

(5) 任意の基礎項 t, u について，t が非構成子項かつ u が構成子項ならば $t \succ u$ である．

性質 (1) ～ (5) を持つように項を順序付ける方法は実際にいくつか知られている [4, 15] ので，以下，このような任意の順序 $\succ$ を一つ固定して用いる．

また，$\rhd$ を $T(F,V)$ 上の具体化に関する順序とする．つまり，2 項 t, u について $t \rhd u$ となるのは，t のある部分項が u の例であり，しかも逆が真でない場合に限られる．

$t \Leftrightarrow_E u$ かつ $t \succ u$ が成立するとき，$t \to_E u$ と書いて，E による項 t から項 u への**簡約** (reduction) と呼ぶ．つまり，$t \to_E u$ であるのは，E の等式 $l \cong r$，代入 θ，項 c があって，$t \equiv c[l\theta], u \equiv c[r\theta]$ かつ $t \succ u$ となるとき，かつ，このときに限られる．$\to_E$ の反射推移閉包を $\overset{*}{\to}_E$ で表す．$t \to_E u$ なる u が存在しないとき t は (E に関して) **既約** (irreducible)，そうでないとき**可約** (reducible) という．$t \overset{*}{\to}_E u$ かつ既約であるような u を $t\downarrow_E$ によって表す．

任意の $t \overset{*}{\Leftrightarrow}_E u$ なる基礎項 t, u について，基礎項 v が存在して $t \overset{*}{\to}_E v$，$u \overset{*}{\to}_E v$ となるならば，E は**基礎完備** (ground convergent) である[1]．基礎完備な等号系 E においては，任意の基礎項 t について $t\downarrow_E$ が一意に定まる．$\overset{*}{\to}_E$ の対称閉包を $\overset{*}{\leftrightarrow}_E$ で表すと，$\overset{*}{\Leftrightarrow}_E$ と $\overset{*}{\leftrightarrow}_E$ は一般に一致しない．しかし，E が基礎完備ならば $T(F) / \overset{*}{\Leftrightarrow}_E$ と $T(F) / \overset{*}{\leftrightarrow}_E$ は一致し，しかも任意の基礎項 t, u について $t \overset{*}{\Leftrightarrow}_E u$ となるのは，$t\downarrow_E \equiv u\downarrow_E$ のときに限られる．$g[s] \cong d$ と $l \cong r$ を等式とする．ここで，s は出現位置 p における g の部分項で，しかも変数でないものとする．s と l が単一化可能で，その最汎単一化子 θ によって $g[s]\theta \not\succeq d\theta$ かつ $l\theta \not\succeq r\theta$ となるとき，等式 $g[r]\theta = d\theta$ を $g[s] \cong d$ に $l \cong r$ を重ねた**要対** [10] (critical pair) と呼び，p を**重像位置** (superposition occurence) と呼ぶ．E_1, E_2 を 2 つの等号系とするとき，$CP(E_1, E_2)$ によって，E_1 の等式に E_2 の等式を重ねて得られるすべての要対の集合を表す．

5 帰納的定理証明のための推論規則

等式の構文的な矛盾は容易に判定でき，構文的な矛盾からは直ちに帰納的定理の不成立が帰結できる．しかし，証明する等式が帰納的に矛盾していても，それが構文的に矛盾するとは限らない．そこで本節において，帰納的に矛盾する等式を構文的に矛盾するものへ変換する推論規則を与える．

図-1に，推論規則 $\mathcal{I}$ を示す．ここで，E, G, L は等号系である (L は $\mathcal{I}$ によって変化しない)．これらの推論規則は 2 つの手続きの融合とみることができる．1 つは公理のための推論規則で，等号系 E を基礎完備なものへ変換する一種の基礎完備化手続きである．もう 1 つは定理のための推論規則で，帰納的に矛盾する G の等式を構文的に矛盾するものへ変換する．基礎完備化は文献 [2] における無向完備化手続き (ordered completion) を基本としている．但し，定理証明において E を既約に保つことはさほど重要でないため，E を簡約する推論規則は除かれている (E の簡約については最後に補足する)．推論規則のいずれかによって (E, G) を (E', G') へ変換することを推論規則の適用といい，$(E, G) \vdash (E', G')$ と書く．$(E_0, G_0) \vdash (E_1, G_1) \vdash \cdots$

[1] 項書換え系 (term rewriting systems) では，停止性と合流性を同時に持つものを完備と定義するが，ここでは $\succ$ が整礎であることより，停止性は常に保証されている．

公理のための推論規則

E 削除: $\dfrac{(E \cup \{l = t\}, G)}{(E, G)}$

E 生成: $\dfrac{(E, G)}{(E \cup \{l = r\}, G)}$ $\quad l = r \in CP(E, E)$ のとき

定理のための推論規則

G 削除: $\dfrac{(E, G \cup \{t = t\})}{(E, G)}$

G 簡約: $\dfrac{(E, G \cup \{g \cong d\})}{(E, G \cup \{g' = d\})}$

(1) $g \rightarrow_{E \cup L} g'$ のとき，または
(2) $g \prec d$ で，$g \rightarrow_G g'$ のとき，または
(3) $g \not\prec d$ で，$g \equiv c[l\theta]$ かつ $g \rhd t$ なる等式 $t \cong u \in G$ によって，$g \rightarrow_G g'$ のとき

