2U－4

制約オブジェクトの単一化を用いたプログラム変換

大松 正樹† 赤間 清†† 宮本 衛市††

†学習情報通信システム研究所 ††北海道大学

1 はじめに

本論文では制約論理プログラムに対するプログラム変換の一方法を示す。宣言的プログラムによって表現された制約論理プログラムにおいては、制約解消および制約伝搬の仕組みがユニフィケーションの中にあるためプログラム変換の適用を容易にしている。このことは、本プログラム変換が宣言的プログラムによって表すことのできる他の言語や表現にも適用可能性が高いことを示唆している。

2 プログラム変換の概要

本論文のプログラム変換は、次のようなものである。

1. 一意unfold変換が可能な限り、繰り返し行う。
2. 一意unfold変換が不可能ならば、呼び出しによる特殊化変換 (図1) を行い、1に行く。
3. 1も2も不可能ならば、同型変換 (図2) を行い、停止する。

unfold変換 [7] [2] を行う際、ある節のbodyアトムに対するresolventが1つしか存在しなければ、その節は新たな唯一の節に置き換える。これを一意unfold変換と呼ぶ。

呼び出しによる特殊化変換は、あるbodyアトムから呼び出されるすべての節を複製してその呼び出しによって特殊化した節を生成する変換である。その際、bodyアトムと新たに生成される節のheadにおける述語をひとつの新述語で置き換える。このときの呼び出しとそれに対応する新述語のペアをCPという集合として保持する (new pattern)。すでに呼び出しが登録されているbodyアトムの場合新述語を生成せずCPを参照して述語名を得る (old pattern)。

同型変換は不用な定数や変数などを削除する。その際、プログラムに現れるすべてのアトムに対する最も特殊な一般化パターン (msg) を述語ごとに求

Spec:

1. if $FR = \emptyset$ then stop.
2. target $(C,T) \in FR$ を1つ選ぶ。ただし、$C = A\text{:-}B \in P$, $T \in B$ とする。
3. (old pattern) if T が, ある $(E,pred) \in CP$ に対して Eのインスタンスである then
   - $T = E\delta$ を満たす δ を求める.
   - 節 C' を $A\text{:-}B'$ とする。ここで, B' は Cの body B 中の T の述語名を $pred$ に変更したアトム列である.。
   - 節 Cに関する target を FR から取り除いて FR'とする。
   - $FR := FR' \cup \{(C',T') | T' \in B'\}$
     $P = P - \{C\} \cup \{C'\}$.
4. (new pattern) else if T がすべての $(E,pred) \in CP$ に関して Eと排反である then
   - 新述語 new_pred を生成する。
   - 節 C' を $A\text{:-}B'$ とする。ここで, B' は Cの body B 中の T の述語のみを new_pred に変更したアトム列である.
   - T と unify 可能な head を持つ節 C_j を
     $$H_j\text{:-}F_j \in P(j \in J)$$
     とする。
   - T と H_j の mgu (θ_j, σ_j) を求める。
   - 節 C_j に対して, 次のように C'_j を求める。$C'_j = (H'_j\text{:-}F_j)\sigma_j$ ここで, H'_j は H_jの述語名を new_pred で置き換えたアトムである。
   - 節 Cに関するすべての target を FR から取り除いて FR'とする。
   - $FR := FR' \cup \{(C',T') | T' \in B'\}$
     $\cup \bigcup_{j \in J} \{(C'_j, T'_j) | T'_j \in F_j\sigma_j\}$
   - $CP := CP \cup \{T, new\_pred)\}$
     $P := P - \{C\} \cup \{C'\} \cup \{C'_j | j \in J\}$
5. else failure.

図1 呼び出しによる特殊化アルゴリズム

Program Transformation Using Unification of Constraint Objects

†Masaki Omatsu, Software Research Laboratory

††Kiyosi Akama, Eiichi Miyamoto, Hokkaido University

Itrans:

