宣言的プログラムのアンフォールド変換

赤間 清[†] 繁田良則[††] 宮本衛市[†††]

宣言的プログラムは，論理プログラムを任意のデータ構造を利用できるように拡張した問題記述のクラスである．本論文では，宣言的プログラムのアンフォールド変換の理論を与える．一般の宣言的プログラムでは,「健全でない変換の存在」や「最汎ユニファイアの非存在」などの特徴があり，論理プログラムに関するアンフォールド変換の理論の単純な拡張では，一般的な理論を構築することは困難である．本論文では，既存のユニファイアのかわりにペアユニファイアという概念を導入し，健全なペアユニファイア集合や完全なペアユニファイア集合という概念を提案する．またこれらの定義に基づいて，宣言的プログラムのアンフォールド変換の健全性と完全性の定理を提案し，その正当性を証明する．この理論は，宣言的プログラムを用いた問題解決の基礎を与える．

Unfolding of Declarative Programs

Kiyoshi Akama,[†] Yoshinori Shigeta[††] and Eiichi Miyamoto[†††]

Declarative programs form a class of problem descriptions that are extended from logic programs so as to utilize arbitrary data structures. In this paper, we propose a theory for unfolding of declarative programs. Since unfolding of declarative programs may be unsound and most general unifiers do not exist for many declarative programs, it is very difficult to develop a general theory of unfolding for declarative programs as a simple extension of the one for logic programming. We introduce a new concept of pair unifiers in place of usual unifiers and propose concepts of sound pair-unifier sets and complete pair-unifier sets. Based on these definitions, we also propose and prove a theorem of sound and complete unfolding for declarative programs. This theory gives a foundation of problem solving in terms of declarative programs.

1. はじめに

「等価変換パラダイム」は，「問題を等価変換によって簡単化して解く」という考えに基づき，幅広い問題記述に対して適用できる問題解決の枠組である[3]．多くの問題は，問題文を簡単な形に言い替えることによって解くことができる．たとえば，連立方程式の通常の解法は，連立方程式の解集合を保存しつつ連立方程式を簡単な形に逐次変形するものである．論理プログラミングにおけるSLDレゾリューションによる問題の解法も，見方を変えれば，問題を記述した確定節集合の宣言的意味を保存しながら，確定節集合をアンフォールド変換で簡単化するものと見ることができる[4]．

† 北海道大学 情報メディア教育研究総合センター
Center for Information and Multimedia Studies, Hokkaido University
†† 東芝システム LSI 技術研究所
System ULSI Engineering Laboratory, Toshiba Corporation
††† 北海道大学 大学院 システム情報工学専攻
Deptartment of System and Information Engineering, Hokkaido University

「等価変換パラダイム」では次の手順で問題を解く．

(1) [問題の定式化] 問題をあるクラスの宣言的記述で正確に定式化する．
(2) [ルールの準備] その宣言的記述を等価変換するルールを準備する．
(3) [制御の決定] 宣言的記述のどの部分にどのルールを適用するかを指定する規則を決定する．
(4) [簡単化の実行] 上記のルールと制御を，問題の定式化である宣言的記述に適用して，実際に問題を簡単化する．
(5) [解の計算] 簡単化された宣言的記述から問題の解を得る．

より具体的には，問題の定式化は，確定節の集合を用いて行う．等価変換による問題解決の簡単な例は，論文15)，18) などに見られる．

等価変換に基づいた問題解決の能力は，問題を表現する方法（表現システム[3]）に大きく依存する．連立方程式や論理プログラムは自然に表現システムをなす

が，それらは表現力や計算速度の点で限界がある．そこで論理プログラムと宣言的意味の構造を保存したまま，多彩なデータ構造を許すように，最大限表現力を高めた☆プログラムの概念が導入された．その豊富な表現力は，データ構造が項に限定されている論理プログラムの表現力を本質的に拡大する．表現力の拡張は予想以上に大きく，論理プログラムだけでなく，既存の多くの知識表現系をサブクラスとして包含する力がある．

本論文では，宣言的プログラムの最も基本的な等価変換の1つであるアンフォールド変換について議論する．フォールド変換は本論文の範囲外とする．等価変換による問題解決の第2ステップでは，アンフォールド変換に基づいて多くのルールが作られ，利用される．アンフォールド変換に基づくルールは，ルールの自動生成の基礎ともなる[8]．

論理プログラムの範囲では，リネーミングや最汎ユニファイアの概念を用いてアンフォールド変換が定義され，アンフォールド変換の正当性が示されている．しかし，その定義や理論を宣言的プログラムの場合に適切に拡張するのは，自明な問題ではない．宣言的プログラムは，抽象的な公理に基づくプログラムのクラスであり，論理プログラムに比べてはるかに広い範囲の記述を包含している．そこでは，リネーミングという概念は特殊すぎるし，最汎ユニファイアのように，他のユニファイアすべてを代表できるユニファイアが存在する保証もない．このようないくつかの差異を乗り越えて，宣言的プログラムに広く適用できるアンフォールド変換の枠組を構築するために，本論文では，ユニファイアの概念を変更する．

本理論は，宣言的プログラムのアンフォールド変換を十分一般性を持って扱っている．多くの言語や知識表現で書かれたプログラムや知識は，それらを宣言的プログラムと見なすことによって，本理論のアンフォールド変換を適用することができる．

2. 特殊化システム上の宣言的プログラム

2.1 特殊化システム

論理プログラムの基礎には，項という対象とそれに作用している代入から構成された構造がある．しかし，広範な応用問題を扱うためには，項代入構造は特殊すぎる．そこで項代入構造を拡張した特殊化システムの概念が導入されている．

☆ 宣言的プログラムの概念は，すでに否定を扱うように拡張されている[11]が，本論文では否定を含まない範囲で議論する．

定義 1 特殊化システムとは，集合 $\mathcal{A}$, $\mathcal{G}$, $\mathcal{S}$ と写像 $\mu : \mathcal{S} \to partial\_map(\mathcal{A})$ からなる4項組 $\Gamma = \langle \mathcal{A}, \mathcal{G}, \mathcal{S}, \mu \rangle$ で，次の条件を満たすものである．ただし，$partial\_map(\mathcal{A})$ は $\mathcal{A}$ 上の部分写像全体の集合を表す．$\mu(s_1) \circ \mu(s_2)$ は2つの部分写像 $\mu(s_1)$ と $\mu(s_2)$ の合成写像である☆☆．

(1) $\forall s_1, s_2 \in \mathcal{S}, \exists s \in \mathcal{S} : \mu(s) = \mu(s_1) \circ \mu(s_2)$

(2) $\exists s \in \mathcal{S}, \forall a \in \mathcal{A} : \mu(s)(a) = a$

(3) $\mathcal{G} \subset \mathcal{A}$

$\mathcal{A}$ の元をオブジェクトと呼ぶ．$\mathcal{G}$ の元を基礎オブジェクト，$\mathcal{S}$ の元を特殊化と呼ぶ． □

$\theta \in \mathcal{S}$ のとき，$\mu(\theta)(a)$ を後置記法を用いて $a\theta$ と表すことがある．$\mu(\theta)(a) = b$ を満たす b が存在するとき，θ は a に適用可能であるという．$\theta \in \mathcal{S}$ が $\mathcal{A}$ の部分集合 B に適用可能であるとは，θ が B の任意の元に適用可能なことである．そのとき，$B\theta = \{b\theta \mid b \in B\}$ と定義する．

2.2 特殊化システムの例

特殊化システムの例を与える．これらの例は，後の章でアンフォールド変換を説明するために利用される．

例 1 [論理式と代入のなす特殊化システム]
$\mathcal{A}_1$ が通常の論理学における原子論理式の集合，$\mathcal{G}_1$ が変数を含まない原子論理式の集合，$\mathcal{S}_1$ が代入の集合，μ_1 が，任意の代入 $s \in \mathcal{S}$ に対して，s が引起こす $\mathcal{A}_1$ 上の全域写像を対応させる写像であるとすれば，$\Gamma_1 = \langle \mathcal{A}_1, \mathcal{G}_1, \mathcal{S}_1, \mu_1 \rangle$ は明らかに特殊化システムである．

例 2 [クラス束縛変数を扱う特殊化システム]
変数集合 V，述語集合 R，クラス集合 C を，

$V = \{\mathtt{X}, \mathtt{Y}\}$
$R = \{\mathtt{p}, \mathtt{q}\}$
$C = \{\mathtt{a}, \mathtt{d}, \mathtt{c}\}$

とする☆☆☆．それらの元を，それぞれ，変数，述語，クラスと呼ぶ．クラス集合 C 上の写像 set が存在し，a, d, c には，それぞれ，以下に示す集合が対応している．

$set(\mathtt{a})$ = 動物全体の集合
$set(\mathtt{d})$ = 犬全体の集合
$set(\mathtt{c})$ = 猫全体の集合

C の元 c は，$set(c)$ が集合の包含関係のもとで $set(C)$ において極小であるとき，極小クラスと呼ばれる．C の中の極小クラス全体の集合を極小クラス集合と呼ぶ．ここの例では，$D = \{\mathtt{d}, \mathtt{c}\}$ が極小クラス集合である．

☆☆ 部分写像の合成に関する定義は付録で与えている．
☆☆☆ 後で宣言的意味を簡単に記述できるように，有限の具体的な例にしている．

クラス束縛変数を，V の元と C の元のペアと定義する．それらは，X^a やY^c などのように記述する．極小クラスを用いたクラス束縛変数を，極小クラス束縛変数と呼ぶ．たとえば X^d やY^c などがその例である．この準備のもとに，$\mathcal{A}_2, \mathcal{G}_2, \mathcal{S}_2, \mu_2$ を次のように定める．

(1) $\mathcal{A}_2$ は R の元を述語とし，クラス束縛変数を引数とする "原子論理式" 全体の集合☆．たとえば p(X^a,Y^c) は $\mathcal{A}_2$ の元である．

(2) $\mathcal{G}_2$ は R の元を述語とし，極小クラス束縛変数を引数とする "原子論理式" 全体の集合．たとえば，p(X^d,Y^c) は $\mathcal{G}_2$ の元である．

(3) $\mathcal{S}_2$ は $(V \times V) \cup (V \times C)$ の元の有限列の集合．たとえば，(X,Y) と (Y,c) はそれぞれ，$V \times V$ と $V \times C$ の元である．ここでは $V \times V$ と $V \times C$ の元を，たとえば，[X/Y]，[Y▷c] のように表し，$\mathcal{S}_2$ の元を，たとえば，[X/Y] ∘ [Y▷c] のように書くことにする．$\mathcal{S}_2$ は長さ 0 の列を含むが，それは ϵ で表す．