G 分配: $\dfrac{(E, G \cup \{f(g_1, \ldots, g_n) = f(d_1, \ldots, d_n)\})}{(E, G \cup \{g_1 = d_1, \ldots, g_n = d_n\})}$ $\quad f \in C$ のとき

G 生成: $\dfrac{(E, G)}{(E, G \cup \{g = d\})}$ $\quad g = d \subset CP(G, E)$ のとき

図 1: 推論規則 $\mathcal{I}$

なる列を**演繹** (derivation) という．

定義 5.1 ($\mathcal{I}$ による定理証明)　E を帰納的に完全な公理の集合，G を証明すべき定理の集合，L を補題の集合とする．初期状態 $E_0 = E, G_0 = G$ から演繹 $(E_0, G_0) \vdash (E_1, G_1) \vdash \cdots$ を得ることを $\mathcal{I}$ による定理証明，または単に $\mathcal{I}$ という■

ここで，L の要素である等式は，すべて E の帰納的定理でなければならない．L の存在は手続きの健全性，完全性について何ら影響を与えないが，手続きの効率を上げる可能性がある．特に，後で述べるように，手続きが停止しない場合に，L の存在により手続きが停止するという効果をもたらすこともある．

演繹 $(E_0, G_0) \vdash (E_1, G_1) \cdots$ において，$\bigcup_{i=0}^{\infty} \bigcap_{j=i}^{\infty} G_j$ を G^∞ と略記する．ある $i\ (\geq 0)$ について G_i が構文的に矛盾する等式を含むとき，G^∞ は矛盾するという．ある i 以降 $\bigcup_{j=0}^{i} G_j = \bigcup_{j=0}^{i+1} G_j = \cdots$ となるとき，G^∞ は飽和するという．G^∞ が矛盾も飽和もしないとき，G^∞ は無限であるという．構文的な矛盾と同様に，飽和状態の検出も容易である．$\mathcal{I}$ による定理証明の結果 G^∞ が矛盾するならば，G は帰納的定理として成立しない．このとき，反証 (disproof) が得られたという．$\mathcal{I}$ によって G^∞ が飽和するならば，G は帰納的定理である．G^∞ が無限ならば G は帰納的定理であるが，有限の演繹でそれを確定することはできない．この場合，適当な補題や公理を追加することで演繹が有限になる，つまり G^∞ が飽和することがある [1]．

E^∞ についても G^∞ と同様に定義すると，$\mathcal{I}$ によって E^∞ は飽和するか無限であるかのいずれかである．E^∞ が飽和するのは，E に対応する基礎完備な項書換え系が有限個の要素で表現可能な場合で，実際に飽和した E が基礎完備な項書換え系となる[2]．E^∞ が無限となるのは，E に対応する完備な項書換え系で，与えられた単純化順序に適合するものが，有限の要素では表現不能な場合である．従来の証明方法では，公理に対応する基礎完備な有限の項書換え系の存在が前提となっていたのに対し，$\mathcal{I}$ は E^∞ が無限であっても証明を行える点が大きな特徴となっている．

6　証明例

自然数を 0 と s (successor) で表現し，加算，乗算と指数計算の公理から，指数法則 $x^{(y+z)} = x^y \cdot x^z$ の成立を証明する．E, G, L として，以下を与える．

$$
\begin{array}{rll}
E = \{ & x+0 = x & \text{(e1)}, \\
 & x+s(y) = s(x+y) & \text{(e2)}, \\
 & x \cdot 0 = 0 & \text{(e3)}, \\
 & x \cdot s(y) = x \cdot y + x & \text{(e4)}, \\
 & x^0 = s(0) & \text{(e5)}, \\
 & x^{s(y)} = x \cdot x^y & \text{(e6)} \quad \} \\
G = \{ & x^{(y+z)} = x^y \cdot x^z & \text{(g1)} \quad \} \\
L = \{ & 0+x = x & \text{(l1)}, \\
 & x \cdot (y \cdot z) = y \cdot (x \cdot z) & \text{(l2)} \quad \}
\end{array}
$$

ここで，補題 l1 は e1 を加算について交換したもの，l2 は乗算の交換律を特殊化したもので，l1,l2 共に帰納的定理であることが $\mathcal{I}$ で簡単に証明できる．s と 0 を構成子，それ以外の関数記号を演算子とすると，$\mathcal{I}$ によって以下の演繹が得られる．

(E, G)
$\vdash (E, G \cup \{x^{s(y+r')} = x^y \cdot x^{s(z')}\ \text{(g2)}\})$　G 生成 (g1,e2 より)
$\vdash (E, G \cup \{\text{(g2)},\ x^y = x^y \cdot x^0\ \text{(g3)}\})$　G 生成 (g1,e1 より)
$\vdash (E, G \cup \{\text{(g2)},\ x^y = x^y \cdot s(0)\})$　G 簡約 (e5 による)

[2] さらに既約性を要求すれば，このような E は $\succ$ の下で変数名の違いを除いて一意である．

$\vdash(E, G\cup\{(\mathrm{g2}),\ x^y=x^y\cdot 0+x^y\})$ G 簡約 (e4 による)
$\vdash(E, G\cup\{(\mathrm{g2}),\ x^y=0+x^y\})$ G 簡約 (e3 による)
$\vdash(E, G\cup\{(\mathrm{g2}),\ x^y=x^y\})$ G 簡約 (l1 による)
$\vdash(E, G\cup\{(\mathrm{g2})\})$ G 削除
$\vdash(E, G\cup\{x\cdot x^{(y+z')}=x^y\cdot x^{s(z')}\})$ G 簡約 (e6 による)
$\vdash(E, G\cup\{x\cdot x^{(y+z')}=x^y\cdot(x\cdot x^{z'})\})$ G 簡約 (e6 による)
$\vdash(E, G\cup\{x\cdot x^{(y+z')}=x\cdot(x^y\cdot x^{z'})\})$ G 簡約 (l2 による)
$\vdash(E, G\cup\{x\cdot(x^y\cdot x^{z'})=x\cdot(x^y\cdot x^{z'})\})$ G 簡約 (g1 による)
$\vdash(E, G)$ G 削除

