知能と複雑系 113－3
(1998. 7. 23)

一階論理制約による宣言的記述の拡張

吉田 忠行　　赤間 清　　宮本 衛市
北海道大学 大学院 システム情報工学専攻
札幌市北区北 13 条西 8 丁目. Tel. 011-706-6814
{chuko,akama,miya}@complex.eng.hokudai.ac.jp

論理プログラミングでは, はじめは確定節で問題を表現していたが, 否定の表現を取りいれ, さらには一階述語論理の表現も論理式で扱うことができるようになった. このような問題の記述における一階述語論理の表現は, より柔軟でかつ高度な問題解決に貢献できる. しかし, 論理プログラミングは計算における問題を抱えている. 本論文ではこの問題を解決するために, 等価変換パラダイムにおいて, 一階述語論理の表現を扱うための枠組みを構築し, 宣言的記述のクラスを拡大する.

等価変換　　一階論理制約　　制約システム　　宣言的記述

Extension of a Class of Declarative Descriptions Using First-order Logical Constraints

Tadayuki Yoshida　　Kiyoshi Akama　　Eiichi Miyamoto
Division of System and Information Engineering, Hokkaido University
North 13 West 8 Kita-ku Sapporo. Tel. 011-706-6814
{chuko,akama,miya}@complex.eng.hokudai.ac.jp

To attain efficient computation, the class of definite clauses, which is a small subclass of first-order predicate logic, has been adopted. Then the concept of logic program is extended to include negative atoms and arbitrary first-order formulas in the body of definite clauses. Such extension increases the expressive power of logic programs greatly. Computation of first-order expressions has, however, some limitatoins in flexiblity, efficiency, and correctness. In this paper, we adopt the equivalent transformation paradigm to overcome such difficulties. We extend the existing class of declarative descriptions, and construct a new framework for representation and computation of first-order expressions.

equivalent transformation, first-order logical constriant, constraint system, declarative description

1 まえがき

論理パラダイムでは, 問題を論理式によって表現し, 推論により計算をする [4]. 論理プログラミングは確定節の範囲に表現を制限し, 高速な処理を達成したことにより, 知識処理の分野に大きく貢献した. 次に, この成功を拡大するために, 否定をボディにもつ表現の導入が研究されている. そして, ボディに一階述語論理式も取り入れることができるようになり, 表現力が飛躍的に拡大した. 例えば,「X は Y の部分集合である」という関係を表す述語 *subset* を確定節を用いて表現すると次のようになる.

$$subset(X,Y) \leftarrow \forall E\,(member(E,X) \Rightarrow member(E,Y)). \cdots (1)$$

論理パラダイムで一階述語論理式に対する枠組を構築するのは自然な流れであるように思えるかもしれない. しかし論理パラダイムは, 正当で高速な計算に関して問題が残っている.

本論文では, 論理パラダイムが抱える問題を解決するために, 等価変換パラダイムを採用する. 等価変換パラダイムでは, 計算は「宣言的記述の意味の保存」のみで規定するといった, 単一で, しかも基本的な原理を採用している. 計算では, この規定を満たす等価変換ルールであれば任意のルールを任意の順序で適用することができる. このような包括的な計算の構造は, 正当で高速な計算を達成するための理論的基礎を備えている.

等価変換パラダイムで, 一階述語論理の表現が適切に扱えるということを示すには「表現」と「計算」の両面から述べる必要があるが, 本論文では, どのような「表現」を用いるかという問題を中心に議論する.

2 論理パラダイムでの表現と計算

この章では, 文献 [4] に基づき, 論理パラダイムでの一階述語論理式の表現と計算について概略を述べ, 問題点を指摘する.

2.1 表現の変換と推論

論理パラダイムでは計算を推論によって行う. 一階述語論理を用いた論理式を計算するには, 推論の前段階として, もとの論理式をそれと論理的に同値な論理積と否定だけを用いた表現に変換する. そして変換されたあとの論理式に対して推論をし, 推論の際の代入により解を得る.

先のプログラム (1) に論理的同値性を保つ変換を施すと次のようなプログラムが得られる.

$$subset(X,Y) \leftarrow \neg q(X,Y).$$
$$q(X,Y) \leftarrow member(Z,X), \neg member(Z,Y). \cdots (2)$$

このプログラム (2) は,「Z が X に含まれていて, Z が Y に含まれていないなら $q(X,Y)$ が成り立ち, $q(X,Y)$ でないなら $subset(X,Y)$ が成り立つ」ということを表現している.

プログラム (2) は, 例えば,「[1,2] は [1,2,3] の部分集合であるか」というような, 集合が確定した問題については, 正しく yes,no を判定することができる. しかし,「 [Z] が [1,2,3] の部分集合となるとき, Z を求めよ」といった問題が与えられたときには実行は次のようになる.

$$\leftarrow subset([Z],[1,2,3]).$$
$$subset(X,Y) \leftarrow \neg q(X,Y).\ を$$
$$\{X/[Z], Y/[1,2,3]\}\ として$$
$$\leftarrow \neg q([Z],[1,2,3]).$$

このあと SLDNF-レゾリューションは $\neg q([Z],[1,2,3])$ を選択することができない. なぜなら, SLDNF-レゾリューションには否定を含むリテラルが中に変数を含んでいる場合, そのリテラルを含む節を選択できないという制限があるからである. つまり, SLDNF-レゾリューションでは, この Z を求める問題に対しては, 解を得ることができない[1].

