サブエルブラン領域の特定およびモデル生成定理証明への応用

何　立 風[†]　巣　宇 燕[††]　川 那 宜 充[†††]
加 藤 昇 平[†††]　中 村 剛 士[†††]　伊 藤 英 則[†††]

本稿では，節集合 $\mathcal{S}$ に現れる述語および関数の引数ごとに対応する $\mathcal{S}$ のエルブラン領域の部分領域（サブエルブラン領域と呼ぶ）を計算するアルゴリズムを提案する．また，節集合 $\mathcal{S}$ が充足不能であるための必要十分条件は，$\mathcal{S}$ に現れの各変数にその変数に対応するサブエルブラン領域の要素を代入して得られる基礎節のある有限集合が充足不能であるということを証明する．さらに，提案した手法のモデル生成定理証明への応用について述べ，ベンチマーク問題における実験結果を用いて提案手法の有効性を示す．

Specification of Sub-Herbrand Universes and Its Application to Model Generation Theorem Proving

Lifeng He,[†] Yuyan Chao,[††] Norimitsu Kawana,[†††]
Shohey Kato,[†††] Tsuyoshi Nakamura[†††] and Hidenori Itoh[†††]

This paper presents a method to specify a subuniverse of the Herbrand universe of a clause set $\mathcal{S}$ for each argument of the predicates and functions in $\mathcal{S}$. We prove that a clause set $\mathcal{S}$ is unsatisfiable if and only if there is a finite unsatisfiable set of ground instances of clauses of $\mathcal{S}$ derived by only instantiating the variables in $\mathcal{S}$ over their corresponding subuniverses derived by our algorithm, respectively. We introduce an application of our approach to model generation theorem proving for non-range-restricted problems, show our range-restriction algorithm and provide examples on the benchmark problems to demonstrate the power of our approach.

1. まえがき

多くの一階述語自動定理証明方法はエルブランの定理に基づいており，モデル発見と定理証明の両方に適応するモデル生成法 SATCHMO [10] はその中の 1 つである[3),5),7)~9),11)]．SATCHMO に基づくモデル生成定理証明法では，領域限定な問題だけ対象としている．非領域限定節集合については，領域限定の形式への変換を通じて SATCHMO で取り扱うことが可能である[3),10)]．このとき，非領域限定変数が節集合のエルブラン領域に基礎化される．これにより，推論中に必要なエルブラン領域の要素を特定できるという単一化のメリットが放棄されるため，モデル生成定理証明法は非領域限定問題に非効率な場合が多い[9),11),12)]．

本稿では，節集合 $\mathcal{S}$ にある述語記号と関数記号の引数ごとに対応する $\mathcal{S}$ のエルブラン領域の部分領域（**サブエルブラン領域**と呼ぶものとする）を計算するアルゴリズムを提案する．また，節集合 $\mathcal{S}$ が充足不能であるための必要十分条件は，$\mathcal{S}$ の各変数にたいしてその変数に対応するサブエルブラン領域の要素を代入して得られる基礎節のある有限集合が充足不能であるということを証明する．非領域限定問題にたいして，計算で得られたサブエルブラン領域は，通常のエルブラン領域の部分集合であるため，提案手法により，モデル生成定理証明の効率を改善できると考える．これにより，領域限定問題向けとされてきたモデル生成定理証明法に非領域限定問題への応用の可能性をもたらすことになる．

2. 準　備

本稿では，定理証明に関する基礎知識[1),4)] を有することを前提とする．

† 愛知県立大学情報科学部
Faculty of Information Science and Technology, Aichi Prefectural University
†† 名古屋産業大学環境情報ビジネス学部
Faculty of Enviornment, Information and Business, Nagoya Sangyo University
††† 名古屋工業大学知能情報システム学科
Department of Intelligence and Computer Science, Nagoya Institute of Technology

2.1 基本用語と表記法

本稿では，節 $\neg A_1 \vee \ldots \vee \neg A_m \vee C_1 \vee \ldots \vee C_n$ $(m, n \geq 0)$ を $A_1, \ldots, A_m \rightarrow C_1; \ldots; C_n$ の形で表現する．$A_1, \ldots, A_m$ を**本体**，$C_1; \ldots; C_n$ を**頭部**と呼ぶ．$n = 0$，つまり，節の頭部が空であるとき，その頭部を $false$ で表す．一方，$m = 0$，つまり，節の本体が空である場合，その本体を $true$ で表す．なお，$true$ は任意のモデルに充足され，$false$ はすべてのモデルに充足されない．