他の演繹は存在しないので，E^∞, G^∞ 共に飽和している．つまり，$\mathcal{I}$ の結果は「定理成立」となる．

次に，E^∞ が無限となる例を示す．最大公約数を求める関数 g の定義を以下のように与え，$g(x\cdot y, z\cdot y) = y$ が必ずしも成り立たないことを証明する．E, G, L として，以下を与える．

$$
\begin{array}{lll}
E=\{ & x+0=x & (\mathrm{e1}),\\
 & x+s(y)=s(x+y) & (\mathrm{e2}),\\
 & x\cdot 0=0 & (\mathrm{e3}),\\
 & x\cdot s(y)=x\cdot y+x & (\mathrm{e4}),\\
 & g(x,0)=x & (\mathrm{e5}),\\
 & g(0,x)=x & (\mathrm{e6}),\\
 & g(x+y,y)=g(x,y) & (\mathrm{e7}),\\
 & g(x,y+x)=g(x,y) & (\mathrm{e8})\ \ \}\\
G=\{ & g(x\cdot y, x\cdot z)=x & (\mathrm{g1})\ \ \}\\
L= & \phi &
\end{array}
$$

この E に対して完備化を行うと，

$$
\begin{aligned}
g(s^n(x+y), s^n(y)) &= g(x, s^n(y))\\
g(s^n(x), s^n(0)) &= g(x, s^n(0))\\
g(s^n(x), s^n(y+x)) &= g(s^n(x), y)\\
g(s^n(0), s^n(x)) &= g(s^n(0), x)
\end{aligned}
$$

なる無限の要対が獲得され，完備な項書換え系は有限の要素によって表現できない[7]．このような場合でも，以下のような演繹が得られる．先の例と同様に，s と 0 を構成子，それ以外の関数記号を演算子とする．

(E, G)
$\vdash(E, G\cup\{g(x\cdot y'+x, z\cdot s(y'))=s(y')\})$ G 生成 (g1,e4 より)
$\vdash(E, G\cup\{g(x\cdot y'+x, z\cdot y'+z)=s(y')\ (\mathrm{g2})\})$ G 簡約 (e4 による)
$\vdash(E, G\cup\{(\mathrm{g2}), g(0\cdot y', z\cdot y'+z)=s(y')\ (\mathrm{g3})\})$ G 生成 (g2,e1 より)
$\vdash(E, G\cup\{(\mathrm{g2}),(\mathrm{g3}), g(0, z\cdot 0+z)=s(0)\})$ G 生成 (g3,e3 より)
$\vdash(E, G\cup\{(\mathrm{g2}),(\mathrm{g3}), g(0, 0+z)=s(0)\})$ G 簡約 (e3 による)
$\vdash(E, G\cup\{(\mathrm{g2}),(\mathrm{g3}), 0+z=s(0)\ (\mathrm{g4})\})$ G 簡約 (e6 による)
$\vdash(E, G\cup\{(\mathrm{g2}),(\mathrm{g3}),(\mathrm{g4}), 0=s(0)\ (\mathrm{g5})\})$ G 生成 (g4,e1 より)

ここで，g5 は構文的に矛盾している．つまり，E^∞ は無限であっても，G^∞ が矛盾するので，証明結果は「定理不成立」となる．

7 推論規則の完全性

本節では $\mathcal{I}$ が反駁的に完全であることを示す．まず，演繹の健全性を確認する．

定理 7.1 ($\mathcal{I}$ の健全性) $\mathcal{I}$ において反証が得られたならば，G は E と帰納的に矛盾する▌

(証明) $(E, G)\vdash(E', G')$ を演繹とするとき，次がいえればよい．

(1) E が帰納的に完全ならば E' も帰納的に完全である．かつ，

(2) G' が E' と帰納的に矛盾するならば G は E と帰納的に矛盾する．

$\mathcal{I}$ の定義より，$T(F)\ /\ E = T(F)\ /\ E'$，かつ $T(F)\ /\ E\cup G = T(F)\ /\ E'\cup G'$ であることが示せるので，(1)，(2) の成立は明らかである． □

E を帰納的に完全な等号系，G を等号系とする．G の等式 $g=d$ が E と帰納的に矛盾していると，基礎代入 θ，異なる基礎構成子項 u, v があって，関係 $u \overset{*}{\Leftrightarrow}_E g\theta \Leftrightarrow_G d\theta \overset{*}{\Leftrightarrow}_E v$ が成り立つ．つまり，以下の列が存在する．

$$t_0\Leftrightarrow_E\cdots\Leftrightarrow_E t_m\Leftrightarrow_G t_{m+1}\Leftrightarrow_E\cdots\Leftrightarrow_E t_n\quad(0\le m<n)$$