(4) $V \times V$ と $V \times C$ の各元は $\mathcal{A}_2$ 上の部分写像を定める．たとえば，

p(X^d)[X/Y] = p(Y^d)
p(X^a)[X▷d] = p(X^d)
p(X^d)[X▷c] = 未定義

などである．一般に，[X▷ c_2] が X^c_1 に適用できるのは，$set(c_2) \subset set(c_1)$ の場合であり，適用の結果，X^c_1 は X^c_2 になる．

$\mu_2 : \mathcal{S}_2 \rightarrow partial\_map(\mathcal{A}_2)$ は次のように定義される．$\mathcal{S}_2 \ni s = (m_1, m_2, \cdots, m_k)$ のとき，$\mu_2(s)$ は $m_1, m_2, \cdots, m_k$ のそれぞれに付随する $\mathcal{A}_2$ 上の部分写像をその順序で合成した部分写像である☆☆．たとえば，

$a =$ p(X^c, Y^a, Z^d)
$s =$ [X/Y]∘[Y▷c]

のとき，

$\mu_2(s)(a) =$ p(X^c,Y^a,Z^d) [X/Y]∘[Y▷c]
$=$ p(Y^c,Y^a,Z^d) [Y▷c]
$=$ p(Y^c,Y^c,Z^d)

このとき，$\Gamma_2 = \langle \mathcal{A}_2, \mathcal{G}_2, \mathcal{S}_2, \mu_2 \rangle$ は明らかに特殊化システムである．

例 3 [集合を扱う特殊化システム]

変数集合を $V = \{\mathtt{X}, \mathtt{Y}, \mathtt{Z}\}$，定数集合を $K = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ とする．

(1) $\mathcal{A}_3 = \{\mathtt{p}(s) \mid s$ は $V \cup K$ の部分集合 $\}$

(2) $\mathcal{G}_3 = \{\mathtt{p}(s) \mid s$ は K の部分集合 $\}$

(3) $\mathcal{S}_3 = V, K$ 上の代入の集合

(4) μ_3 は代入操作によって決まる，$\mathcal{S}_3$ から $map(\mathcal{A}_3)$ への写像である．ただし $map(X)$ は X 上の写像全体の集合を表す．たとえば，$a =$ p({X,Y}), $\theta =$ {X/a,Y/a} のとき，$a\theta =$ p({a}) である．

このとき，$\Gamma_3 = \langle \mathcal{A}_3, \mathcal{G}_3, \mathcal{S}_3, \mu_3 \rangle$ は明らかに特殊化システムである．

例 4 [きわめて人工的な特殊化システム]

(1) $\mathcal{A}_4 = \{a, b, c, d, Z\}$

(2) $\mathcal{G}_4 = \{a, b, c, d\}$

(3) $\mathcal{S}_4 = \{e, s, c\}$

(4) μ_4 は $\mathcal{S}_4$ から $partial\_map(\mathcal{A}_4)$ への写像で，以下のように定義する☆☆☆．

$$\mu_4(e) = \langle \mathcal{A}_4, \mathcal{A}_4, \{(x, x) \mid x \in \mathcal{A}_4\} \rangle$$
$$\mu_4(s) = \langle \mathcal{A}_4, \mathcal{A}_4, \{(a, Z), (b, d), (c, c)\} \rangle$$
$$\mu_4(c) = \langle \mathcal{A}_4, \mathcal{A}_4, \{(c, c)\} \rangle$$

このとき，$\mu_4(e) \circ \mu_4(e) = \mu_4(e)$, $\mu_4(e) \circ \mu_4(s) = \mu_4(s)$, $\mu_4(s) \circ \mu_4(e) = \mu_4(s)$ であり，それ以外の組合せは，すべて $\mu_4(X) \circ \mu_4(Y) = \mu_4(c)$ となる．明らかに $\Gamma_4 = \langle \mathcal{A}_4, \mathcal{G}_4, \mathcal{S}_4, \mu_4 \rangle$ は特殊化システムである．

2.3 宣言的プログラム

特殊化システム上の宣言的プログラムを定義する．

定義 2 特殊化システム $\Gamma = \langle \mathcal{A}, \mathcal{G}, \mathcal{S}, \mu \rangle$ 上の確定節とは，$\mathcal{A}$ の元 $H, B_1, \cdots, B_n$ から作られた $H \leftarrow B_1, \cdots, B_n$ $(n \geq 0)$ の形の式である．H を確定節のヘッド，$B_1, \cdots, B_n$ をボディと呼ぶ．確定節に出現するオブジェクト $H, B_1, \cdots, B_n$ はしばしばアトムと呼ばれ，ボディに出現するアトムはボディアトムと呼ばれる． □

本論文で扱う節は確定節だけである．まぎれがないので，簡単のために，以下では確定節を単に節と呼ぶことがある．

$\mathcal{G}$ の元だけからなる確定節を基礎節と呼ぶ．特殊化システム $\Gamma = \langle \mathcal{A}, \mathcal{G}, \mathcal{S}, \mu \rangle$ 上の基礎節全体の集合を $Gclause(\Gamma)$ で表す．

$\theta \in \mathcal{S}$ が節 $C = (H \leftarrow B_1, B_2, \cdots, B_n)$ に適用可能であるとは，θ が H, B_1, B_2, $\cdots$, B_n のすべてに適用可能なことである．そのとき，$C\theta = (H\theta \leftarrow B_1\theta, B_2\theta, \cdots, B_n\theta)$ と定義する．$C\theta = C'$ のとき，C' は C の例であるという．C' が C の基礎例であるとは，C' が基礎節で，しかも C の例であることをいう．

☆ 本論文では，$p(t_1, t_2, \cdots, t_n)$ の形の式を "原子論理式" の呼び名で指している．

☆☆ $k = 0$ の場合は，合成してできる写像は恒等写像とする．

☆☆☆ X から Y への部分写像を $\langle X, Y, R \rangle$ の形で書く（付録参照）．

定義 3 特殊化システム $\Gamma = \langle \mathcal{A}, \mathcal{G}, \mathcal{S}, \mu \rangle$ 上の宣言的プログラム（declarative program）とは，Γ 上の確定節の集合である． □
特殊化システム Γ 上の宣言的プログラムを単にプログラムと呼ぶことがある．

例 5 例 3 の Γ_3 上の節を，

$C_{31} = (\mathtt{p(\{a,X\})} \leftarrow \mathtt{p(\{X\})})$

$C_{32} = (\mathtt{p(\{b\})} \leftarrow)$

とするとき，$P_{31} = \{C_{31}, C_{32}\}$ は Γ_3 上のプログラムである．また，例 4 の Γ_4 上の節を，$C_{41} = (b \leftarrow a)$, $C_{42} = (a \leftarrow c)$, $C_{43} = (c \leftarrow)$ とするとき，$P_{41} = \{C_{41}, C_{42}, C_{43}\}$ は Γ_4 上のプログラムである．

3. 宣言的意味と導出木

3.1 宣言的プログラムの宣言的意味

Γ 上の宣言的プログラム P に対して，P の宣言的意味 $\mathcal{M}(P)$ を，

$\mathcal{M}(P) = \bigcup_{n=1}^{\infty} [T_P]^n(\emptyset)$

で与える☆．ただし T_P は $\mathcal{G}$ のベキ集合 $2^{\mathcal{G}}$ 上の写像であり，$2^{\mathcal{G}}$ の元（すなわち $\mathcal{G}$ の部分集合）x に対して，

$$T_P(x) = \{\, head(C) \mid body(C) \subset x,\ C \in Gclause(P) \,\}$$

を対応させる．ここで，$head(C)$ は節 C のヘッド，$body(C)$ は節 C のボディに出現するアトム全体の集合，$Gclause(P)$ はプログラム P の中の節の基礎例全体の集合である．

3.2 導出木とその高さの定義

導出木とその高さの概念を帰納的に定義する．

定義 4 P をプログラム，g を $\mathcal{G}$ の要素とする．以下の条件（1）～（7）を満たす $\mathcal{A}$ 上のオブジェクト $H, B_1, B_2, \cdots, B_m$ が存在するとき，

$pt = \langle g, C, \theta, [p_1, p_2, \cdots, p_m] \rangle$

は P 上の g の導出木である．

（1） $m \geq 0$,
（2） C は P の節，
（3） $C = (H \leftarrow B_1, B_2, \cdots, B_m)$,
（4） $\theta \in \mathcal{S}$ は C に適用可能である，
（5） $g = H\theta$,
（6） $B_i\theta \in \mathcal{G} \quad (i = 1, 2, \cdots, m)$,
（7） p_i は P 上の $B_i\theta$ の導出木である．
$(i = 1, 2, \cdots, m)$.

導出木 pt の高さ $height(pt)$ を，$m = 0$ のとき 1，$m \geq 1$ のとき p_1, p_2, $\cdots$, p_m の高さの最大値に 1 を加えた数で定義する． □

定義から，$pt_1 = \langle g, C, \theta, [\,] \rangle$ が P 上の g の導出木であるのは，$\mathcal{A}$ の元 H が存在して，(1) $C = (H \leftarrow)$ は P の節，(2) $\theta \in \mathcal{S}$ は H に適用可能，(3) $g = H\theta$, の 3 つの条件を満たすときである．pt_1 の高さは 1 である．P 上の g の導出木全体の集合を $tree(g, P)$ で表す．

導出木の部分木の概念を帰納的に定義する．

定義 5 導出木

$pt = \langle g, C, \theta, [p_1, p_2, \cdots, p_m] \rangle$

の部分木とは，pt それ自身，あるいは，導出木 $p_1, p_2, \cdots, p_m$ のいずれかの部分木のことである． □
直観的には，pt の部分木とは，pt に出現する導出木を意味する．pt 自身も pt の部分木である．

pt を導出木 $\langle g, C, \theta, [p_1, p_2, \cdots, p_m] \rangle$ とするとき，C を pt の頭節と呼ぶ．S が節の集合であるとき，S の節を頭節として持つ pt の部分木の数を $\#(pt, S)$ で表す．