2.2 論理パラダイムの問題点

論理パラダイムでは, 一階述語論理式を含む論理プログラムと問い合わせに対する計算は,「論理的な同値性」を保つ表現の変換の段階と, SLDNF$_r$レゾリューションに基づく推論の段階という, 2つの段階に分かれている. この計算が分断されているという構造は複雑であり, さらに計算を推論に限定するので自由度が低いといえる. したがって, 正当で高速な計算を達成するには困難な理論体系になっている.

3 等価変換パラダイムによるアプローチ

この章では, 等価変換パラダイムについて概略を説明し, 論理パラダイムとの理論的な違いを明確にする.

3.1 等価変換パラダイム

等価変換パラダイムでは,「計算は等価変換である」と考える. すなわち, 意味が与えられた「表現」に対し,

[1] 論理パラダイムの言語 Prolog では, このような場合でも無理矢理計算を進めるが正しい解が得られる保証がない.

その意味を保存する変換が「計算」である.

問題を解の集合の宣言記述で表現し, その宣言的記述を等価変換し, 十分簡単化された宣言的記述から解を得る, というのが等価変換に基づく問題解決の枠組である.

等価変換パラダイムでは, データ構造に依存しない形で共通に宣言的意味を与えるために, 必要な構造を公理で規定している. 確定節の中のヘッドアトムやボディアトムは特殊化システムで特徴づけ, 確定節に対する制約はアトムとは独立に真偽の決まるものであるから制約システムという特殊化システムから派生した構造で扱う.

扱う問題を表現する 宣言的記述 は特殊化システム, 制約システムの上で構成される. その宣言的記述を 等価変換ルール を用いて変換する.

3.2 論理表現を扱う上での理論的な相違点

論理パラダイムでは, 問題の構造を項領域を基本とした論理式で表現する. このように「表現」が先にあり, そのための計算構造を後から与えるという理論的構造をしているため, 表現方法が拡大すると, それに伴う新しい計算構造が必要となる場合がある. 一階述語論理式の導入では, それまでの否定を含む式を扱うためのSLDNF-レゾリューションのほかに, 「論理的同値性」による変換の段階が必要となっている [4]. 計算の順序も変換のあとレゾリューションというように, 高速な計算を達成するには容易ではない構造をしている. さらに, 理論体系の基礎にはモデルや論理的帰結 ($\models$) などの概念が不可欠であり, 計算の正当性はレゾリューション全体に対して与えられているため, 表現能力が拡大するとそれに対する拡張されたSLDNF-レゾリューションの正当性は一から議論し直さねばならない. このように, 計算の基礎を与える理論的なコストが非常に高いといえる.

いっぽう, 等価変換の枠組では, 「計算とは宣言的記述の等価変換である」と考え, モデルや論理的帰結ではなく, 宣言的意味が計算を規定している. これは計算方法に対する「厳しい」制限ではなく, 「非常に緩い」制限である. このように計算の構造をまず与え, 意味を与え得るどんな表現でも理論の対象としている. 計算は等価変換ルールの適用によって行われ, それぞれのルールを独立に議論することができる. したがって, 新しい表現機構が導入されると, それに対する等価変換ルールを開発すればよい. つまり個々のルールが「宣言的意味を保存」していることを理論的に証明すればよい.

等価変換パラダイムは, 論理パラダイムと比較して, データの表現や計算に関して自由度が高く, 豊かな表現力, 高速な計算などの利点を享受することができる. また, 理論を拡張するためのコストが論理パラダイムに比べて低いといえる.

3.3 等価変換パラダイムにおける表現の枠組

等価変換の理論的基礎の概略を述べる. 数学的に厳密な定義は付録A章を参照されたい.

3.3.1 特殊化システム

多様な問題を解決するとき, これまでは, 問題ごとに個別にデータ構造を提案してきた. 等価変換の枠組では, 対象となるデータを特殊化システムという統一的な構造で特徴づける. 特殊化システムとは, 意味を与え得るデータ構造の公理を満たす対象とそれへの操作を定義した数学的な構造である. 論理プログラムや制約論理プログラム, 単一化文法などを含む多くの知識表現系の基礎には, ある特殊化システムの構造が存在し, 扱う問題の表現と計算の能力は特殊化システムの構造に大きく依存している.

3.3.2 制約システム

問題を表現する宣言的記述の中に制約を取りいれることができる. 制約は宣言的記述の内容とは独立にその真偽を定めることができる. 計算を高速に行うには制約の表現が欠かせないため, 制約を柔軟に表現できることが高速な計算に寄与すると考えられる.

制約システムとは, 特殊化システムとは独立に意味を与え得る, 制約を表現するための数学的な構造である. これは, 特殊化システムをベースとし, 基礎オブジェクトに真偽を与えるための集合を付加した構造をしている.

3.3.3 制約システムの適合性

特殊化システムと制約システムは独立に構成するのが普通である. しかし問題を表現する宣言的記述にはこれらを合わせて用いるため, 特殊化システムと制約システムが共存するには, 何かしらの条件が必要となる. その条件とは,

(1) 特殊化システムのオブジェクト (アトム) と制約システムのオブジェクト (制約) の集合が排反であること

(2) 特殊化の集合が一致していること

である. この条件を満たす特殊化システムと制約システムは「適合する」といい, 適合する両者を用いて宣言的記述を構成することができる.