ここで，$t_0\equiv u, t_m\equiv g\theta, t_{m+1}\equiv d\theta, t_n\equiv v$ である．このように，すべての $t_i\ (0\le i\le n)$ が基礎項で，唯一の $\Leftrightarrow_G$ ステップと，複数 (0 個以上) の $\Leftrightarrow_E$ ステップを含み，両端の項が異なる基礎構成子項であるような列を，$E\cup G$ による $g\theta\Leftrightarrow_G d\theta$ の**反証列** (disproof sequence) と呼ぶ．反証列が $\Leftrightarrow_E$ ステップを含まないとき，**正規** (normal) であるという．

補題 7.2 E, G を等号系とする．$E\cup G$ による $g\theta\Leftrightarrow_G d\theta$ の正規な反証列が存在するとき，かつこのときに限り，G は構文的に矛盾する等式を含む▌

(証明) 定義より明らか． □

$\mathcal{I}$ によって任意の反証列が正規なものに変換されることを示すために，反証列の上に順序を導入する．$t\Leftrightarrow t'$ を，E または G の等式 $l\cong r$，代入 θ，項 c について $t\equiv c[l\theta], t'\equiv c[r\theta]$ となるような反証列のステップとする．このステップの重み w を以下のように定める．

$$
w(t\Leftrightarrow t')\overset{\mathrm{def}}{=}\begin{cases}(\{t\}, l, t') & t\succ t'\ \text{のとき}\\ (\{t'\}, r, t) & t\prec t'\ \text{のとき}\\ (\{t, t'\}, \bot, \bot) & t\equiv t'\ \text{のとき}\end{cases}
$$

ここで，$\{\cdots\}$ は多重集合である．w の値は，$\succ$ を用いた多重集合順序[3]($\succ_{\mathrm{mul}}$ で表す) と $\rhd$，$\succ$ の3順序を辞書式に組み合わせることで比較できる．この順序を $\sqsupset$ で表す．すなわち，$(x_1, x_2, x_3), (y_1, y_2, y_3)$ を2つのステップの重みとするとき，$(x_1, x_2, x_3) \sqsupset (y_1, y_2, y_3)$ であるのは次のときに限られる．

(1) $x_1 \succ_{\mathrm{mul}} y_1$ である．または，

(2) $x_1 = y_1$ かつ $x_2 \rhd y_2$ である．または，

(3) $x_1 = y_1, x_2 = y_2$ かつ $x_3 \succ y_3$ である．

さらに $\mathcal{P}$ を $t_0 \Leftrightarrow_E \cdots \Leftrightarrow_E t_m \Leftrightarrow_G t_{m+1} \Leftrightarrow_E \cdots \Leftrightarrow_E t_n$ なる反証列とするとき，$\mathcal{P}$ の重み W を以下のように定義する．

$$W(\mathcal{P}) \stackrel{\mathrm{def}}{=} (w(t_m \Leftrightarrow_G t_{m+1}), \{w(t_0 \Leftrightarrow_E t_1), \ldots, w(t_{n-1} \Leftrightarrow_E t_n)\})$$

つまり $W(\mathcal{P})$ は，$\mathcal{P}$ の中で唯一のステップ $t \Leftrightarrow_G t'$ についてその重みを第1要素とし，複数の $t \Leftrightarrow_E t'$ についてそれらすべての重みの多重集合を第2要素とする組である．w の値の多重集合は $\sqsupset$ を用いた多重集合順序 ($\sqsupset_{\mathrm{mul}}$ で表す) によって比較できる．そこで，$\sqsupset$ と $\sqsupset_{\mathrm{mul}}$ を辞書式に組み合わせると，$W(\mathcal{P})$ の値を比較する順序が得られる．この順序を $\sqsupset$ で表す．すなわち，上記の $\mathcal{P}$ に加えて，$\mathcal{P}'$ を $u_0 \Leftrightarrow_E \cdots \Leftrightarrow_E u_p \Leftrightarrow_G u_{p+1} \Leftrightarrow_E \cdots \Leftrightarrow_E u_q$ なる反証列とすると，$W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ であるのは次のときに限られる．

(1) $w(t_m \leftrightarrow_G t_{m+1}) \sqsupset w(u_p \leftrightarrow_G u_{p+1})$ である．または，

(2) $w(t_m \Leftrightarrow_G t_{m+1}) = w(u_p \Leftrightarrow_G u_{p+1})$ かつ $\{l_i \Leftrightarrow_E l_{i+1}\} \sqsupset_{\mathrm{mul}} \{u_j \Leftrightarrow_E u_{j+1}\}$ である．

元の順序が整礎であれば，それを用いた多重集合順序も整礎となるため[3]，$\sqsupset$ も整礎である．

次に，推論規則 $\mathcal{I}$ の適用に関する条件を設定するために，文献 [2] と同様の被覆集合を導入する．E, G, G' を等号系とする．G' が E において等式 $g = d \in G$ の**被覆集合** (covering set) であるとは，$g\theta \overset{\bullet}{\Leftrightarrow}_E d\theta$ であるような任意の基礎代入 θ について，等式 $g' = d' \in G'$ と，基礎代入 θ' が存在し，$g'\theta' \overset{\bullet}{\Leftrightarrow}_E d'\theta'$ かつ $w(g\theta \Leftrightarrow_G d\theta) \sqsupseteq w(g'\theta' \Leftrightarrow_{G'} d'\theta')$ となることをいう．G' が G の被覆集合であるとは，G の任意の等式について，G' が被覆集合となることをいう[3]．