P 上の導出木を持つ $\mathcal{G}$ の元全体の集合を $root(P)$ と書く．P 上の n 以下の高さの導出木を持つ $\mathcal{G}$ の元全体の集合を $root(P, n)$ と書く．

$$root(P) = \{g \mid g \in \mathcal{G}, tree(g, P) \neq \emptyset\}$$
$$root(P, n) = \{g \mid g \in \mathcal{G}, \exists pt \in tree(g, P) : height(pt) \leq n\}$$

3.3 導出木と宣言的意味

導出木の概念を用いて，プログラム P の宣言的意味を規定する．まず，$root(P, n)$ と $[T_P]^n(\emptyset)$ が同一の集合であることを示す．

命題 1 任意の $n \geq 1$ に対して $root(P, n) = [T_P]^n(\emptyset)$ が成り立つ． □

これより次の定理が導かれる．

定理 1 P をプログラムとするとき，

$\mathcal{M}(P) = root(P)$

が成り立つ． □

4. 論理プログラムにおけるアンフォールド変換

論理プログラムにおけるアンフォールド変換を復習し，拡張のための問題点を示す．論理プログラムを Γ_1（例 1）上の宣言的プログラムとして扱う．

4.1 基礎概念の定義

論理プログラムのアンフォールド変換を定義するために必要な基礎概念，すなわち，リネーミング，ユニファイア，最汎ユニファイア，単一化可能性，レゾルベントなどを定義する．

☆ 宣言的プログラムの最小モデル意味論や最小不動点意味論がすでに与えられ，それらがここの宣言的意味 $\mathcal{M}(P)$ と一致する結果をもたらすことは既に証明されている（たとえば 1)）．

リネーミングとは，変数を変数に一対一に置き換える代入，すなわち，$X_1, X_2, \cdots, X_n$ が互いに異なる変数で，$Y_1, Y_2, \cdots, Y_n$ もまた互いに異なる変数であるとき，$\{X_1/Y_1, X_2/Y_2, \cdots, X_n/Y_n\}$ の形の代入である．

アトム x とアトム y のユニファイアとは，$x\theta = y\theta$ を満たす代入 θ のことである．Θ がアトム x とアトム y の**最汎ユニファイア**であるとは，Θ が x と y のユニファイアであり，しかも，$x\theta = y\theta$ なる任意の代入 θ に対して，ある代入 σ が存在して，$\Theta\sigma = \theta$ が成り立つことである．アトム x と y が**単一化可能**とは，$x\theta = y\theta$ を満たす代入 θ が存在することである．アトム x と y が**弱単一化可能**とは，片方をリネーミングすれば単一化可能になること，すなわち，$x\theta = y\rho\theta$ を満たすリネーミング ρ と代入 θ が存在することである．

節 C_1, C_2 を，

$$C_1 = (H \leftarrow A_1, \cdots, A_i, \cdots, A_m)$$
$$C_2 = (K \leftarrow B_1, \cdots, B_n)$$

とし，互いに変数を共有していないものとする．また，θ を A_i と K のユニファイアとする．そのとき，節 C_1 から，そのボディアトム A_i，節 C_2，ユニファイア θ とによって得られるレゾルベントとは，

$$C_3 = (H\theta \leftarrow A_1\theta, \cdots, A_{i-1}\theta, B_1\theta, \cdots, B_n\theta, A_{i+1}\theta, \cdots, A_m\theta)$$

で与えられる節 C_3 のことである．それは，

$$resolvent(C_1, A_i, \theta, C_2)$$

と表記される．

4.2 論理プログラムにおけるアンフォールド変換の定義

論理プログラムにおけるアンフォールド変換の定義を与える．“プログラム $P_1 = \{C_j \mid j \in J\}$ から節 C_a とそのボディアトム A_i を選んでアンフォールド変換し，P_2 を得る” とは，厳密には次のように定義される．

定義 6 [論理プログラムにおけるアンフォールド変換]

(1) $P_1 = \{C_j \mid j \in J\}$ から節 $C_a = (H \leftarrow A_1, \cdots, A_i, \cdots, A_m)$ を選ぶ．

(2) 節 C_a のボディから A_i を選ぶ．

(3) リネーミング ρ_j $(j \in J)$ を適切に選んで，$P'_1 = \{C_j\rho_j \mid j \in J\}$ を作り，P'_1 の各節 $C_j\rho_j$ が，C_a と変数を共有しないようにする．

(4) プログラム P'_1 の節のうち，A_i と単一化可能なヘッドを持つ節をすべて選ぶ．それらの節の集合を P''_1 とする．

(5) P''_1 の各節 $C_j\rho_j$ のヘッドと A_i の最汎ユニファイアの任意の1つを θ_j とする．

(6) $P_2 = (P_1 - \{C_a\}) \cup \{resolvent(C_a, A_i, \theta_j, C_j\rho_j) \mid C_j\rho_j \in P''_1\}$ により P_2 を得る．

4.3 論理プログラムのアンフォールド変換の例

節 C が，そのヘッドと弱単一化可能なボディアトムを持つとき，C は自己再帰的であるという．たとえば，特殊化システム Γ_1（例 1）上のプログラム P_{11} を

$$P_{11} = \{C_{11}, C_{12}\}$$
$$C_{11} = (\texttt{append([],Y,Y)} \leftarrow)$$
$$C_{12} = (\texttt{append([A|X],Y,[A|Z])} \leftarrow \texttt{append(X,Y,Z)})$$

とする☆．節 C_{12} は自己再帰的である．

自己再帰的な節を持つプログラムでは，節を自分自身の節によってアンフォールドする場合がありうる．プログラム $P_{11} = \{C_{11}, C_{12}\}$ のアンフォールド変換はその例となる．以下では，プログラム P_{11} から節 C_{12} とそのボディに出現するアトム append(X,Y,Z) を選んでアンフォールド変換し，その結果を求める．

例 6 [append プログラムのアンフォールド変換]

(1) プログラム P_{11} から節 C_{12} を選ぶ．

(2) 節 C_{12} のボディから append(X,Y,Z) を選ぶ．

(3) C_{11} と C_{12} に対して変数のリネーミングをして，

$$C'_{11} = (\texttt{append([],W,W)} \leftarrow)$$
$$C'_{12} = (\texttt{append([B|P],Q,[B|R])} \leftarrow \texttt{append(P,Q,R)})$$

を得る．

(4) C'_{11} と C'_{12} のうち，append(X,Y,Z) と単一化可能なヘッドを持つ節をすべて選ぶ．この場合それは，C'_{11} と C'_{12} の両方である．

(5) C'_{11} のヘッドと append(X,Y,Z) の最汎ユニファイアを1つ求める．ここでは最汎ユニファイアとして $\theta = \{\texttt{X/[]}, \texttt{Y/W}, \texttt{Z/W}\}$ を選ぶ．

(6) 節 C_{12} から，そのボディアトム append(X,Y,Z) と節 C'_{11} とによって得られるレゾルベントは，

$$C_{13} = (\texttt{append([A],W,[A|W])} \leftarrow)$$

である．

(7) 同様に，C'_{12} のヘッドと append(X,Y,Z) の最汎ユニファイアを1つ求める．ここでは最汎ユニ

☆ 本論文では DEC-10 Prolog の記法を用いる．

ファイアとして $\theta = \{$X/[B|P],Y/Q,Z/[B|R]$\}$ を選ぶ．

(8) 節 C_{12} から，そのボディアトム append(X,Y,Z) と節 C'_{12} とによって得られるレゾルベントは，

C_{14} = (append([A,B|P],Q,[A,B|R]) ← append(P,Q,R))

である．

これより，新しいプログラム P_{12} が得られる．

$P_{12} = \{C_{11}, C_{13}, C_{14}\}$

C_{11} = (append([],Y,Y) ←)

C_{13} = (append([A],W,[A|W]) ←)

C_{14} = (append([A,B|P],Q,[A,B|R]) ←
append(P,Q,R))

4.4 アンフォールド変換の拡張における問題点

ここでは，論理プログラムにおけるアンフォールド変換が，一般の宣言的プログラムの場合に比べてどのように特殊であるか，そして，論理プログラムにおけるアンフォールド変換の定義を一般の宣言的プログラムの場合に拡張しようとするとどのような問題が起こるかを観察する．

- 健全でないアンフォールド変換の存在

  論理プログラムの場合には，任意のユニファイアによるアンフォールド変換が“健全な”結果を与える．ここで健全とは，P_1 が P_2 に変換されたとき，$\mathcal{M}(P_1) \supset \mathcal{M}(P_2)$ となることをいう（定義15参照）．しかし一般の宣言的プログラムの範囲では，任意のユニファイアによる“アンフォールド変換”が常に健全な結果を生み出すとはかぎらない．たとえば，特殊化システム Γ_2（例2）において，

  C_{21}: p(X^a) ← q(X^d)

  C_{22}: q(Y^d) ←

  からなるプログラム $P_{21} = \{C_{21}, C_{22}\}$ を考える．C_{21} の q(X^d) と C_{22} の q(Y^d) とのユニファイア [X/Y] をとり，アンフォールドすれば，

  C_{23}: p(Y^a) ←

  C_{22}: q(Y^d) ←

  からなるプログラム $P_{22} = \{C_{23}, C_{22}\}$ が得られる．しかしこの変換は正しくない．実際，

  $\mathcal{M}(P_{21})$ = {p(X^d),q(X^d), p(Y^d),q(Y^d)}

  $\mathcal{M}(P_{22})$ = { p(X^d),p(X^c),q(X^d),
  p(Y^d),p(Y^c),q(Y^d)}

  であり，$\mathcal{M}(P_{21}) \not\supset \mathcal{M}(P_{22})$ である．したがって，宣言的プログラムのアンフォールド変換の定義は，健全な変換を引き起こすための条件を正しく扱えるものにする必要がある．

- 最汎ユニファイアの非存在

  論理プログラムの範囲では，2つのアトムが弱単一化可能ならばそれらの最汎ユニファイアが存在することが知られている．しかし，一般の宣言的プログラムの場合には，たとえ弱単一化可能であっても任意の2つのアトムに対して最汎ユニファイアが存在するとはかぎらない．たとえば，集合を扱う特殊化システム Γ_3（例3）において，{X/a,Y/b}と{X/b,Y/a}はp({X,Y})とp({a,b})のユニファイアである．しかしどちらも最汎ユニファイアではない．したがって，宣言的プログラムのアンフォールド変換の定義は，最汎ユニファイアが存在しないような場合をうまく包含したものであることが望ましい．