3.3.4 宣言的記述と宣言的意味

アトムをヘッドに持ち, アトムと制約をボディに持つ確定節の集合を宣言的記述と呼ぶ. 宣言的意味は, $\emptyset$ から

無限回推論して得られる基礎単位節全体の集合である．ただし，この推論という言葉は，論理パラダイムから見たときの言い方にすぎず，等価変換パラダイムでは，推論という考え方は導入せず，宣言的意味を数学的に定義し，それがすべてであるとする．

3.4 本論文の方針

等価変換の枠組では，問題の解を宣言的記述で表現し，それを等価変換することで，十分簡単化された問題表現から解を得る．本論文では宣言的記述の中で一階述語論理の表現をどのように扱うかを議論する．より具体的には，制約に着目し，一階述語論理に相当する論理表現を制約に取りいれる．

このために，まず基本となる制約の集合 (基本制約システム) を定め，その元を論理表現を用いて組み立てた式のクラスが制約として機能することを示す．これにより宣言的記述の中に論理表現を用いた制約 (一階論理制約と呼ぶ) を導入することができる．

以降の章では，次のように議論を進める．4章では論理記号や束縛変数を用いた式のクラスを定義する．5章では項領域を基礎とする表現機構を構築する．6章では4章で定義した式のクラスの部分クラスを用いて一階論理制約システムを構築する．

4 拡大された項に基づく式のクラス

この章では，変数，定数，関数を持つ通常の項領域の式のクラスに論理記号と束縛変数を導入した拡大したクラスを定義する．

4.1 論理記号

本論文では論理記号として，表4.1に示されたものを用いる．全称記号と存在記号をあわせて 量化記号 と呼ぶ．

論理操作	論理記号	一般的な読み方
連言	$\wedge$	かつ
選言	$\vee$	または
含意	$\Rightarrow$	ならば
否定	$\neg$	でない
全称記号	$\forall\alpha$	すべての α について
存在記号	$\exists\alpha$	ある α が存在して

表 1: 本論文で使用する論理記号

4.2 拡大された項

K, V, W, F を互いに排反な集合とする．K, V, W, F の元をそれぞれ，定数，自由変数，束縛変数，関数と呼ぶ．以降，例を示すときには，K の定数は $a, b, c\cdots$，自由変数は $x, y, z, \cdots$ 束縛変数は$\alpha, \beta, \gamma, \cdots$，$F$ の関数は $f, g, h, \cdots$ などを用いる．F のそれぞれの元にはアリティと呼ばれる自然数が割り当てられていると仮定する．

K, V, W, F 上の項とは，K の定数，V の変数，W の変数，あるいは，K, V, W, F 上の複合項である．K, V, W, F 上の複合項とは，アリティnの関数$f \in F$ と，K, V, W, F 上の項 $t_1, \cdots, t_n$によって作られた $f(t_1, \cdots, t_n)$ の形の式である．K, V, W, F 上の項全体の集合を$T(K, V, W, F)$ と表す．自由変数，束縛変数のどちらも含んでいない項を基礎項と呼ぶ．基礎項全体の集合は $T(K, \emptyset, \emptyset, F)$ と書くことができる．

$\mathcal{E}_T$を $V \times T(K, V, \emptyset, F)$ とする．$\mathcal{E}_T$の元を束縛と呼ぶ．写像 $\nu_T : \mathcal{E}_T \to map(T(K, V, W, F))$ は次のように定義される．ただし，$map(X)$ は X 上の全域写像全体の集合を表す．$a \in T(K, V, W, F)$ とする．

(1) a が複合項でないとき
$e = (v, t) \in V \times T(K, V, W, F)$ の とき
$\nu_T(e)(a) = t \qquad a = v$ のとき
$\nu_T(e)(a) = a \qquad$ その他の場合
(2) a が複合項 $f(t_1, \cdots, t_n)$ であるとき
$\nu_T(e)(a) = f(\nu_T(e)(t_1), \cdots, \nu_T(e)(t_n))$

である．

$\mathcal{S}_T$ を $(\mathcal{E}_T)^*$，すなわち $\mathcal{E}_T$ 上のすべての有限列全体の集合とする．$\mathcal{S}_T$ の元を特殊化と呼ぶ．ν_T に基づいて，写像 $\mu_T : \mathcal{S}_T \to map(T(K, V, W, F))$ を次のように定義する．

$\mu_T(\lambda)(a) = (a)$
$\mu_T(e \cdot s)(a) = \mu_T(s)(\nu_T(e)(a)) \qquad e \in \mathcal{E}_T,\ s \in \mathcal{S}_T$
ただし，空列をλ，列の連結演算を$\cdot$とする．

4.3 式のクラス L と$\overline{L}$

K, V, F, W と述語の集合 R，論理記号 $(\neg, \wedge, \vee, \Rightarrow, \forall, \exists)$ を使っててられる式のクラス $L(K, V, F, R)$ および $\overline{L}(K, V, W, F, R)$ を以下のように定義する．

定義 1 【クラス $L(K, V, F, R)$】
述語 $p \in R$ と項 $t_i (i = 1, \cdots, n) \in T(K, V, \emptyset, F)$ によって作られる $p(t_1, \cdots, t_n)$ の形の式の集合を $L(K, V, F, R)$ と略記する． □

定義 2 【クラス$\overline{L}(K, V, W, F, R)$】
$\overline{L}(K, V, W, F, R)$ を以下のように帰納的に定義する．定義の中では$\overline{L}(K, V, W, F, R)$ を単に$\overline{L}$と表す．

(1) 述語 $p \in R$ と項 $t_i (i = 1, \cdots, n) \in T(K, V, W, F)$ によって作られる $p(t_1, \cdots, t_n)$ の形の式は$\overline{L}$ の元である.