本稿では，述語記号，関数記号および個体記号は小文字のアルファベットで，変数は大文字のアルファベットで表す．ギリシア記号は述語と関数の引数，代入およびその他の必要情報を表現する．空節集合については ϕ で表す．また，混乱が生じない場合を除き，節集合の各節は同じ変数を持つことがないものとする．最後に，$\mathcal{I} \vdash A$ は A が論理的に $\mathcal{I}$ から証明できることを表し，$A \in \mathcal{I}$ は A が集合 $\mathcal{I}$ に含まれることを意味する．

2.2 エルブランの定理

節集合 $\mathcal{S}$ のエルブラン領域は次のように定義される[4)]．

定義 2.1 （エルブラン領域）$\mathcal{S}$ に現れるすべての個体記号の集合と関数記号の集合をそれぞれ $\mathcal{C}$，$\mathcal{F}$ とする．また，$H_0 = \mathcal{C}$ とする．ただし，$\mathcal{C} = \phi$ のとき，$H_0 = \{a\}$ とする．ここで，a は仮想個体記号（artificial constant）である．そして，各 $i = 0, 1, 2, \ldots,$ にたいして

$$H_{i+1} = H_i \cup \{f(t_1, \ldots, t_n) \mid f \in \mathcal{F}, t_j \in H_i, 1 \leq j \leq n\}$$

として，H_∞ すなわち $\lim_{i\to\infty} H_i$ を $\mathcal{S}$ のエルブラン領域と呼び，$H(\mathcal{S})$ で表す．

定義 2.2 （基礎アトムと基礎節）変数を含んでいないアトムを**基礎アトム**という．節集合 $\mathcal{S}$ 中の任意の節 C に含まれる変数にエルブラン領域の要素を代入してできる節を $\mathcal{S}$ の**基礎節**という．

定理 2.1 （エルブランの定理）節集合 $\mathcal{S}$ が充足不能であるための必要十分条件は，$\mathcal{S}$ の基礎節のある有限集合が充足不能であることである．

2.3 領 域 限 定

定義 2.3 （領域限定）節の本体に現れる変数を**領域限定変数**といい，節の頭部のみに現れる変数を**非領域限定変数**という．また，非領域限定変数を含まない節を**領域限定節**といい，そうでない節を**非領域限定節**という．さらに，すべての節が領域限定である節集合を**領域限定節集合**という．

領域限定節の最大の特徴は，節の本体が基礎アトムによって充足されれば，その節の頭部アトムが必ず基礎アトムになることである．

文献 3)，10) において指摘されたように，任意の非領域限定節集合は領域限定の形に変換できる．

定義 2.4 （領域限定変換）$\mathcal{S}$ を非領域限定節集合としたとき，$\mathcal{S}$ の領域限定の形 $\mathcal{S}'$ は $\mathcal{S}$ から次のように変換される．

1. 領域限定節にたいしては，同形変換を行う．
2. 非領域限定変数 X_1, ..., X_n を含む非領域限定節 $\mathcal{A} \rightarrow \mathcal{C}$ は，$\mathcal{A}, dom(X_1), \ldots, dom(X_n) \rightarrow \mathcal{C}$ に変換する．ただし，$\mathcal{A}$ が $true$ であるとき，$\mathcal{A}$ は省略する．
3. 節集合 $\mathcal{S}$ に含まれるすべての個体記号 c にたいして，節 $true \rightarrow dom(c)$ を加える．ただし，$\mathcal{S}$ が個体記号を含まない場合，節 $true \rightarrow dom(a)$ を加える．ここで，a は仮想個体記号である．
4. $\mathcal{S}$ に現れるすべての m 引数関数記号 f にたいして，節 $dom(Y_1), \ldots, dom(Y_m) \rightarrow dom(f(Y_1, \ldots, Y_m))$ を加える．

非領域限定変数が領域限定変換によってエルブラン領域に基礎化（instantiate）されることは明らかである．

例 2.1 $\mathcal{S}$ を次の非領域限定節集合 とする．

$p_1(c) \rightarrow false.$ $\quad p_2(f(c)) \rightarrow false.$

$true \rightarrow p_1(X); p_2(X).$

$\mathcal{S}$ の領域限定の形 $\mathcal{S}'$ は次のようになる．

$p_1(c) \rightarrow false.$ $\quad p_2(f(c)) \rightarrow false.$

$dom(X) \rightarrow p_1(X); p_2(X).$ $\quad true \rightarrow dom(c).$

$dom(Y) \rightarrow dom(f(Y)).$

なお，非領域限定変数 X は $\mathcal{S}$ のエルブラン領域に基礎化されていることが分かる．

2.4 SATCHMO

定理証明器 SATCHMO は，領域限定である節集合 $\mathcal{S}$ の充足可能性を判定するために，空集合から $\mathcal{S}$ の各節を順番に充足させながら，$\mathcal{S}$ のモデルを求めていく．$\mathcal{S}$ のすべての節を充足するモデルを発見できれば，$\mathcal{S}$ は充足可能であり，そうでなければ，$\mathcal{S}$ は充足不能である．