- リネーミングの問題

  論理プログラムのアンフォールド変換にはリネーミングのステップが含まれている．しかしリネーミングは一般の宣言的プログラムの観点からは特殊すぎる．宣言的プログラムの基礎である特殊化システムは，変数の代入という特別な特殊化に依存した論理プログラムの概念を一般化する意味があるが，リネーミングは変数と代入に依存した概念である．リネーミングに類する概念を導入することは，宣言的プログラムの理論の適用可能な範囲を狭める恐れがある．したがって，アンフォールド変換の定義はリネーミングを使わない定義の方が望ましい．

以上のことから，論理プログラムから宣言的プログラムへのアンフォールド変換の拡張は，単なる素朴な延長だけでは不十分であることがわかる．

- プログラム P_{21} の例は，ユニファイアとして，2つのアトムを一致させる特殊化を選ぶだけでは，健全な結果は保証されないことを示している．一般には節の他の部分（一致させるべきアトム以外の部分）に対しても条件を課す必要がある．
- $p(X)$ と $p(f(X))$ は単一化可能ではない．しかし通常のアンフォールド変換の定義では前もってリネーミングするので，節 $H(X) \leftarrow p(X)$ を節 $p(f(X)) \leftarrow$ でアンフォールドできる．リネーミングを利用しない新しいアンフォールド変換の定義を採用するなら，既存の理論でリネーミングが果たした役割を別のもので代替する必要がある．

本論文では，新しいユニファイアの定義を導入して，これらの問題を解決する．

5. 基礎概念の新しい定義

5.1 記法の約束

節 C_a, C_b を,

$$C_a = (H \leftarrow A_1, \cdots, A_{i-1}, A_i, A_{i+1}, \cdots, A_m)$$
$$C_b = (K \leftarrow B_1, \cdots, B_n)$$

とする．このとき,

(1) $(A_i, \{H, A_1, \cdots, A_{i-1}, A_{i+1}, \cdots, A_m\})$

(2) $(K, \{B_1, \cdots, B_n\})$

の2つをよく利用する．ここで (1) に出現するアトムを観察すれば，節 C_a に出現する A_i 以外のアトムが $H, A_1, \cdots, A_{i-1}, A_{i+1}, \cdots, A_m$ であるから，$(A_i; C_a)$ から (1) を決定することができる．同様に $(K; C_b)$ から (2) が一意に定まる．そこで, (1) のかわりに $(A_i; C_a)$ と，また, (2) のかわりに $(K; C_b)$ と書くことができるものと約束する．

また，次の節では，x, X, y, Y を,

$$x = A_i$$
$$X = \{H, A_1, \cdots, A_{i-1}, A_{i+1}, \cdots, A_m\}$$
$$y = K$$
$$Y = \{B_1, \cdots, B_n\}$$

と決めて，(x, X) と (y, Y) のユニファイアや基礎ユニファイアなどを導入する．

5.2 ユニファイアの定義

アトムとアトムの集合のペアに対するユニファイアの概念を新しく導入する．すなわち，x, y をアトム, X, Y をアトムの集合と仮定して，(x, X) と (y, Y) のユニファイアについて考える．

定義 7 (x, X) と (y, Y) のペアユニファイアとは,

(1) θ は X に適用可能

(2) σ は Y に適用可能

(3) $x\theta = y\sigma$

を満たす特殊化のペア (θ, σ) のことである．(x, X) と (y, Y) のペアユニファイアが存在するとき，(x, X) と (y, Y) はペア単一化可能であるといわれる． □

(x, X) と (y, Y) のペアユニファイア全体の集合を, $punif((x, X), (y, Y))$ と書くことにする．

定義 8 (x, X) と (y, Y) の基礎ペアユニファイアとは,

(1) θ は X に適用可能で，$X\theta \subset \mathcal{G}$ が成り立つ

(2) σ は Y に適用可能で，$Y\sigma \subset \mathcal{G}$ が成り立つ

(3) $x\theta = y\sigma \in \mathcal{G}$

を満たす特殊化のペア (θ, σ) のことである． □

(x, X) と (y, Y) の基礎ペアユニファイア全体の集合を, $gpunif((x, X), (y, Y))$ と書くことにする．

5.3 ペアユニファイア集合の健全性と完全性

ここでは，ペアユニファイア集合の健全性と完全性を定義する．まずその準備からはじめる．以下の直観的説明では，(x, X) や (y, Y) や $X\theta + Y\sigma$ などを節と呼ぶことにする．$X\theta + Y\sigma$ は，節 (x, X) と (y, Y) がペアユニファイア (θ, σ) によってもたらすレゾルベントを意図した式である．

節 (x, X) と節 (y, Y) から得られるレゾルベントの“意味”について考えてみる．(θ, σ) を (x, X) と (y, Y) のペアユニファイアとする．(x, X) に θ を, (y, Y) に σ を適用して，節 $(x\theta, X\theta)$ と節 $(y\sigma, Y\sigma)$ を得る．このとき $x\theta$ と $y\sigma$ が一致し，節 (x, X) と節 (y, Y) のレゾルベントとして，節 $X\theta + Y\sigma$ が得られる．これを任意の特殊化 ρ で基礎化して，基礎節 $X\theta\rho + Y\sigma\rho$ を得る．そのような基礎節全体の集合がレゾルベント $X\theta + Y\sigma$ の“意味”である．この直観をもとに，次の定義を導入する．

定義 9 (θ, σ) を (x, X) と (y, Y) のペアユニファイアとする．$set(X, \theta, Y, \sigma)$ を,

$$\{f : (X + Y) \rightarrow \mathcal{G} \mid f(z) = z\theta\rho \in \mathcal{G} \quad (z \in X), \\ f(z) = z\sigma\rho \in \mathcal{G} \quad (z \in Y), \\ \rho \in \mathcal{S}\ \}$$

によって定義する． □

たとえば，特殊化システム Γ_4 (例 4) において, $(a, \{b\})$ と $(a, \{c\})$ のペアユニファイア (e, e) に対して

$$set(\{b\}, e, \{c\}, e) = \{f_1, f_2\}$$
$$f_1 = \langle\{b, c\}, \mathcal{G}_4, \{(b, b), (c, c)\}\rangle$$
$$f_2 = \langle\{b, c\}, \mathcal{G}_4, \{(b, d), (c, c)\}\rangle$$

である．ただし，写像 f_1 は,

$$b \xrightarrow{\mu_4(e)} b \xrightarrow{\mu_4(e)} b \in \mathcal{G}_4,$$
$$c \xrightarrow{\mu_4(e)} c \xrightarrow{\mu_4(e)} c \in \mathcal{G}_4$$

から得られ，写像 f_2 は,

$$b \xrightarrow{\mu_4(e)} b \xrightarrow{\mu_4(s)} d \in \mathcal{G}_4,$$
$$c \xrightarrow{\mu_4(e)} c \xrightarrow{\mu_4(s)} c \in \mathcal{G}_4$$

より得られる．$(a, \{b\})$ と $(a, \{c\})$ のもう1つのペアユニファイア (s, s) に対しては,

$$set(\{b\}, s, \{c\}, s) = \{f_3\}$$
$$f_3 = \langle\{b, c\}, \mathcal{G}_4, \{(b, d), (c, c)\}\rangle$$

となる．写像 f_3 は,

$$b \xrightarrow{\mu_4(s)} d \xrightarrow{\mu_4(e)} d \in \mathcal{G}_4,$$
$$c \xrightarrow{\mu_4(s)} c \xrightarrow{\mu_4(e)} c \in \mathcal{G}_4$$

より得られる☆．

節 (x, X) と節 (y, Y) から基礎ペアユニファイアを

☆ 実は f_2 と同一の写像である．

用いて得られるレゾルベントについて考える．(x,X) と (y,Y) のそれぞれを θ と σ で基礎化して，基礎式 $(x\theta,X\theta)$ と $(y\sigma,Y\sigma)$ を得る．$x\theta$ と $y\sigma$ が一致するとき，それらのレゾルベントとして，$X\theta+Y\sigma$ が得られる．節 (x,X) と節 (y,Y) の任意の基礎ペアユニファイア (θ,σ) により得られる基礎節 $X\theta+Y\sigma$ 全体の集合が，正しい変換で得るべきレゾルベントに対応する結果である．この直観をもとに，次のように定義する．

定義 10 (x,X) と (y,Y) に対して，$gpunif\text{-}set(x,X,y,Y)$ を，

$$\{f:(X+Y)\to\mathcal{G} \mid f(z)=z\theta \quad (z\in X),\ f(z)=z\sigma \quad (z\in Y),\ (\theta,\sigma)\in gpunif((x,X),(y,Y))\ \}$$

によって定義する． □

たとえば，特殊化システム Γ_4（例 4）において，

$$\{(e,e)\}=gpunif((a,\{b\}),(a,\{c\}))$$
$$b\xrightarrow{\mu_4(e)}b\in\mathcal{G}_4,$$
$$c\xrightarrow{\mu_4(e)}c\in\mathcal{G}_4,$$
$$a\xrightarrow{\mu_4(e)}a\in\mathcal{G}_4$$

であるから，$gpunif\text{-}set(a,\{b\},a,\{c\})=\{f_1\}$ である．

定義 11 U を (x,X) と (y,Y) のペアユニファイアの集合とする．$Set(U)$ を

$$Set(U)=\bigcup_{(\theta,\sigma)\in U} set(X,\theta,Y,\sigma)$$

で定義する． □

たとえば，特殊化システム Γ_4 （例 4）において，$(a,\{b\})$ と $(a,\{c\})$ のペアユニファイア集合 $\{(e,e),(s,s)\}$ に対して，

$$\begin{aligned}&Set(\{(e,e),(s,s)\})\\&\quad=set(\{b\},e,\{c\},e)\cup set(\{b\},s,\{c\},s)\\&\quad=\{f_1,f_2\}\cup\{f_3\}\\&\quad=\{f_1,f_2,f_3\}\end{aligned}$$

である．

定義 12 (x,X) と (y,Y) のペアユニファイアの集合 U に対して，

- U が健全 $\Leftrightarrow$ $Set(U)\subset gpunif\text{-}set(x,X,y,Y)$
- U が完全 $\Leftrightarrow$ $Set(U)\supset gpunif\text{-}set(x,X,y,Y)$
- U が健全かつ完全 $\Leftrightarrow$ $Set(U)=gpunif\text{-}set(x,X,y,Y)$

と定義する． □

たとえば Γ_4（例 4）において，

$$gpunif\text{-}set(a,\{b\},a,\{c\})=\{f_1\}$$
$$Set(\{(e,e),(s,s)\})=\{f_1,f_2,f_3\}$$