(2) E が $\overline{L}$ の元であるとき, $(\neg E)$ も$\overline{L}$ の元である.

(3) E_1 と E_2 が $\overline{L}$ の元であるとき, $(E_1 \vee E_2)$, $(E_1 \wedge E_2)$, $(E_1 \Rightarrow E_2)$ はいずれも $\overline{L}$ の元である.

(4) E' が $\overline{L}$ の元で, α が 束縛変数であり, E'が内部に $\forall\alpha$あるいは $\exists\alpha$を含まないとする. このとき $(\forall\alpha\, E')$ や $(\exists\alpha\, E')$ は$\overline{L}$の元である. □

条件 (1) より, $\overline{L}(K, V, W, F, R)$ は $L(K, V, F, R)$ を包含している. また, 条件 (4) で, αは式E'の中に引数として含まれていてもいなくてもよい. ただし, 「E'が内部に $\forall\alpha$あるいは $\exists\alpha$を含まないとする.」という条件により, E' がすでに

$(p_1(x) \Rightarrow \forall\alpha\, p_2(\alpha))$

のように内部に $\forall\alpha$または $\exists\alpha$を含んでいるとき,

$(\forall\alpha\, E') = (\forall\alpha\, (p_1(x) \Rightarrow \forall\alpha\, p_2(\alpha)))$

のように, 同一の束縛変数αに対して, $\forall\alpha$あるいは $\exists\alpha$で二重に束縛してはならない.

例 1 $\overline{L}(K, V, W, F, R)$ の元の例として次のような式がある.

$p_1(x, \alpha, a)$, $(\forall\alpha\, p_1(f(x), \alpha, \beta))$, $(\exists\beta\, (\neg p_1(x, y)))$, $(\forall\alpha\, (p_1(\alpha, x) \Rightarrow \exists\beta\, (y, \beta)))$

次に変数の出現という概念を定義する.

定義 3 【束縛されている出現】
E' を$\overline{L}(K, V, W, F, R)$ の元としたとき, $(\forall\alpha\, E')$ あるいは $(\exists\alpha\, E')$ の E'の中でのαの出現は 束縛されている という. □

$\overline{L}$の定義 (4) より, $\overline{L}$の元 E の引数に束縛変数があるときは, その束縛変数の出現が束縛されているための量化記号 $(\forall, \exists)$ はただ 1 つ存在する.

例 2 $\overline{L}(K, V, W, F, R)$ の元である式 $(\forall\alpha\, p(x, \alpha, \beta))$ において, $p(x, \alpha, \beta)$ のαの出現は束縛されている. これに対し, βの出現は束縛されていない.

4.4 $\overline{L}$の式への代入

4.2節で定義した写像μ_Tを用いて, 特殊化の集合 $\mathcal{S}_T$ を $\overline{L}$の元に作用させたときの写像 $\mu_{\overline{L}} : \mathcal{S}_T \to map(\overline{L})$ を次のように定義する.

定義 4 【写像 $\mu_{\overline{L}}$】
E, E_1, E_2 を $\overline{L}$ の元, α を束縛変数, s を特殊化とする.

(1) E に論理記号が含まれていないとき
$t_1, \cdots, t_n \in T(K, V, W, F), E = p(t_1, \cdots, t_n)$ とする.
$\mu_{\overline{L}}(s)(E) = \mu_{\overline{L}}(s)(p(t_1, \cdots, t_n))$
$= p(\mu_T(s)(t_1), \cdots, \mu_T(s)(t_n))$

(2) $\mu_{\overline{L}}(s)(\neg E) = \neg\mu_{\overline{L}}(s)(E)$

(3) $*$ が $\wedge, \vee, \Rightarrow$ のいずれかのとき
$\mu_{\overline{L}}(s)(E_1 * E_2) = \mu_{\overline{L}}(s)(E_1) * \mu_{\overline{L}}(s)(E_2)$

(4) $\mu_{\overline{L}}(s)(\forall\alpha\, E) = \forall\alpha\, \mu_{\overline{L}}(s)(E)$
$\mu_{\overline{L}}(s)(\exists\alpha\, E) = \exists\alpha\, \mu_{\overline{L}}(s)(E)$ □

5 項領域での特殊化システムと制約システム

5.1 項領域のアトムのための特殊化システム

R_Γ を任意の集合とする. $\mathcal{A}_\Gamma$,$\mathcal{G}_\Gamma$,$\mathcal{S}_\Gamma$を次のように定義する.