- アトムの集合 $\mathcal{A}$ に対して写像 $f : \mathcal{A} \rightarrow \mathcal{A}'$ を定め、P のすべての節 $C = A : -B_1, \ldots, B_n (n \geq 0)$ に対して $P' = \{C' | C' = f(A) : -f(B_1), \ldots, f(B_n)(n \geq 0)\}$ とする。
- 写像 f を以下のように定める。
  Program P に出現するすべてのアトムの集合 ATOM(P) を述語名によって排反に分割して、それらを $Atom(p_j)(j \in J)$ とする。ここで、p_j は述語名である。
  すべての $j \in J$ に対して $Atom(p_j)$ の最も特殊な一般化パターンを求めてそれらを $msg(p_j)$ とする。
  写像 $f : ATOM(P) \rightarrow ATOM(P')$ を次のように定める。
  ある $A \in ATOM(P)$ の述語名を p、$msg(p)$ と A との mgu を (θ, σ) とする。もし、$msg(p)$ が trivial でないなら、$f(A) = atom(p, vargs(msg(p)))\theta$。そうでないなら $f(A) = A\sigma$ とする。
  ここで、$vargs(pat)$ は pat における相異なる変数からなる引数を表し、$atom(p, args)$ は述語名を p、引数を $args$ とするアトムを表す。

図 2　同型変換のアルゴリズム

め、その情報を用いてどの項が不用であるかを決定する。

3　制約オブジェクトの単一化とプログラム変換

制約論理プログラム (CLP) [6] における確定節 $H : -C | B_1, B_2, \ldots, B_n . (n \geq 0)$ を本論文では次のような宣言的プログラムの確定節として表す [1]。

$$< H, [C] > : - < B_1, [C] >, \ldots, < B_n, [C] > . \quad (n \geq 0)$$

ここでは、H、...、B_n 等を単にアトム、$[C]$ を制約オブジェクト、$< H, [C] >$、...、$< B_n, [C] >$ は制約つきアトムと呼ぶ。また、このような節を制約節と呼ぶ。2 つの制約つきアトム $< A, [C_1] >$、$< D, [C_2] >$ のユニフィケーションは次のように行われる。

- アトム A、D は通常に mgu (α, δ) を求める。
- 制約オブジェクト $[C_1]\alpha$、$[C_2]\delta$ の和集合 $[C_u]$ を制約解消系に渡し、その結果 $[C_s]$ を得る。

ユニフィケーションの結果、$[C_s]$ は $[C_1]$、$[C_2]$ に置き変わるだけでなく、導出の際には瞬時に伝搬する。

制約節集合によって表現された制約論理プログラムに対する本プログラム変換の適用は、制約解消および制約伝搬の仕組みがユニフィケーション枠組みの中にあるため、unfold 変換はそのまま適用可能である。新たに定義すべきなのはインスタンス、msg における制約オブジェクトに対する扱いである。KMP ストリングマッチャーの例 [5] の実行のためには、インスタンス、msg において制約オブジェクトを定数と同様に扱うことで意図した結果が得られている。

4　おわりに

宣言的プログラムに対するプログラム変換の一方法を提案し、制約論理プログラムのプログラム変換において適用可能であることを示した。そこでは制約オブジェクトのユニフィケーションこそが重要な役割を担っている。

オブジェクト空間を適切に構成することによって、制約論理プログラムのみではなく、項書換えシステム、タイプつきユニフィケーション文法をふくむ、数多くのプログラミング言語や知識表現系の基礎を宣言的プログラムと見なすことができる。扱う対象が拡大してもユニフィケーションを拡張するだけで理論的枠組は変化しないため、本論文の方法はそれらの広範囲の言語に適用可能であると考えられる。

本論文のプログラム変換方法は、宣言的プログラムの理論に基礎づけられたプログラミング言語 UL/αで実現されている。

参考文献

[1] Akama,K.:A New Theoretical Foundation of Constraint Logic·Programs,*Hokkaido University Information Engineering Technical Report*, HIER-LI-9220(1992).

[2] Akama,K.:Unfolding of Generalized Logic Programs,*Preprints Work. Gr. for Artifficial Intelligence,IPSJ* 83-3-AI(1992),pp.43-52.

[3] Akama,K.:Homomorphism Theorem of Generalized Logic Programs, Hokkaido University Information Engineering Technical Report, HIER-LI-9213 (1992).

[4] 大松 正樹:UL/αのプログラム変換, 日本ソフトウェア科学会第 9 回大会論文集,1992.

[5] Fujita, H.:An Algolithm for Partial Evaluation with Constra ins, ICOT TM-0367(1987).

[6] Jaffer,J. and Lassez,J.L.:Constraint Logic Programing,Technical Report, Department of Computer Science, Monash University, June 1986. について,

[7] Tamaki,H. and Sato,T.:Unfolding / Folding Transformation of Logic Programs,Proc. of 2nd ILPC(1984),pp127-138.