[3] ここでは，無意味な等式を被覆集合から排除するために，Bachmair の定義を拡張している．また，この被覆集合は，Zhang らによる被覆集合[16] とはまったく異なるものである．

推論規則の適用条件を定める．演繹において，削除，簡約，分配規則は常に優先的に適用することにする．また，生成規則の適用に関して，以下の仮定を設ける．

(F1) $\mathcal{CP}(E^\infty, E^\infty) \subset \bigcup_{i=0}^{\infty} E_i$ である．かつ，

(F2) G^∞ は E^∞ において $\mathcal{CP}(G^\infty, E^\infty)$ の被覆集合である．

$\mathcal{I}$ がこれらの条件を満足するとき，公平 (fair) であるという．

定理 7.3 $\mathcal{I}$ は公平であるとする．このとき，$E \cup G$ による反証列があれば，$E^\infty \cup G^\infty$ による正規な反証列が存在する▌

(証明) 仮定より，$(\bigcup_{i=0}^{\infty} E_i) \cup (\bigcup_{i=0}^{\infty} G_i)$ による反証列は明らかに存在する．

$t_0 \Leftrightarrow_{\cup E} \cdots \Leftrightarrow_{\cup E} t_m \Leftrightarrow_{\cup G} u_n \Leftrightarrow_{\cup E} \cdots \Leftrightarrow_{\cup E} u_0$ $(\mathcal{P})$ をそのような反証列で，極小の重みを持つものとする．ここで，等式 $g \simeq d \in \bigcup_{i=0}^{\infty} G_i$ と基礎代入 θ について，$t_m = g\theta, u_n \equiv d\theta$ であるとする．このような $\mathcal{P}$ が $E^\infty \cup G^\infty$ による正規な反証列であることをいう．このためには，(1) $g \simeq d \in G^\infty$ で，(2) $m \simeq 0$，かつ (3) $n \equiv 0$ であることがいえればよい．ここで，$t_0 \overset{\bullet}{\Leftrightarrow}_E u_0$ より，$t_m \equiv u_n$ であることはあり得ないので，$g\theta \Leftrightarrow_{\cup G} d\theta$ としても一般性を失わない．

(1 の証明) $g \simeq d$ が，ある i において G_i に含まれ，かつ G^∞ には含まれないとする．これが起こるのは，G 簡約規則または G 分配規則が適用された場合であるが，どちらの場合も矛盾することを以下で示す．

G 簡約規則が適用され，$g \simeq d \in G_i$ がある j $(j > i)$ において $g' \simeq d \in G_j$ に置き換わったとする．このとき，g と g' の関係は (a) $g\theta \overset{\bullet}{\Leftrightarrow}_{E_j} g'\theta$，(b) $g\theta \overset{\bullet}{\Leftarrow}_{E_j} g'\theta$ のどちらかである．(a) の場合，$\mathcal{P}$ の $g\theta \rightarrow_{G_i} d\theta$ を $g\theta \overset{\bullet}{\Leftrightarrow}_{E_j} g'\theta \Leftrightarrow_{G_j} d\theta$ で置き換えた $\mathcal{P}'$ を考えると，$\mathcal{P}'$ は $E_j \cup G_j$ による反証列で，しかも $g\theta \succ g'\theta, g\theta \succ d\theta$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となるので，$\mathcal{P}$ が極小という仮定に反する．(b) の場合，$g \equiv c[l_1\theta_1], g' \equiv c[r_1\theta_1]$ なる G_j の等式 $l_1 \simeq r_1$ が存在する．そこで，新たな $\mathcal{P}'$ として $t_0' \overset{\bullet}{\Leftrightarrow}_{E_j} l_1\theta_1\theta \rightarrow_{G_j} r_1\theta_1\theta \overset{\bullet}{\Leftrightarrow}_{E_j} u_0'$ を考えると (t_0', u_0' は異なる基礎構成子項)，$g\theta \succeq l_1\theta_1\theta, g \rhd l_1$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となり，仮定に反する．

G 簡約規則が適用され，$g \simeq d \in G_i$ が $g \simeq d' \in G_j$ に置き換わったとする．このとき，d と d' の関係は

(c) $d\theta \overset{\cdot}{\Leftrightarrow}_{E_j} d'\theta$, (d) $d\theta \overset{\cdot}{\not\Leftrightarrow}_{E_i} d'\theta$ のどちらかである. (c) の場合, $g\theta \rightarrow_{G_i} d\theta$ を $g\theta \rightarrow_{G_j} d'\theta \overset{\cdot}{\Leftrightarrow}_{E_j} d\theta$ で置き換えた $\mathcal{P}'$ を考えると, $d\theta \succ d'\theta$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となり, 仮定に反する. (d) の場合, $d \equiv c[l_1\theta_1], d' \equiv c[r_1\theta_1]$ なる G_j の等式 $l_1 \cong r_1$ が存在するので, (b) と同様の $\mathcal{P}'$ を考えると, $g\theta \succ d\theta \succeq l_1\theta_1\theta$ より $W(\mathcal{P}) \sqsupset\!\!\!\sqsupset W(\mathcal{P}')$ となり, 仮定に反する.