$\mathcal{A}_\Gamma = L(K, V, F, R_\Gamma)$
$\mathcal{G}_\Gamma = L(K, \emptyset, F, R_\Gamma)$
$\mathcal{S}_\Gamma = \mathcal{S}_T$

$\mu_{\overline{L}}$に関して $\mathcal{A}_\Gamma$が閉じていることは $\mathcal{S}_T$の定義から明らかなので, 写像 $\mu_\Gamma : \mathcal{S}_\Gamma \to map(\mathcal{A}_\Gamma)$ を

任意の $s \in \mathcal{S}_\Gamma, A \in \mathcal{A}_\Gamma$に対して,
$\mu_\Gamma(s)(A) \stackrel{\text{def}}{=} \mu_{\overline{L}}(s)(A)$

で定義する.

このとき, $\Gamma = \langle \mathcal{A}_\Gamma, \mathcal{G}_\Gamma, \mathcal{S}_\Gamma, \mu_\Gamma \rangle$ は特殊化システムである.

5.2 項領域の制約システム

R_Π を任意の集合とする. $\mathcal{A}_\Pi$,$\mathcal{G}_\Pi$,$\mathcal{S}_\Pi$を次のように定義する.

$\mathcal{A}_\Pi = L(K, V, F, R_\Pi)$
$\mathcal{G}_\Pi = L(K, \emptyset, F, R_\Pi)$
$\mathcal{S}_\Pi = \mathcal{S}_T$

$\mu_{\overline{L}}$に関して $\mathcal{A}_\Pi$が閉じていることは $\mathcal{S}_T$の定義から明らかなので, 写像 $\mu_\Pi : \mathcal{S}_\Pi \to map(\mathcal{A}_\Pi)$ を以下のように定義する.

任意の $s \in \mathcal{S}_\Pi, A \in \mathcal{A}_\Pi$に対して,
$\mu_\Pi(s)(A) \stackrel{\text{def}}{=} \mu_{\overline{L}}(s)(A)$

$\mathcal{G}_\Pi$ の元には, それぞれ真と偽のどちらかの値を割り当てる. このために, $\mathcal{D}$ を $\mathcal{G}_\Pi$ の部分集合として, $\mathcal{G}_\Pi$ の元 g は, $g \in \mathcal{D}$ のとき真, それ以外のとき偽であるとする.

このとき, $\Pi = \langle \mathcal{A}_\Pi, \mathcal{G}_\Pi, \mathcal{D}, \mathcal{S}_\Pi, \mu_\Pi \rangle$ は制約システムである.

6 項領域の一階論理制約システム

6.1 項領域の基本制約システム

5.2節で定義した項領域での制約システム Π を $\Pi_B = \langle \mathcal{A}_B, \mathcal{G}_B, \mathcal{D}_B, \mathcal{S}_B, \mu_B \rangle$ とし，本章では，Π_B をもとに，論理記号と W の変数を用いて拡大した一階論理制約システム $\Pi_L = \langle \mathcal{A}_L, \mathcal{G}_L, \mathcal{D}_L, \mathcal{S}_L, \mu_L \rangle$ を構築する．

Π_L を構築するもとの制約システムを一階論理制約システムに対し基本制約システムと呼ぶ．また，$\mathcal{A}_B$ の元を 基本制約 と呼ぶ．

6.2 クラス $\mathcal{A}_L$

$\overline{L}(K, V, W, F, R_\Pi)$ の部分集合として，式のクラス $\mathcal{A}_L$ を定める．以降では，K, V, W, F, R_Π は固定して議論するので，$\overline{L}(K, V, W, F, R_\Pi)$ を単に $\overline{L}$ と表記することがある．

定義 5 【$\mathcal{A}_L$の元の条件】
$\overline{L}(K, V, W, F, R_\Pi)$ の元のうち，次の条件を満たすものを $\mathcal{A}_L$の元とする．

> $\overline{L}$の任意の元に含まれる束縛変数の，量化記号の直後ではないすべての出現は束縛されている

上の定義は $\mathcal{A}_L$の元が束縛されていない出現を引数に含んではいけないことを要請している．したがって，$p(x, \alpha, y, z)$ のような式は $\mathcal{A}_L$ の元ではない．基本制約は束縛変数を含んでいないので $\mathcal{A}_L$の元である．

例 3 $p_1(x), p_2(x, y), p_3(x, y, z)$ が基本制約であるとき，
$(\forall\alpha(p_1(\alpha) \Rightarrow (\neg p_2(\alpha, f(\alpha)))))$
$(\forall\alpha(\exists\beta(p_2(f(\alpha), \beta) \Rightarrow (\exists\gamma(p_3(\beta\gamma, w))))))$
などは $\mathcal{A}_L$の元である．

誤解を招くおそれのない限り，冗長なカッコは適当に省略することにする．例えば，先の例3の式はそれぞれ，
$\forall\alpha\, (p_1(\alpha) \Rightarrow \neg p_2(\alpha, f(\alpha)))$
$\forall\alpha\exists\beta(p_2(f(\alpha), \beta) \Rightarrow \exists\gamma\, p_3(\beta, \gamma, w))$
と書くことができる．