であるから，$(a,\{b\})$ と $(a,\{c\})$ のペアユニファイア集合 $\{(e,e),(s,s)\}$ は完全である．しかし $f_1\neq f_2=f_3$ であるから，それは健全ではない．

5.4 レゾルベントの定義とペアユニファイア集合

レゾルベントの定義を改定する．

定義 13 節 C_a,C_b を，

$$C_a=(H\leftarrow A_1,\cdots,A_i,\cdots,A_m)$$
$$C_b=(K\leftarrow B_1,\cdots,B_n)$$

とする．また，(θ,σ) を $(A_i;C_a)$ と $(K;C_b)$ のペアユニファイアとする．そのとき，節 C_a から，そのボディアトム A_i，節 C_b，ペアユニファイア (θ,σ) とによって得られるレゾルベントとは，

$$\begin{aligned}C_r=(H\theta\leftarrow\ &A_1\theta,\cdots,A_{i-1}\theta,\\&B_1\sigma,\cdots,B_n\sigma,\\&A_{i+1}\theta,\cdots,A_m\theta)\end{aligned}$$

で与えられる節 C_r のことである．それは，

$$resolvent(C_a,A_i,\theta,C_b,\sigma)$$

と表記される． □

レゾルベントと健全なペアユニファイア集合の関係を述べる．

命題 2 節 C_a,C_b を，

$$C_a=(H\leftarrow A_1,\cdots,A_i,\cdots,A_m)$$
$$C_b=(K\leftarrow B_1,\cdots,B_n)$$

とする．U_b を $(A_i;C_a)$ と $(K;C_b)$ の健全なペアユニファイア集合，(θ,σ) を U_b の元とする．さらに，

$$C_r=resolvent(C_a,A_i,\theta,C_b,\sigma)$$

とする．このとき，$C_r\rho_r\in Gclause(\Gamma)$ となる任意の $\rho_r\in\mathcal{S}$ に対して，

$$C_r\rho_r=resolvent(C_a,A_i,\theta_a,C_b,\sigma_b)$$

を満たす $(A_i;C_a)$ と $(K;C_b)$ の基礎ペアユニファイア $(\theta_a,\ \sigma_b)$ が存在する． □

レゾルベントと完全なペアユニファイア集合の関係を述べる．

命題 3 節 C_a,C_b を，

$$C_a=(H\leftarrow A_1,\cdots,A_i,\cdots,A_m)$$
$$C_b=(K\leftarrow B_1,\cdots,B_n)$$

とする．U_b を $(A_i;C_a)$ と $(K;C_b)$ の完全なペアユニファイア集合とする．また (θ_a,σ_b) を $(A_i;C_a)$ と $(K;C_b)$ の基礎ペアユニファイアとする．そのとき，次の式を満たすペアユニファイア $(\theta,\sigma)\in U_b$，節 C_r と特殊化 $\rho_r\in\mathcal{S}$ が存在する．

$$C_r=resolvent(C_a,A_i,\theta,C_b,\sigma)$$
$$C_r\rho_r=resolvent(C_a,A_i,\theta_a,C_b,\sigma_b)$$

□

6. 宣言的プログラムのアンフォールド変換

6.1 アンフォールド変換の新しい定義

宣言的プログラムに対するアンフォールド変換を新しく定義しよう．“宣言的プログラム $P_1 = \{C_j \mid j \in J\}$ から節 C_a とそのボディアトム A_i を選んで，ペアユニファイア集合 U_j $(j \in J)$ を用いて，アンフォールド変換して P_2 を得る” とは，厳密には次のように定義される．

定義 14 [宣言的プログラムのアンフォールド変換]

(1) $P_1 = \{C_j \mid j \in J\}$ から節 $C_a = (H \leftarrow A_1, \cdots, A_i, \cdots, A_m)$ を選ぶ．

(2) 節 C_a のボディから A_i を選ぶ．

(3) プログラム P_1 の各節 $C_j (j \in J)$ に対して，$(A_i; C_a)$ と $(head(C_j); C_j)$ のペアユニファイア集合の任意の 1 つを U_j $(j \in J)$ とする．

(4) U_j $(j \in J)$ を用いて，

$$R_j = \{resolvent(C_a, A_i, \theta, C_j, \sigma) \mid (\theta, \sigma) \in U_j\}$$
$$R = \cup_{j \in J} R_j$$
$$P_2 = (P_1 - \{C_a\}) \cup R$$

として P_2 を得る． □

6.2 宣言的プログラムのアンフォールド変換の例

上で述べた定義にしたがって，宣言的プログラムをアンフォールド変換する例について述べる．

例 7 Γ_1（例 1）上のプログラム P_{11} を再記する．

$P_{11} = \{C_{11}, C_{12}\}$
$C_{11} = ($append([],Y,Y) $\leftarrow)$
$C_{12} = ($append([A|X],Y,[A|Z]) $\leftarrow$ append(X,Y,Z))

P_{11} において，節 C_{12} とボディアトム append(X,Y,Z) を選ぶ．この例を用いて，論理プログラムの従来のアンフォールド変換の定義を適用する過程を例 6 で示したが，ここでは，上で述べた新しい定義にしたがってアンフォールド変換する過程を示す．

節 C_{11} に対して考える．{({X/[],Z/Y},{})} は

(append(X,Y,Z),{append([A|X],Y,[A|Z])})

と

(append([],Y,Y),{})

の健全で完全なペアユニファイア集合である．このとき，

$R_5 = \{$append([A],Y,[A|Y]) $\leftarrow\}$

である．

同様に節 C_{12} に対して考えると，

{({X/[B|X],Z/[B|Z]}, {A/B})}

は

(append(X,Y,Z),{append([A|X],Y,[A|Z])})

と

(append([A|X],Y,[A|Z]),{append(X,Y,Z)})

の健全で完全なペアユニファイア集合である．このとき，

$R_6 = \{$append([A,B|X],Y,[A,B|Z]) $\leftarrow$ append(X,Y,Z)$\}$

である．

以上より，新しいプログラム

$P'_{12} = \{C_{11}, C'_{13}, C'_{14}\}$
$C_{11} = ($append([],Y,Y) $\leftarrow)$
$C'_{13} = ($append([A],Y,[A|Y]) $\leftarrow)$
$C'_{14} = ($append([A,B|X],Y,[A,B|Z]) $\leftarrow$ append(X,Y,Z))

を得る．P'_{12} が以前に述べた $P_{12} = \{C_{11}, C_{13}, C_{14}\}$ と等価なプログラムであることは明らかである．

7. アンフォールド変換の健全性と完全性

7.1 アンフォールド変換の健全性と完全性の定義

定義 15 宣言的プログラム P_1 から節 C_a とそのボディアトム A_i を選んでアンフォールド変換して P_2 を得たとする．そのとき，

- 変換が健全 $\Leftrightarrow \mathcal{M}(P_1) \supset \mathcal{M}(P_2)$
- 変換が完全 $\Leftrightarrow \mathcal{M}(P_1) \subset \mathcal{M}(P_2)$
- 変換が健全かつ完全 $\Leftrightarrow \mathcal{M}(P_1) = \mathcal{M}(P_2)$

と定義する． □

7.2 健全性の定理

本節以降では，定義 14 に出現する記号を仮定して議論する．健全性の定理は，ペアユニファイア集合 U_j $(i \in J)$ の健全性のもとで，P_1 から P_2 へのアンフォールド変換の健全性が成り立つことを主張する定理である（定理 2）．P_1 から P_2 へのアンフォールド変換が健全であることは，$\mathcal{M}(P_1) \supset \mathcal{M}(P_2)$ で定義され，$root(P_1) \supset root(P_2)$ と等価であるから，健全性の定理を証明するためには，P_2 上の g の導出木から P_1 上の g の導出木を構成すればよい．$P_2 = (P_1 - \{C_a\}) \cup R$ であるから，P_2 上の g の導出木の中の $(R - P_1)$ 節☆を 1 つずつ P_1 節に置き換え，結果として P_1 上の g の導出木を得ればよい．

命題 4 pt_1 を $P_1 \cup R$ 上の g の導出木，$\#(pt_1, R - P_1) = k + 1 \geq 1$ とする．もし U_j $(j \in J)$ がすべて健全ならば，$\#(pt_2, R - P_1) = k$ を満たす $P_1 \cup R$ 上の g の導出木 pt_2 が存在する． □

この命題を繰り返し適用すれば次の命題を得る．

☆ S を節の集合とするとき，S に含まれる節を S 節と呼ぶ．

命題 5 pt_x を $P_1 \cup R$ 上の g の任意の導出木とする．もし，$U_j\,(j \in J)$ がすべて健全ならば，P_1 上の g の導出木 pt_y が存在する． □

アンフォールド変換の健全性定理を与える．

定理 2 宣言的プログラム P_1 を，P_1 の節 C_a，そのボディアトム A_i, ペアユニファイア集合 $U_j\,(j \in J)$ を用いてアンフォールド変換し，その結果 P_2 が得られたとする．もしすべての $U_j\,(j \in J)$ が健全ならば，アンフォールド変換は健全である，すなわち，$\mathcal{M}(P_1) \supset \mathcal{M}(P_2)$ が成り立つ． □

7.3 完全性の定理

完全性の定理は，ペアユニファイア集合 $U_j\,(i \in J)$ の完全性のもとで，P_1 から P_2 へのアンフォールド変換の完全性が成り立つことを主張する定理である（定理 3）．P_1 から P_2 へのアンフォールド変換が完全であることは，$\mathcal{M}(P_1) \subset \mathcal{M}(P_2)$ と定義され，$root(P_1) \subset root(P_2)$ と等価である．したがって，完全性の定理を証明するためには，P_1 上の g の導出木から P_2 上の g の導出木を構成すればよい．

$C_a \in R$ のときは，P_1 上の導出木はそのままで P_2 上の導出木となるので，$C_a \notin R$ のときを考える．$P_2 = (P_1 - \{C_a\}) \cup R$ であるから，P_1 上の g の導出木のなかの $\{C_a\}$ 節を 1 つずつ R 節に置き換え，結果として P_2 上の（すなわち $\{C_a\}$ 節を持たない）g の導出木を得ればよい．

この置き換えは，もし $\{C_a\}$ 節を祖先から子孫に向かう順に置き換えなければ，最後まで実行できない恐れがある．それは $\{C_a\}$ 節を取り除くためにはその“子供節”（証明における $C_b \in P$）を必要とするからである．