$g \equiv f(g_1, \ldots, g_p), d \equiv f(d_1, \ldots, d_p), f \in C$ で, G 分配規則が適用され, $g \cong d \subset G_i$ が $g_1 = d_1, \cdots, g_p = d_p \in G_j$ に置き換わったとする. このとき, 少なくとも 1 つの k $(1 \leq k \leq p)$ について $g_k\theta \overset{\cdot}{\not\Leftrightarrow}_{E_j} d_k\theta$ となるので, 新たな $\mathcal{P}'$ として $t'_0 \overset{\cdot}{\Leftrightarrow}_{E_j} g_k\theta \Leftrightarrow_{G_j} d_k\theta \overset{\cdot}{\Leftrightarrow}_{E_j} u'_0$ を考えると, $g\theta \succ g_k\theta, g\theta \succ d\theta \succ d_k\theta$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となり, 仮定に反する.

(2 の証明) $m \neq 0$ であるとする. まず, $\mathcal{P}$ の $t_0 \Leftrightarrow_{\cup E} \cdots \Leftrightarrow_{\cup E} t_m$ 部分が以下の (S1) ～ (S3) のステップを含まないことをいう.

$$t_k \Leftrightarrow_{\cup E} t_{k+1} \tag{S1}$$

で, $t_k \equiv t_{k+1}$ なるステップが含まれるとする. そのとき, (S1) を単独の t_k で置き換えた反証列 $\mathcal{P}'$ を考えると, $\{(\{t_k, t_{k+1}\}, \bot, \bot)\} \sqsupset_{\mathrm{mul}} \phi$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となり, 仮定に反する.

$$t_{k-1} \leftarrow_{\cup E} t_k \rightarrow_{\cup E} t_{k+1} \tag{S2}$$

が含まれ, $l_1 \cong r_1, l_2 \cong r_2 \in \bigcup_{i=0}^{\infty} E_i$ について, $t_{k-1} \equiv c_1[r_1\theta_1], t_k \equiv c_1[l_1\theta_1] \equiv c_2[l_2\theta_2], t_{k+1} \equiv c_2[r_2\theta_2]$ であるとする. このとき l_1 と l_2 の位置関係は, (a) 重なりを持たない, (b) l_1 が l_2 の変数部分に重なる, (b') l_2 が l_1 の変数部分に重なる, (c) l_1 と l_2 が互いの非変数部分に重なるのいずれかである. (a) の場合, ある項 e があって, (S2) は $e[r_1\theta_1, l_2\theta_2] \leftarrow_{\cup E} e[l_1\theta_1, l_2\theta_2] \rightarrow_{\cup E} e[l_1\theta_1, r_2\theta_2]$ なる形である. すると, (S2) を $e[r_1\theta_1, l_2\theta_2] \rightarrow_{\cup E} e[r_1\theta_1, r_2\theta_2] \leftarrow_{\cup E} e[l_1\theta_1, r_2\theta_2]$ で置き換えた $\mathcal{P}'$ が存在し, $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となるので仮定に反する. (b) の場合, $t_{k-1} \overset{\cdot}{\rightarrow}_{\cup E} c_2[l_2\theta'_2], r_2\theta_2 \overset{\cdot}{\rightarrow}_{\cup E} r_2\theta'_2$ なる代入 θ'_2 が存在する. すると, (S2) を $t_{k-1} \overset{\cdot}{\rightarrow}_{\cup E} c_2[l_2\theta'_2] \rightarrow_{\cup E} c_2[r_2\theta'_2] \overset{\cdot}{\leftarrow}_{\cup E} c_2[r_2\theta_2]$ で置き換えた $\mathcal{P}'$ が存在し, $W(\mathcal{P}) \sqsupset\!\!\!\sqsupset W(\mathcal{P}')$ となるので仮定に反する. (b') は (b) の左右対称な場合である. (c) の場合, $l_1 \not\equiv r_1, l_2 \not\equiv r_2$ より $l_1 \cong r_1, l_2 \cong r_2 \in E^{\infty}$ なので, $t_{k-1} \equiv c_2[l_3\theta_3], t_{k+1} \equiv c_3[r_3\theta_3], l_3 \cong r_3 \in \mathcal{CP}(E^{\infty}, E^{\infty})$ なる要対 $l_3 \cong r_3$ が存在し, 公平の条件 (F1) は, この要対が $\bigcup_{i=0}^{\infty} E_i$ に存在することを保証する. すると, (S2) を $c_3[l_3\theta_3] \Leftrightarrow_{E_j} c_3[r_3\theta_3]$ で置き換えた $\mathcal{P}'$ が存在し, $t_k \succ c_3[l_3\theta_3], t_k \succ c_3[r_3\theta_3]$ より $W(\mathcal{P}) \sqsupset W(\mathcal{P}')$ となるので, 仮定に反する.