6.3 閉式の集合とその真偽

E の中にある自由変数の集合を $\mathit{freeV}(E)$ と表す．また，E が自由変数を 1 つも持たないとき，すなわち，$\mathit{freeV}(E) = \emptyset$ のとき，「E は 閉じている」といい，E を 閉式 という．

例 4 $\mathcal{A}_L$の元 E が $\exists\alpha\, p_1(x, \alpha, y) \wedge \exists\beta\, p_2(x, \beta)$ であるとする．このとき，
$\mathit{freeV}(E) = \{x, y\}$ である．

定義 6 【閉式の集合 $\mathcal{G}_L$】
閉式の集合 $\mathcal{G}_L$ を次のように定義する．
$\mathcal{G}_L = \{ E \mid E \in \mathcal{A}_L, \mathit{freeV}(E) = \emptyset \}$ □

この定義から，$\mathcal{G}_L$は明らかに $\mathcal{A}_L$の部分集合である．

定義 7 【集合 $\mathcal{D}_L$】
$\mathcal{G}_L$の部分集合である集合 $\mathcal{D}_L$を次のように定義する．E' を $\overline{L}$ の元，E, E_1, E_2 を $\mathcal{G}_L$の元，α を W の変数とする．

(1) E が基本制約のとき，
$E \in \mathcal{D}_L \iff E \in \mathcal{D}_B$
(2) E が $(\neg E_1)$ の形のとき，
$(\neg E_1) \in \mathcal{D}_L \iff E_1 \notin \mathcal{D}_L$
(3) E が $(E_1 \wedge E_2)$ の形のとき，
$(E_1 \wedge E_2) \in \mathcal{D}_L \iff$
$E_1 \in \mathcal{D}_L$ かつ $E_2 \in \mathcal{D}_L$
(4) E が $(E_1 \vee E_2)$ の形のとき，
$(E_1 \vee E_2) \in \mathcal{D}_L \iff$
$E_1 \in \mathcal{D}_L$ または $E_2 \in \mathcal{D}_L$
(5) E が $(E_1 \Rightarrow E_2)$ の形のとき，
$(E_1 \Rightarrow E_2) \in \mathcal{D}_L \iff$
$(\neg E_1 \vee E_2) \in \mathcal{D}_L$
(6) E が $(\forall\alpha\, E')$ の形のとき，
$(\forall\alpha\, E') \in \mathcal{D}_L \iff$
$\forall g \in \mathcal{G}_T : E'[\alpha/g] \in \mathcal{D}_L$
(7) E が $(\exists\alpha\, E')$ の形のとき，
$(\exists\alpha\, E') \in \mathcal{D}_L \iff$
$\exists g \in \mathcal{G}_T : E'[\alpha/g] \in \mathcal{D}_L$

ただし，(6), (7) で $E'[\alpha/g]$ は E'の中のすべてのαを g に置き換えた式とする． □

閉式 E は $E \in \mathcal{D}_L$のとき真，$E \notin \mathcal{D}_L$のとき偽であるという．閉式は定義7により必ず真偽が決まる[2]．

6.4 写像 μ_L

4.4節で定義した写像$\mu_{\overline{L}}$に関して，次の命題が成立する．

命題 1 $\mu_{\overline{L}}$に関して $\mathcal{A}_L$は閉じている． □

$\mathcal{S}_L$を $\mathcal{S}_T$と同一とする．$\mathcal{A}_L$は$\mu_{\overline{L}}$に関して閉じているので，$\mu_{\overline{L}}$を用いて$\mu_L : \mathcal{S}_L \to map(\mathcal{A}_L)$ を定義する．

定義 8 【写像 μ_L】
任意の $\mathcal{A}_L$の元 E，特殊化 s に対して，
$\mu_L(s)(E) \stackrel{\text{def}}{=} \mu_{\overline{L}}(s)(E)$ □

---
[2] 任意の閉式はこの (1) から (7) の定義により論理記号や W の変数を除去していくことで，ついには基本制約になるので well-defined である．

この定義により, $\mathcal{A}_L$の元に対して, 通常の意味での代入を施すことができる.

6.5 一階論理制約システム

ここまでで必要な概念が準備できたので, 一階論理制約システムを定義する.

命題 2 $\Pi_L = \langle \mathcal{A}_L, \mathcal{G}_L, \mathcal{D}_L, \mathcal{S}_L, \mu_L \rangle$ は制約システムである. □

この制約システムΠ_Lを (Π_Bに基づく) 一階論理制約システムと呼ぶ. $\mathcal{A}_L$ の元を 一階論理制約 と呼ぶ.

また, 一階論理制約システムの適合性について次の命題が成り立つ.

命題 3 制約システムΠ_Bが特殊化システムΓに適合するとき, Π_Bに基づく一階論理制約システム Π_L もΓに適合する. □

特殊化システムΓと, それに適合する制約システムΠ_Bに基づく一階論理制約システム Π_Lを用いて, 一階論理制約を含む宣言的記述のクラスを導入することができる.

7 むすび

本論文では論理パラダイムに代わり新しく等価変換パラダイムを採用し, 一階述語論理の表現に対する適切な枠組を構築するための「表現」に関する部分を議論した.