命題 6 $C_a \notin R$を仮定する．pt_1 が $P_1 \cup R$上の g の導出木であり，pt_1 の部分木で頭節が C_a であるものは，すべて P_1 上の導出木であるとする．$\#(pt_1, \{C_a\}) = k+1 \geq 1$ と仮定する．もし $U_j\,(j \in J)$ がすべて完全ならば，$P_1 \cup R$ 上の g の導出木 pt_2 が存在し，$\#(pt_2, \{C_a\}) \leq k$ を満たし，pt_2 の部分木で頭節が C_a であるものは，すべて P_1 上の導出木である． □

この命題を繰り返し適用すれば次の命題を得る．

命題 7 $C_a \notin R$ を仮定する．pt_x が $P_1 \cup R$ 上の g の導出木であり，pt_x の部分木で頭節が C_a であるものは，すべて P_1 上の導出木であるとする．もし $U_j\,(j \in J)$ がすべて完全ならば，$P_1 \cup R$ 上の g の導出木 pt_y が存在して，頭節が C_a である部分木を持たない． □

アンフォールド変換の完全性定理を証明する．

定理 3 宣言的プログラム P_1 を，P_1 の節 C_a，そのボディアトム A_i, ペアユニファイア集合 $U_j\,(j \in J)$ を用いてアンフォールド変換した結果 P_2 が得られたとする．もしすべての $U_j\,(j \in J)$ が完全ならば，アンフォールド変換は完全である，すなわち，$\mathcal{M}(P_1) \subset \mathcal{M}(P_2)$ が成り立つ． □

7.4 健全かつ完全なアンフォールド変換

定理 2 と定理 3 を合わせて，次の定理を得る．

定理 4 すべての $U_j\,(j \in J)$ が健全かつ完全ならば，P_1 から P_2 へのアンフォールド変換は健全かつ完全である． □

健全かつ完全なアンフォールド変換の例をあげる．

例 8 例 3 の，集合を扱う特殊化システム $\Gamma_3 = \langle \mathcal{A}_3, \mathcal{G}_3, \mathcal{S}_3, \mu_3 \rangle$ 上の節を，

$C_{33} = (\texttt{p(\{a,X\})} \leftarrow \texttt{p(\{X,Y\})})$

$C_{34} = (\texttt{p(\{a,b\})} \leftarrow)$

とし，$P_{32} = \{C_{33}, C_{34}\}$ とおく．P_{32} において，節 C_{33} とボディアトム $\texttt{p(\{X,Y\})}$ を選び，アンフォールド変換する．

C_{33} による展開を考える．$(\texttt{p(\{X,Y\})}, \{\texttt{p(\{a,X\})}\})$ と $(\texttt{p(\{a,X\})}, \{\texttt{p(\{X,Y\})}\})$ の健全かつ完全なペアユニファイア集合の 1 つは，

`{({X/a,Y/X},{ }),({Y/a},{ })}`

である．そのとき，

$C_{35} = (\texttt{p(\{a\})} \leftarrow \texttt{p(\{X,Y\})})$

$C_{36} = (\texttt{p(\{a,X\})} \leftarrow \texttt{p(\{X,Y\})})$

が得られる．

C_{34} による展開を考える．$(\texttt{p(\{X,Y\})}, \{\texttt{p(\{a,X\})}\})$ と $(\texttt{p(\{a,b\})}, \{\})$ の健全かつ完全なペアユニファイア集合の 1 つは，

`{({X/a,Y/b},{ }),({X/b,Y/a},{ })}`

である．そのとき，

$C_{37} = (\texttt{p(\{a\})} \leftarrow)$

$C_{38} = (\texttt{p(\{a,b\})} \leftarrow)$

が得られる．

以上より，$P_{33} = \{C_{34}, C_{35}, C_{36}, C_{37}, C_{38}\}$ が得られる．

8. 比　　較

8.1 プログラム変換との比較

アンフォールド変換は，そもそも関数型や論理型のプログラムにおけるプログラム変換の研究で提案された．プログラム変換は，正しいことが保証されたプログラムから，逐次的に等価変形を繰り返して，効率的なプログラムをつくり出すためのソフトウエア開発法として，1970 年代よりこれまで，多くの研究が行われてきた（文献 5)，6)，10)，12) 他）．

アンフォールド変換は，プログラムの変換において最も基本的な変換方法の1つであり，フォールド変換と組み合わせてよく利用されている．論理プログラムに対しては，Tamaki と Sato がアンフォールド／フォールド変換を提案し，それが最小モデルを保存することを示した[17]．Kawamura と Kanamori は，その結果を拡張し，その変換が成功集合をも保存することを示した[7]．また Seki は，フォールド変換を修正し，有限失敗集合を保存する変換方法を提案し，層状プログラムの正しい変換を達成した[14]．

以上の研究は，超論理述語や副作用のある述語やカットなどを含まない“純粋な”論理プログラムを対象としている．それに対して Sahlin は完全な Prolog（超論理述語や副作用のある述語やカットなどを含む実際の Prolog）に対する部分評価の研究を行っている[13]．これは Prolog の手続き的意味を保存した変換であり，実際の Prolog プログラムに適用可能なものである．

プログラム変換とはそもそも実行前にプログラムの効率を改善する技術である．それに対して等価変換パラダイムでは等価変換を，プログラムの改善ではなく，実際の計算を行うものと位置付けている．両者の数学的な意味での共通部分は,「確定節集合の等価変換」である．しかしそれに対する位置付けは異なる．完全な Prolog のプログラム変換からみれば,「確定節集合の等価変換」の研究は次の意味で不十分である．

(1) 確定節集合は実際の Prolog の宣言的近似に過ぎない．
(2) 宣言的意味の保存は実際の Prolog の手続き的意味の保存を正しく保証しない．

一方，等価変換パラダイムからみれば,「確定節集合の等価変換」はまさに研究すべき対象である．というのは，確定節集合は重要な表現システムの1つであり，その意味を保存する変換は等価変換における計算に他ならないからである．等価変換パラダイムは問題の宣言的記述を変換するものであり，実際の Prolog のような手続きの記述を変換することは考えない．等価変換パラダイムでは純粋に宣言的な記述を変換するにもかかわらず，豊富なデータ構造，多彩なルール，柔軟な制御などのために正当で高速な実行が可能である[3),15]．これは，論理パラダイムが高速化のために非論理的な要素（上述のカットなど）を取り入れざるを得ず，そのために実行の正当性の保証や，高度なプログラム変換の達成を困難にしたのと対照的である．

8.2 一 般 性

従来の論理プログラムのプログラム変換との重要な違いの1つは，理論の一般性である．宣言的プログラムは非常に一般的な宣言的表現のクラスを形成している．(純粋な）論理プログラムだけでなく，従来提案され議論されてきた多くの知識表現を，特定のクラスの特殊化システムを用いた特殊な宣言的プログラムとして捉えることができる（文献 2), 3) など）．

本理論に基づけば，それらの知識表現における正当なアンフォールド変換は，以下の手順で達成できる．

(1) 適切な特殊化システム（のクラス）を想定することによって，与えられた知識表現を宣言的プログラムの部分クラスとして捉える．
(2) その特殊化システムのもとで,（健全かつ完全な）ペアユニファイア集合を与えるアルゴリズムを提案する．
(3) そのアルゴリズムが本当に健全かつ完全なペアユニファイア集合を与えることを本論文の定義に基づいて証明する．
(4) 上ですでに正当化されたアルゴリズムを用いて計算し，健全かつ完全なアンフォールド変換を達成する．

本論文で与えた健全かつ完全なペアユニファイア集合の定義は十分一般的な形をしており，特殊化システムで特徴付けることのできる多くの有用な領域（項や文字列の領域，クラス変数や制約を扱う領域など）において，上記の手順を用いて正当なアンフォールド変換が達成できると考えられる．

8.3 論理プログラムへの本理論の適用

個別の特殊化システムのもとで健全かつ完全なペアユニファイア集合を得る方法についての議論は本論文の範囲外であるが，伝統的な論理の領域における最汎ユニファイア（mgu）によるアンフォールド変換を，本理論の枠組に当てはめる方法の概略を述べる．

まず，通常の論理プログラムを，特殊化システム Γ_1（例 1）上の宣言的プログラムとみなす．2つの節を (x, X) と (y, Y) とするとき，x と y が弱単一化可能でなければ，(x, X) と (y, Y) の健全かつ完全なペアユニファイア集合は空集合である．x と y が弱単一化可能であれば，(x, X) と (y, Y) の健全かつ完全なペアユニファイア集合の一例として $(\theta, \rho\theta)$ からなる単元集合を選ぶことができる．ただし，ρ は節 (x, X) と節 (y, Y) の共有変数をなくすために節 (y, Y) に作用させるリネーミングであり，θ は x と $y\rho$ の最汎ペアユニファイアである．

従来の理論は，健全かつ完全なペアユニファイア集合の求め方の特殊例を与えている．一方本論文の定理は，もっと緩い条件（健全かつ完全なペアユニファイ

ア集合）の下でアンフォールド変換が宣言的意味を保存することを示している．

9. むすび

本論文では，論理プログラムのアンフォールド変換の定義を拡張し，宣言的プログラムのアンフォールド変換を定義した．ユニファイアは，通常の論理においては $x\theta = y\theta$ を満たす代入 θ と定義されるが，この定義はリネーミングを前提にしなければアンフォールド変換の正当性（完全性）に不都合を引き起こす．しかし十分広い範囲で正当性を持つためにどのようなアンフォールド変換が必要かを基本に立ち戻って考えると，リネーミング（あるいはその拡張概念）をアンフォールド変換の手続きの中に持ち込む必然性はないことがわかる．本論文では，ユニファイアの概念を，(1) アトム同士に対するものから，アトムとアトムの集合のペア同士に対するものへ，また，(2) 単一の代入（特殊化）から特殊化のペアへ，と変更した．新しいユニファイアの定義へのアトム集合の導入は，単一化に当たって，同一にする２つのアトムだけでなく節全体を操作することを可能にする．また，特殊化のペアによるユニファイア（ペアユニファイア）は，２つの節を別々に操作することを可能にする．これによって,「リネーミングの効果を含んだ単一化」を行うことができ，リネーミングを使わない汎用のアンフォールド変換が達成された．