$$t_{m-1} \rightarrow_{\cup E} g\theta \tag{S3}$$

が含まれ, $l_1 \cong r_1 \in \bigcup_{i=0}^{\infty} E_i$ について $t_{m-1} \equiv c[r_1\theta_1], g\theta \equiv c[l_1\theta_1]$ であるとする. l_1 と g の位置関係は, (d) l_1 が g の変数部分に重なる, (c) l_1 が g の非変数部分に重なるのいずれかである. (d) の場合, (b) と同様の議論により, 矛盾が導かれる. (c) の場合, $g \cong d \in C^{\infty}$ なので, $t_{m-1} \equiv g_2\theta_2, d\theta \equiv d_2\theta_2, g_2 \cong d_2 \in \mathcal{CP}(G^{\infty}, E^{\infty})$ なる要対 $g_2 \cong d_2$ が存在し, 公平の条件 (F2) は, $g_3\theta_3 \overset{\cdot}{\not\Leftrightarrow}_{E^{\infty}} d_3\theta_3$ かつ $w(g_2\theta_2 \Leftrightarrow d_2\theta_2) \sqsupseteq w(g_3\theta_3 \Leftrightarrow d_3\theta_3)$ であるような基礎代入 θ_3 と等式 $g_3 = d_3 \subset G^{\infty}$ の存在を保証する. そこで, 新たな $\mathcal{P}'$ として $t'_0 \overset{\cdot}{\Leftrightarrow}_{\cup E} g_3\theta_3 \Leftrightarrow_{G^{\infty}} d_3\theta_3 \overset{\cdot}{\Leftrightarrow}_{\cup E} u'_0$ を考えると, $g\theta \succ g_3\theta_3, g\theta \succ d_3\theta_3$ より $W(\mathcal{P}) \sqsupset\!\!\!\sqsupset W(\mathcal{P}')$ となり, 仮定に反する.

以上により (S1) ～ (S3) を含まない $\mathcal{P}$ は, $t_0 \rightarrow_{\cup E} \cdots \rightarrow_{\cup E} t_m \cdots$ なる形である. しかし, これは t_0 が基礎構成子項であることに反する.

(3 の証明) $m = 0$ より, $g\theta$ は基礎構成子項である. すると, 当然 $d\theta$ も基礎構成子項でなければならない. よって $n = 0$ である. □

定理 7.3 によって, $\mathcal{I}$ の完全性は明らかとなる.

系 7.4 ($\mathcal{I}$ の完全性) 公平な $\mathcal{I}$ は反駁的に完全である. つまり, 証明すべき定理の集合 G の中に, 公理の集合 E と帰納的に矛盾するものがあれば, $\mathcal{I}$ によって反証が得られる▌

(証明) G の等式 $l = r$ が E と帰納的に矛盾しているとする. すると, 基礎代入 θ があって, $E \cup G$ による $l\theta \neq r\theta$ の反証列が存在する. $\mathcal{I}$ は公平なので, $E^{\infty} \cup G^{\infty}$ による正規な反証列が存在する (定理 7.3). つまり, G^{∞} は矛盾する等号系である (補題 7.2). これは, 有限の演繹によって反証が得られることを意味する. □

8 証明手続きの実現

$\mathcal{I}$ は, 代数的仕様の開発支援環境 *Metis*[12] の拡張機能として実現済みであり, この上でいくつかの仕様検証実験も行われている. 本節では, $\mathcal{I}$ に基づく手続きを実現する際の要点について簡単に説明する.

E の帰納的完全性を検査する方法について述べる.

帰納的完全性の条件(C2)を満たさないEについては，証明の過程でEが矛盾するので明らかとなる．条件(C1)は，任意の演算子$f \in F$と任意の基礎構成子項$t_i \in T(C)$について$f(t_1,\ldots,t_n)$がEに関して可約であるならば満足される．この判定は文献[13]の方法によって有限のEに対して決定可能であるため，理論的には演繹の任意の時点でこれを検査することができる．しかし，文献[13]の方法は，計算コストが非常に高いので，実際にどうするかは今後の課題といえよう．

被覆集合の計算は$\mathcal{I}$において重要である．$CP(G^\infty, E^\infty)$全体を$\bigcup_{i=0}^{\infty} G_i$に獲得すれば，公平の条件(F2)は満足される．しかし，E^∞による要対をすべて獲得するこの方法では，E^∞が無限で，かつG^∞が飽和するときに，定理の成立を有限の演繹で確定することができない．たとえE^∞が無限であっても，有限のEによってG^∞の被覆集合を獲得できれば，有限の演繹で定理の正しさを帰結できる．Fribourgは，潜在帰納法において要対を獲得する際に，1つの定理について，ある条件を満たす1つの重像位置のみを考慮すれば十分であることを示した[5]．Bachmairは，この制限の妥当性を被覆集合の考え方で説明した[2]．この制限による被覆集合の獲得を$\mathcal{I}$に導入する．

Eを等号系，tを項とする．出現位置pにおけるtの部分項をsとするとき，任意の基礎代入θについて$s\theta$全体がEに関して可約であるならば，出現位置pはEに関して帰納的に完全であるという(このとき条件よりsは変数ではあり得ない)．Eを等号系，$g = d$を等式とする．$g = d$にEの等式を重ねる重像位置の中で，Eに関して帰納的に完全なpが存在するとき，pにおいて得られるすべての要対の集合を$g = d$のpにおけるE完全要対集合と呼ぶ．等式の任意のE完全要対集合は，単独でその等式の被覆集合となる[2]．さらには，G生成規則によってGの各等式について少なくとも1つのE完全要対集合を獲得すれば，公平の条件(F2)は満たされる．うまくE完全要対集合を選択すると，E^∞が無限である場合に，有限のEによってG^∞の被覆集合を獲得することが可能となる．

$T(F,V)$上の順序$\succ$は，再帰経路順序 (recursive path ordering)に基づく順序法[4]を拡張することで，容易に実現できる．具体的には，演算子の集合Dと構成子の集合Cをそれぞれ全順序に整列し，さらに，いかなる構成子も演算子より小さいことを仮定すれば，再帰経路順序の辞書式比較法や勝抜き比較法[15]などが求める順序となる．