等価変換パラダイムでは, 問題の記述を解の集合の宣言的記述とし, それを等価変換するという方法を採用している. 基本となる宣言的記述は確定節であるが, その確定節のボディに論理表現を用いた制約 (一階論理制約) を記述できるようにし, 「一階論理制約つき宣言的記述」のクラスを導入した. これにより従来の「(論理表現のない) 制約つき宣言的記述」のクラスでは表現が難しかった問題に対しても素直な記述が可能になる.

等価変換パラダイムで, 一階述語論理の表現をその体系に組み込む全体の枠組はすでに構築されている [6]. しかし, この枠組が真に適切であることを示すには, 一階論理制約に対する計算, すなわち等価変換ルールが与えられることを述べなければならない. この「計算」に関する議論はあらためて別の論文で与えることとするが, 等価変換パラダイムでは「宣言的意味の保存」のみが計算を規定しているので, この規定に基づく任意の一階論理制約が計算できる等価変換ルールを与えればよい. このように各等価変換ルールが独立に議論できるのも, 等価変換パラダイムの理論的な強みといえる.

また, 本論文では項領域を前提として議論を進めてきたが項領域に限らずある公理を満たすデータ構造ならば同様の議論で一階論理制約を導入することが可能である. これについても別の論文で議論を与えることにする.

参考文献

[1] 赤間清, 繁田良則, 宮本衛市: 論理プログラムの等価変換による問題解決の枠組, 人工知能学会誌, Vol.12, No.2, pp.90–99 (1997).

[2] 赤間清, 川口雄一, 宮本衛市: 項領域における包含制約の等価変換, 人工知能学会誌, Vol.13, No.2, pp.112-120 (1998).

[3] 赤間清, 岡田浩一, 宮本衛市：否定を表現する新しい枠組– 等価変換に基づくアプローチ –, 電子情報通信学会技術研究報告, SS 97-37, pp.1-8 (1997).

[4] Lloyd,J.W.: Foundations of Logic Programming, Second Edition,Springer-Verlag(1987).

[5] 小野寛晰： 情報科学における論理, 日本評論社 (1994).

[6] 吉田忠行, 赤間清, 宮本衛市：一階述語制約の導入による宣言的記述の拡大, 情報処理北海道シンポジウム '98, Info-Hokkaido 98, 42 札幌, pp.92-95 (1998).

A 制約つき宣言的記述のクラス

この章では, 論文中で詳しく述べなかった, 等価変換の理論的基礎の数学的定義を与える.

A.1 特殊化システム

定義 9 【特殊化システム】
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. また, $\mathcal{G}$ 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.

$a\theta = b$ 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. $a\theta = b$ のとき, a は θ で b に特殊化されるという. さらに b が基礎オブジェクトであるとき, b は a の基礎例であるという. $\theta \in \mathcal{S}$ 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. $a\theta$ や $B\theta$ を含む条件があるときには, その条件が, 「θ が a や B に適用可能である条件」も含んでいると約束する.

A.2 制約システム

定義 10 【制約システム】
制約システムとは，集合 $\mathcal{A},\mathcal{G},\mathcal{D},\mathcal{S}$ と写像 $\mu:\mathcal{S}\rightarrow partial\_map(\mathcal{A})$ からなる 5 項組 $\Pi=\langle\mathcal{A},\mathcal{G},\mathcal{D},\mathcal{S},\mu\rangle$ で，次の条件を満たすものである．

(1) $\langle\mathcal{A},\mathcal{G},\mathcal{S},\mu\rangle$ が特殊化システムの公理 (1) (2) (3) をそれぞれ満たす．

(2) $\mathcal{D}\subset\mathcal{G}$

$\mathcal{A}$ の元を 制約 と呼ぶ．また，$\mathcal{G}$ の元を 基礎制約，$\mathcal{S}$ の元を 特殊化，$\mathcal{D}$ の元を 制約領域 と呼ぶ． □

基礎制約 g は $g\in\mathcal{D}$ のとき，真であるという．また，特殊化の適用可能性については，特殊化システムと同様とする．
特殊化システムに対する制約システムの適合性を次のように定義する．

定義 11 【制約システムの適合性】
特殊化システム$\Gamma=\langle\mathcal{A}_\Gamma,\mathcal{G}_\Gamma,\mathcal{S}_\Gamma,\mu_\Gamma\rangle$，制約システム$\Pi=\langle\mathcal{A}_\Pi,\mathcal{G}_\Pi,\mathcal{D},\mathcal{S}_\Pi,\mu_\Pi\rangle$ とする．このとき，$\mathcal{A}_\Gamma\cap\mathcal{A}_\Pi=\emptyset$ かつ $\mathcal{S}_\Gamma=\mathcal{S}_\Pi$ ならば，Π は Γ に 適合する という． □

A.3 制約付き宣言的記述のクラス

定義 12 【Γ,Π上の宣言的記述のクラス】

(1) 特殊化システム $\Gamma=\langle\mathcal{A}_\Gamma,\mathcal{G}_\Gamma,\mathcal{S},\mu_\Gamma\rangle$ と，Γ に適合する制約システム $\Pi=\langle\mathcal{A}_\Pi,\mathcal{G}_\Pi,\mathcal{D},\mathcal{S},\mu_\Pi\rangle$ があるとき，$\mathcal{A}_\Gamma$ の元 H と，$\mathcal{A}_\Gamma$ の元または $\mathcal{A}_\Pi$ の元 (制約) $B_1,B_2,\cdots,B_n$ からなる式
$H\leftarrow B_1,B_2,\cdots,B_n$
は，Γ,Π 上の確定節である．