このような新しいペアユニファイアの定義のもとに，健全なペアユニファイア集合と完全なペアユニファイア集合の概念を導入した．また，プログラムの宣言的意味を用いて，アンフォールド変換の健全性と完全性を定義した．健全なペアユニファイア集合を使えば，アンフォールド変換は健全である．完全なペアユニファイア集合を使えば，アンフォールド変換は完全である．健全かつ完全なペアユニファイア集合を使えば，アンフォールド変換は健全かつ完全である．

参考文献

1) Akama, K.: Logical Formalization of Problems Using Declarative Programs, *Hokkaido University Information Engineering Technical Report*, HIER-LI-9504 (1995).
2) 赤間清, 繁田良則, 宮本衛市: 特殊化システムの拡張による知識表現系の変更, 人工知能学会誌, Vol.13, No.1, pp.131–138 (1998).
3) 赤間清, 清水伴訓, 宮本衛市: 宣言的プログラムの等価変換による問題解決, 人工知能学会誌, Vol.13, No.6, pp.944–952 (1998).
4) 赤間清, 川口雄一, 宮本衛市: 論理的問題の等価変換による解法（2）, SLD 導出の限界, 人工知能学会誌, Vol.13, No.6, pp.936–943 (1998).
5) Burstall, R.M. and Darlington, J.: A Transformation System for Developing Recursive Programs, *J. ACM*, Vol.24, No.1, pp.44–67 (1977).
6) Futamura, Y.: Partial evaluation of computation process - an approach to a compiler-compiler, *Systems, Computers, Controls*, Vol.2, No. 5, pp. 45–50 (1971).
7) Kawamura, T. and Kanamori, T.: Preservation of Stronger Equivalence in Unfold/Fold Logic Program Transformation, *Proc. of International Conference on Fifth Generation Computer Systems 1988*, ICOT, Tokyo, pp. 413–421 (1988).
8) 小池英勝, 赤間清, 宮本衛市: 仕様からの等価変換ルールの生成法, 電子情報通信学会技術研究報告 KBSE98-8, pp33–40 (1998).
9) Lloyd, J. W.: *Foundations of Logic Programming*, 2nd edition, Springer-Verlag, 212pp. (1987).
10) Manna, Z. and Waldinger, R. J.: Synthesis: Dreams *Longrightarrow* Programs, *IEEE Trans. on Software Engineering*, Vol. 5, No. 4, pp. 294–328 (1979).
11) 岡田浩一, 赤間清, 宮本衛市: 否定を含む宣言的プログラムのプログラム変換, 電子情報通信学会技術研究報告, ソフトウエアサイエンス研究会, SS 95-16, pp. 47–54 (1995).
12) Pettorossi, A. and Proietti, M.: Transformation of Logic Programs: Foundations and Techniques, *J. Logic Programming*, Vol. 19/20, pp. 261–320 (1994).
13) Sahlin, D.: An Automatic Partial Evaluator for Full Prolog, SICS Dissertation Series 04, 170pp. (1991).
14) Seki, H.: Unfold/fold Transformation of Stratified Programs, *Theoretical Computer Science*, Vol. 86, pp. 107–139 (1991).
15) 吹田慶子, 赤間清, 宮本衛市: 等価変換による制約充足問題の解法, 電子情報通信学会ソフトウエアサイエンス研究会, SS 96-18, pp. 1–8 (1996).
16) Takeuchi, A. and Fujita, H.: Competitive Partial Evaluation, *Workshop on Partial Evaluation and Mixed Computation*, pp. 317–326, October (1987).
17) Tamaki, H. and Sato, T.: Unfold/fold Transformation of Logic Programs, *Proc. 2nd Int. Logic Programming Conf.*, Uppsala Univ., pp. 127–138 (1984).
18) Yoshida, T., Akama, K. and Miyamoto, E.: Problem Solving by Equivalent Transformation of First Order Logical Constraints, Preprints

Work. Gr. for ICS, IPSJ 98-ICS-114-3, pp. 13–18 (1998).

19) van Emden, M.H. and Kowalski, R.A.: The Semantics of Predicate Logic as a Programming Language, *J. ACM*, Vol. 23, No. 4, pp. 733–742 (1976).

付　　録

(1) 部分写像に関する定義

集合 X から Y への部分写像とは，(1) $R \subset X \times Y$，(2) $(\forall x, y_1, y_2) : [(x, y_1), (x, y_2) \in R \to y_1 = y_2]$ を満たす 3 項組 $\langle X, Y, R \rangle$ である．部分写像 $f = \langle X, Y, R \rangle$ が $x \in X$ に適用可能であるとは，$(x, y) \in R$ を満たす $y \in Y$ が存在することである．部分写像 $f = X, Y, R$ に対して，$f(x) = y$ とは，$(x, y) \in R$ を意味する．また $f(x) =$ 未定義 とは，f が x に適用可能でないことを意味する．X から Y への部分写像全体の集合を $partial\_map(X, Y)$ で，$partial\_map(X, X)$ を $partial\_map(X)$ で表す．部分写像の合成は $\circ$ で表す．すなわち，部分写像 $f = \langle X, Y, R_f \rangle$，$g = \langle Y, Z, R_g \rangle$ に対して，

$$f \circ g = \langle X, Z, \{(x, z) \mid (x, y) \in R_f, (y, z) \in R_g\} \rangle$$

である．

(2) 命題 1 の証明

帰納法で示す．

(a) $n = 1$ のとき命題が成り立つことを示す．

$$\begin{aligned} & g \in root(P, 1) \\ & \quad \leftrightarrow \exists\, C, \theta, H : \\ & \qquad C = (H \leftarrow) \in P, \\ & \qquad \theta \text{ が } C \text{ に適用可能}, \\ & \qquad g = H\theta, \\ & \quad \leftrightarrow g \in [T_P](\emptyset). \end{aligned}$$

(b) $n = k$ に対して命題が成り立つことを仮定し，$n = k + 1$ に対して命題が成り立つことを示す．

$$\begin{aligned} & g \in root(P, k+1) \\ & \quad \leftrightarrow \exists\, C, \theta, H, B_i\, (1 \le i \le m) : \\ & \qquad C = (H \leftarrow B_1, B_2, \cdots, B_m) \in P, \\ & \qquad \theta \text{ が } C \text{ に適用可能}, \\ & \qquad g = H\theta, \\ & \qquad B_i\theta \in root(P, k) \quad (1 \le i \le m) \\ & \quad \leftrightarrow \exists\, C, \theta, H, B_i\, (1 \le i \le m) : \\ & \qquad C = (H \leftarrow B_1, B_2, \cdots, B_m) \in P, \\ & \qquad \theta \text{ が } C \text{ に適用可能}, \\ & \qquad g = H\theta, \\ & \qquad B_i\theta \in [T_P]^k(\emptyset) \quad (1 \le i \le m) \\ & \quad \leftrightarrow g \in [T_P]^{k+1}(\emptyset). \end{aligned}$$

(3) 定理 1 の証明

命題 1 を用いる．

$$\begin{aligned} \mathcal{M}(P) &= \cup_{n=1}^{\infty} [T_P]^n(\emptyset) \\ &= \cup_{n=1}^{\infty} root(P, n) \\ &= root(P) \end{aligned}$$

(4) 命題 2 の証明

x, X, y, Y を，

$$\begin{aligned} & x = A_i \\ & X = \{H, A_1, \cdots, A_{i-1}, A_{i+1}, \cdots, A_m\} \\ & y = K \\ & Y = \{B_1, \cdots, B_n\} \end{aligned}$$

とする．仮定より

$$\begin{aligned} & (\theta, \sigma) \in U_b, \\ & C_r = resolvent(C_a, A_i, \theta, C_b, \sigma) \end{aligned}$$

である．

$$C_r\rho_r \in Gclause(\Gamma)$$

を満たす任意の特殊化を $\rho_r \in \mathcal{S}$ とする．そのとき，

$$\begin{aligned} & z\theta\rho_r \in \mathcal{G} \quad (z \in X) \\ & z\sigma\rho_r \in \mathcal{G} \quad (z \in Y) \end{aligned}$$

が成り立つ．したがって，写像 $f : (X + Y) \to \mathcal{G}$ が，

$$\begin{aligned} & f(z) = z\theta\rho_r \quad (z \in X) \\ & f(z) = z\sigma\rho_r \quad (z \in Y) \end{aligned}$$

によって定義できる．f は $set(X, \theta, Y, \sigma)$ の元であり，さらに，$Set(U_b)$ の元であることがわかる．一方，U_b が $(A_i; C_a)$ と $(K; C_b)$ の健全なペアユニファイア集合だから，

$$Set(U_b) \subset gpunif\text{-}set(x, X, y, Y)$$

が成り立つ．したがって，f は $gpunif\text{-}set(x, X, y, Y)$ の元である．$gpunif\text{-}set(x, X, y, Y)$ の定義より，$(\theta_a, \sigma_b) \in gpunif((x, X), (y, Y))$ を満たす θ_a, σ_b が存在して，

$$\begin{aligned} & f(z) = z\theta_a \quad (z \in X) \\ & f(z) = z\sigma_b \quad (z \in Y) \end{aligned}$$

となる．よって，

$$\begin{aligned} & z\theta\rho_r = z\theta_a \quad (z \in X) \\ & z\sigma\rho_r = z\sigma_b \quad (z \in Y) \end{aligned}$$

がいえるから，

$$C_r\rho_r = resolvent(C_a, A_i, \theta_a, C_b, \sigma_b)$$

が成り立つ．

(5) 命題 3 の証明

x, X, y, Y を，

$$\begin{aligned} & x = A_i \\ & X = \{H, A_1, \cdots, A_{i-1}, A_{i+1}, \cdots, A_m\} \\ & y = K \\ & Y = \{B_1, \cdots, B_n\} \end{aligned}$$

とする．(θ_a, σ_b) が $(A_i; C_a)$ と $(K; C_b)$ の基礎ペアユニファイアだから，

$$\begin{aligned} & f(z) = z\theta_a \quad (z \in X) \\ & f(z) = z\sigma_b \quad (z \in Y) \end{aligned}$$

によって $f : (X+Y) \to \mathcal{G}$ を定義でき，f は $gpunif\text{-}set(x, X, y, Y)$ の元である．一方，U_b が $(A_i; C_a)$ と $(K; C_b)$ の完全なペアユニファイア集合だから，

$$Set(U_b) \supset gpunif\text{-}set(x, X, y, Y)$$

が成り立つ．したがって，f は $Set(U_b)$ の元である．$Set(U_b)$ の定義より，$(\theta, \sigma) \in U_b$ が存在して，$f \in set(X, \theta, Y, \sigma)$ が成り立つ．これは，

$$\begin{aligned} & f(z) = z\theta\rho_r \in \mathcal{G} \quad (z \in X) \\ & f(z) = z\sigma\rho_r \in \mathcal{G} \quad (z \in Y) \end{aligned}$$