E簡約: $\dfrac{(E \cup \{l \cong r\}, G)}{(E \cup \{l' = r\}, G)}$
(1) $l \prec r$で，$l \to_E l'$のとき，または
(2) $l \not\prec r$で，$l \equiv c[t\theta]$かつ$l \rhd t$なる等式$t \cong u \in G$によって，$l \to_G l'$のとき

E分配: $\dfrac{(E \cup \{f(l_1,\ldots,l_n) = f(r_1,\ldots,r_n)\}, G)}{(E \cup \{l_1 = r_1, \ldots, l_n = r_n\}, G)}$
$f \in C$のとき

図2: E簡約，E分配規則

定理7.3の証明を見ると，$\mathcal{I}$の反駁完全性に本質的に関わるのは，E生成，G生成の2規則のみであることがわかる．その他の規則は，証明効率の向上に重要な役割を果たしている．これらの規則ほど効果的でないにしても，図-2に示すようなE簡約，分配規則を$\mathcal{I}$に追加して，証明の効率をさらに向上させることができる．特に，これらの規則を追加した定理証明によってE^∞が飽和するとき，得られる基礎完備な項書換え系は既約である(よって$\succ$の下で一意である)．図-2の規則を含む定理証明の完全性も，同様の方法によって証明できる．

9 おわりに

本稿では帰納的定理を証明するための推論規則を与え，その反駁完全性を示した．この方法は構成子の存在を前提としており，文献[5, 2]のような帰納的可約性による矛盾の検出は扱えない．これは，無限のE^∞による証明を可能とした結果，帰納的可約性が決定不能となったことによる．しかし，今までに知られている帰納的可約性の検査(例えば文献[13]の方法)は多大な計算コストを必要とし，実装に適した技術とは言い難いため，実用的には構成子を用いる方法への限定に問題はないと考えられる．帰納的に完全でない公理が$\mathcal{I}$に与えられた場合，演繹過程でそれが明らかにされるが，この検査を効率よく行う方法は今後の課題である．また，補題や公理の追加によって，E^∞やG^∞が無限となるのを防げる場合があることを述べたが，適切な補題や公理を自動獲得する方法も今後の課題であ

6.

参考文献

[1] Avenhaus, J.: Proving Equational and Inductive Theorems by Completion and Embedding Techniques, in *Proc. 4th Int. Conf. Rewriting Techniques and Applications*, Lecture Notes in Computer Science 488, pp. 361–373, Springer-Verlag (1991).

[2] Bachmair, L.: *Canonical Equational Proofs*, Progress in Theoretical Computer Science, Birkhäuser, Boston (1991).

[3] Dershowitz, N. and Manna, Z.: Proving termination with multiset orderings, *Comm. ACM*, Vol. 22, No. 8, pp. 465–467 (1979).

[4] Dershowitz, N.: Orderings for term-rewriting systems, *Theor. Comput. Sci.*, Vol. 17, No. 3, pp. 279–301 (1982).

[5] Fribourg, L.: A strong restriction of the inductive completion procedure, *J. Symbolic Computation*, Vol. 8, pp. 253–276 (1989).

[6] Goguen, J. A.: How to prove algebraic induction hypothesis without induction, with application to the correctness of data type implementation, in *Proc. 5th Int. Conf. on Automated Deduction*, Lecture Notes in Computer Science 87, pp. 356–373, Springer Verlag (1980).

[7] Hermann, M. and Privara, I.: On nontermination of Knuth-Bendix algorithm, Research Report CS-842 21, Institute of Socio-Economic Information and Automation (1985), VUSEI-AR-OPS-3/85.

[8] Huet, G. and Oppen, D. C.: Equations and Rewrite Rules: A Survey, *Formal Languages: Perspective and Open Problems* (Book, R. ed.), pp. 349–405, Academic Press (1980).

[9] Huet, G. and Hullot, J.-M.: Proofs by induction in equational theories with constructors, *J. Comput. Syst. Sci.*, Vol. 25, No. 2, pp. 239–266 (1982).

[10] Knuth, D. E. and Bendix, P. B.: Simple word problems in universal algebras, *Proc. Computational problems in abstract algebra* (Leech, J. ed.), pp. 263–297, Pergamon Press, Oxford (1970); also in *Automation of Reasoning 2* (Siekmann, J. H. and Wrightson eds.), Springer-Verlag (1983), pp. 342–376.

[11] Musser, D. R.: On proving inductive properties of abstract data types, in *Proc. 7th ACM Symposium on Principles of Programming Languages*, pp. 154–162 (1980).

[12] Ohsuga, A. and Sakai, K.: Metis: A Term Rewriting System Generator, *Software Science and Engineering* (Nakata, I. and Hagiya, M. eds.), pp. 1–15, World Scientific (1991).

[13] Plaisted, D. A.: Semantic confluence tests and completion methods, *Inf. Control*, Vol. 65, pp. 182–215 (1985).

[14] Reddy, U. S.: Term Rewriting Induction, in *Proc. 10th Int. Conf. on Automated Deduction*, Lecture Notes in Computer Science 449, pp. 162–177, Springer-Verlag (1990).

[15] Sakai, K.: An ordering method for term rewriting systems, Tech. Report TR-062, ICOT (1984).

[16] Zhang, H., Kapur, D., and Krishnamoorthy, M. S.: A Mechanizable Induction Principle for Equational Specification, in *Proc. 9th Int. Conf. on Automated Deduction*, Lecture Notes in Computer Science 310, pp. 162–181, Springer Verlag (1988).