(2) Γ,Π上の確定節の集合は Γ,Π上の宣言的記述である． □

H をヘッド，$B_1,B_2,\cdots,B_n$ をボディと呼ぶ．
確定節 C のヘッドを $head(C)$，ボディ中のオブジェクトの集合を $obj(C)$，ボディ中の制約の集合を $con(C)$，$obj(C)\cup con(C)$ を $body(C)$ で表す．ヘッドまたはボディの構成要素として使われるオブジェクトをしばしば アトム と呼ぶ．
基礎オブジェクトと基礎制約だけからなる確定節を基礎節と呼ぶ．$Gclause(\Gamma,\Pi)$ をΓ,Π上の基礎節全体の集合とする．
確定節
$C=(H\leftarrow B_1,B_2,\cdots,B_n)$
と 特殊化 θ に対して，もし θ が H, B_1, B_2, $\cdots$, B_n のすべてに適用可能であるならば，θ は C に適用可能で，適用結果 $C\theta$ は
$C\theta=(H\theta\leftarrow B_1\theta,B_2\theta,\cdots,B_n\theta)$
であると定義する．$C\theta=C'$ のとき，C' は C の例であるという．基礎節 C' が C の例であるとき，C' は C の基礎例であるという．

A.4 制約付き宣言的記述の宣言的意味

Γ,Π上の宣言的記述の宣言的意味を定義する．
まず，Γ,Π 上の宣言的記述 P に対して，写像 $T_P:2^{\mathcal{G}_\Gamma}\rightarrow 2^{\mathcal{G}_\Gamma}$ を定義する．

定義 13 【写像T_P】
特殊化システム $\Gamma=\langle\mathcal{A}_\Gamma,\mathcal{G}_\Gamma,\mathcal{S},\mu_\Gamma\rangle$，$\Gamma$ に適合する制約システム $\Pi=\langle\mathcal{A}_\Pi,\mathcal{G}_\Pi,\mathcal{D},\mathcal{S},\mu_\Pi\rangle$ とする．P をΓ,Π上の宣言的記述とする．写像 $T_P:2^{\mathcal{G}_\Gamma}\rightarrow 2^{\mathcal{G}_\Gamma}$を，任意の $I\subset\mathcal{G}_\Gamma$ に対して

$$T_P(I)\overset{\mathrm{def}}{=}\{\,head(C\theta)\mid \begin{array}{l}C\in P,\\ \theta\in\mathcal{S},\\ C\theta\in Gclause(\Gamma,\Pi),\\ obj(C\theta)\subset I,\\ con(C\theta)\subset\mathcal{D}\,\}\end{array}$$

で定義する． □

上の定義の中の条件 $con(C\theta)\subset\mathcal{D}$ は，P の節 C から得られる基礎例 $C\theta$ のうち，$C\theta$ のボディの中の制約がすべて真であるような基礎例だけが宣言的意味を求める際に考慮されることを要請している．
次に，写像 T_P を用いて，制約付き宣言的記述の宣言的記述 P の宣言的意味を定義する．

定義 14 【宣言的記述の宣言的意味】
P を Γ,Π 上の宣言的記述とする．宣言的記述 P の宣言的意味 $\mathcal{M}(P)$ は，

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

で定義される． □

B 命題の証明

(1) 命題 2 の証明．μ_T は $V,T(K,V,\emptyset,F)$ 上の代入による写像であり，μ_L が μ_T をもとにしていること，$\mathcal{S}_L$ が $\mathcal{S}_T$ と同一であることから，$\mathcal{A}_L,\mathcal{S}_L,\mu_L$が特殊化システムの公理 (1)(2) を満たすのは明らかである．また，$\mathcal{G}_L,\mathcal{D}_L$ の定義 (定義 6,7) より，
$\mathcal{G}_L\subset\mathcal{D}_L\subset\mathcal{A}_L$
であるから，特殊化システムの公理 (3)，制約システムの公理 (2) を満たす．ゆえに．$\Pi_L=\langle\mathcal{A}_L,\mathcal{G}_L,\mathcal{D}_L,\mathcal{S}_L,\mu_L\rangle$ は制約システムである．

(2) 命題 3 の証明．Γの $\mathcal{A}_\Gamma$は式のクラス$\overline{L}(K,V,\emptyset,R_\Gamma)$ であり，Π_B の $\mathcal{A}_B$は式のクラス$\overline{L}(K,V,\emptyset,R_\Pi)$ である．Γと Π_B が適合することから，$\mathcal{A}_\Gamma\cap\mathcal{A}_B=\emptyset$ より $R_\Gamma\cap R_\Pi=\emptyset$ である．$\mathcal{A}_L$は$\overline{L}(K,V,W,R_\Pi)$ の部分集合であるから，$R_\Gamma\cap R_\Pi=\emptyset$ ならば $\mathcal{A}_\Gamma\cap\mathcal{A}_L=\emptyset$ は明らかである．いっぽう，$\mathcal{S}_\Gamma=\mathcal{S}_B=\mathcal{S}_L$ であるから，Π_LはΓに対する適合性の条件を満たす．したがってΠ_LはΓに適合する．