を満たす $\rho_r \in \mathcal{S}$ が存在することを意味する．f の定義より，

$$\begin{aligned} & z\theta\rho_r = z\theta_a \quad (z \in X) \\ & z\sigma\rho_r = z\sigma_b \quad (z \in Y) \end{aligned}$$

がいえる．$(\theta, \sigma) \in U_b$ より，

$$C_r = resolvent(C_a, A_i, \theta, C_b, \sigma)$$

となる節 C_r が存在し，

$$C_r\rho_r = resolvent(C_a, A_i, \theta_a, C_b, \sigma_b)$$

が成り立つ．

(6) 命題 4 の証明

pt_1 から新しい導出木 pt_2 を次のように構成する．

(a) $\#(pt_1, R - P_1) = k + 1 \ge 1$ より，pt_1 の部分木で，$R - P_1$ の節 C_r を頭節とする導出木 pt_{old} が存在する．

$$pt_{old} = \langle g_r, C_r, \rho_r, [r_1, r_2, \cdots, r_r] \rangle$$

$C_r \in R$ であるから，節 $C_b = (K \leftarrow B_1, \cdots, B_n) \in P_1$，$(A_i; C_a)$ と $(K; C_b)$ の健全なペアユニファイア集合 U_b，ペアユニファイア $(\theta, \sigma) \in U_b$ が存在して，

$$C_r = resolvent(C_a, A_i, \theta, C_b, \sigma)$$

が成り立つ．pt_{old} は導出木であるから，$C_r\rho_r$ は基礎節である．よって命題 2 より，$(A_i; C_a)$ と $(K; C_b)$ の基礎ペアユニファイア (θ_a, σ_b) が存在して，

$$C_r\rho_r = resolvent(C_a, A_i, \theta_a, C_b, \sigma_b)$$

が成り立つ．

(b) 以下の式により，$P_1 \cup R$ 上の導出木 pt_{new} を定める．

$$\begin{aligned} &pt_{new} = \langle g_r, C_a, \theta_a, s\rangle,\\ &s = [p_1, p_2, \cdots, p_{i-1}, p_i, p_{i+1}, \cdots, p_m],\\ &[p_1, p_2, \cdots, p_{i-1}] = [r_1, r_2, \cdots, r_{i-1}]\\ &p_i = \langle K\sigma_b, C_b, \sigma_b, [q_1, q_2, \cdots, q_n]\rangle,\\ &[q_1, q_2, \cdots, q_n] = [r_i, r_{i+1}, \cdots, r_{i+n-1}],\\ &[p_{i+1}, \cdots, p_m] = [r_{i+n}, r_{i+n+1}, \cdots, r_r]. \end{aligned}$$

(c) pt_1 の部分木 pt_{old} を pt_{new} に置き換えることによって，$P_1 \cup R$ 上の g の導出木 pt_2 を得る．

明らかに $\#(pt_2, R - P_1) = k$ である．

(7) 命題 5 の証明

$\#(pt_x, R - P_1)$ に関する帰納法を使う．

(a) もし $\#(pt_x, R - P_1) = 0$ ならば，pt_x 自身が P_1 上の g の導出木である．

(b) $\#(pt, R - P_1) = k \geq 0$ を満たす $P_1 \cup R$ 上の g の任意の導出木 pt について命題が成り立つことを仮定する．$\#(pt_x, R - P_1) = k + 1$ とする．命題 4 より，$\#(pt_z, R - P_1) = k$ を満たす $P_1 \cup R$ 上の g の導出木 pt_z が存在する．したがって帰納法の仮定より，P_1 上の g の導出木 pt_y が存在する．

(8) 定理 2 の証明

$g \in \mathcal{M}(P_2)$ を仮定する．$g \in root(P_2)$ がいえるから，P_2 上の g の導出木 pt_x が存在する．$P_2 = (P_1 - \{C_a\}) \cup R$ なので，pt_x は $P_1 \cup R$ 上の g の導出木である．命題 5 を用いれば，P_1 上の g の導出木 pt_y が存在することがわかる．したがって $g \in root(P_1)$ がいえ，$g \in \mathcal{M}(P_1)$ が導かれる．

(9) 命題 6 の証明

pt_1 から新しい導出木 pt_2 を次のようにして作る．

(a) pt_{old} を，pt_1 の部分木のうち，頭節が C_a である極大な導出木とする☆．

$$pt_{old} = \langle g_a, C_a, \theta_a, [p_1, p_2, \cdots, p_m]\rangle$$

仮定より，pt_{old} は，P_1 上の導出木である．したがって，$p_1, p_2, \cdots, p_m$ も P_1 上の導出木である．

(b) pt_{old} の i 番目の部分木 p_i を

$$p_i = \langle g_b, C_b, \sigma_b, [q_1, q_2, \cdots, q_n]\rangle$$

とする．p_i は P_1 上の導出木であるから，$C_b \in P_1$ が成り立つ．また，$q_1, q_2, \cdots, q_n$ も P_1 上の導出木である．節 C_b を

$$C_b = (K \leftarrow B_1, \cdots, B_n)$$

とする．p_{old} は導出木であるから，(θ_a, σ_b) は $(A_i; C_a)$ と $(K; C_b)$ の基礎ペアユニファイアである．命題 3 より，ペアユニファイア $(\theta, \sigma) \in U_b$，節 C_r，特殊化 $\rho_r \in \mathcal{S}$ が存在して，

$$C_r = resolvent(C_a, A_i, \theta, C_b, \sigma)$$

$$C_r\rho_r = resolvent(C_a, A_i, \theta_a, C_b, \sigma_b)$$

が成り立つ．このとき，$C_r \in R$ であり，

$$pt_{new} = \langle g_a, C_r, \rho_r, [\ p_1, p_2, \cdots, p_{i-1}, q_1, q_2, \cdots, q_n, p_{i+1}, \cdots, p_m]\rangle$$

は $P_1 \cup R$ 上の導出木である．

(c) pt_1 の部分木 pt_{old} を pt_{new} に置き換えて，$P_1 \cup R$ 上の g の導出木 pt_2 を得る．

$C_r \in R$ と $C_a \notin R$ より，$C_r \neq C_a$ が成り立つ．pt_1 から pt_2 への変化では C_a と C_b が C_r に変わるので，pt_2 の部分木で頭節が C_a であるものは，1 つ以上減少する（$C_b = C_a$ のときは 2 つ減少する）．よって明らかに $\#(pt_2, \{C_a\}) \leq k$ である．また，pt_2 の部分木で頭節が C_a であるものは，P_1 上の導出木である．

(10) 命題 7 の証明

$n = \#(pt_x, \{C_a\})$ に関する帰納法を使う．

(a) $n = 0$ のとき，pt_x 自身が，頭節が C_a である部分木を持たない $P_1 \cup R$ 上の g の導出木である．

(b) 命題が $0 \leq n \leq k$ のとき成り立つことを仮定して，$n = k + 1$ でも成り立つことを言う．$\#(pt_x, \{C_a\}) = k + 1 \geq 1$ とする．命題 6 より，$P_1 \cup R$ 上の g の導出木 pt_z で，次の条件を満たすものが存在する．

(i) $\#(pt_z, \{C_a\}) \leq k$

(ii) pt_z の部分木で頭節が C_a であるものはすべて P_1 上の導出木である．

したがって帰納法の仮定より，$P_1 \cup R$ 上の g の導出木 pt_y が存在して，頭節が C_a である部分木を持たない．

(11) 定理 3 の証明

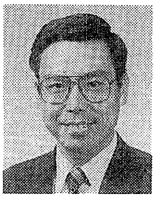
$g \in \mathcal{M}(P_1)$ とする．これは $g \in root(P_1)$ と等価であり，P_1 上の g の導出木 pt_x が存在する．もし $C_a \in R$ ならば，pt_x は $P_2 = (P_1 - \{C_a\}) \cup R$ 上の g の導出木である．次に $C_a \notin R$ の場合を考える．pt_x は $P_1 \cup R$ 上の g の導出木であり，pt_x の部分木で頭節が C_a であるものは，P_1 上の導出木である．したがって命題 7 より，$P_1 \cup R$ 上の g の導出木 pt_y が存在して，頭節が C_a である部分木を持たない．明らかに，pt_y は $P_2 = (P_1 - \{C_a\}) \cup R$ 上の g の導出木である．よって，$C_a \in R$ の場合にも，$C_a \notin R$ の場合にも $g \in root(P_2)$ が成り立ち，$g \in \mathcal{M}(P_2)$ がいえる．

(12) 定理 4 の証明

定理 2 と定理 3 より明らか．

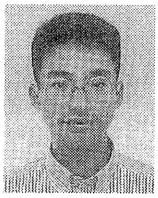
(平成 10 年 12 月 25 日受付)
(平成 11 年 4 月 23 日採録)

☆ 極大性は，部分木をつくる包含関係に基づいて判定する．



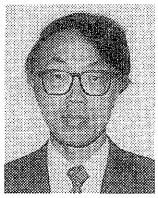
赤間　清（正会員）

1973 年東工大工学部卒業．1975 年同大大学院修士課程修了．1979 年同大大学院博士課程単位修得退学．同年同大学助手．1981 年北海道大学文学部講師，1989 年同大学工学部助教授，1999 年より同大学情報メディア教育研究総合センター教授として現在に至る．人工知能，知識処理，等価変換に基づく問題解決の研究に従事．工学博士．人工知能学会，ソフトウエア科学会，情報処理学会，認知科学会，各会員．



繁田　良則（正会員）

1990 年北大工学部卒業．1992 年同大大学院修士課程修了．1995 年同大大学院博士課程単位修得退学．同年 (株) 東芝入社．プログラミング言語，言語処理系などに興味を持つ．



宮本　衛市（正会員）

1962 年北大工学部卒業．1964 年同大大学院修士課程修了．同年同大学講師．同助教授を経て，1984 年より北海道大学工学部教授として現在に至る．分散システム，並列オブジェクト指向モデル，プログラミング環境などの研究に従事．工学博士．著書に「PASCAL －プログラミングと翻訳技法」など．情報処理学会，ソフトウエア科学会，IEEE，各会員．