SATCHMO を非領域限定節集合 $\mathcal{S}$ に適用する前に，$\mathcal{S}$ を領域限定の形 $\mathcal{S}'$ に変換しなければならない．このとき，$\mathcal{S}$ の非領域限定変数は $\mathcal{S}$ のエルブラン領域の全体に基礎化される．それによって，非領域限定変数にエルブラン領域の要素を代入して得られる基礎節の多くは推論中に違反節となるため，証明に必要ない基礎節が数多く（無限になることもある）生成されることがある．そのため，モデル生成法に基づく

定理証明は非領域限定問題に効率的でないといわれている[9),11),12)].

例 2.2 $\mathcal{S}$ と $\mathcal{S}'$ をそれぞれ例 2.1 に示した節集合とする．
$\mathcal{S}$ を領域限定の形に変換するときに加えた節 $dom(X) \rightarrow dom(f(X))$ は再帰であるため，$\mathcal{S}'$ における SATCHMO の証明は停止しない．

しかし，上記の節集合 $\mathcal{S}$ が充足可能であることは導出原理で簡単に証明できる．

SATCHMO が $\mathcal{S}$ の充足可能性を証明できなかった理由は，領域限定変換によって実行中の推論に必要なエルブラン領域の要素を決定する単一化の利点を放棄したことにある[9),11),12)].

そこで，単一化の利点を領域限定変換に導入できれば，この問題を解決できると考えられる．本稿では，非領域限定変数を導出原理によって証明する際に，その変数のとりうるエルブラン領域の要素のみに制限する方法を提案する．

例 2.3 $\mathcal{S}$ を例 2.1 に示した節集合とする．
非領域限定節 $true \rightarrow p_1(X);\ p_2(X)$ について考える．導出原理によって証明を行う場合，$p_1(X)$ は節 $p_1(c) \rightarrow false$ のリテラル $\neg p_1(c)$ のみに演繹される．一方，$p_2(X)$ は節 $p_2(f(c)) \rightarrow false$ のリテラル $\neg p_2(f(c))$ のみに演繹される．すなわち，導出原理による証明では，上記の節にある非領域限定変数 X は c と $f(c)$ だけに基礎化される．このことから，変数 X にたいして，c と $f(c)$ 以外のエルブラン領域の要素を考慮する必要はない．したがって，節集合 $\mathcal{S}$ は次の領域限定の形 $\mathcal{S}''$ に変換できる．

$p_1(c) \rightarrow false.\quad p_2(f(c)) \rightarrow false.$
$dom(X) \rightarrow p_1(X); p_2(X).\quad true \rightarrow dom(c).$
$true \rightarrow dom(f(c)).$

SATCHMO は $\mathcal{S}''$ が充足可能であることをただちに証明できる．

3. サブエルブラン領域の特定

本章では，節集合の述語記号または関数記号の引数に対応するサブエルブラン領域の導出方法について述べる．便宜上，サブエルブラン領域を SHU（sub-Herbrand Universe）で，n 引数の述語記号または関数記号 α の i $(1 \leq i \leq n)$ 番目の引数を $\alpha^n\langle i\rangle$ で表す．また，引数 $\alpha^n\langle i\rangle$ の値は項 τ であることを $app(\tau, \alpha^n\langle i\rangle)$ で表す．

定義 3.1 （SHU を計算するアルゴリズム）$\mathcal{S}$ を節集合，$\alpha^n\langle i\rangle$ $(1 \leq i \leq n)$ に対応する SHU は次のように計算した集合 $\mathcal{H}$ である．

1. $\mathcal{H} = \phi$, $\mathcal{M} = \{\alpha^n\langle i\rangle\}$, $\mathcal{N} = \phi$ とする．
2. $\mathcal{M} = \phi$ が成立すれば，$\mathcal{H}$ は $\alpha^n\langle i\rangle$ に対応する SHU である．ただし，$\mathcal{H}$ に個体記号が含まれていなければ，$\mathcal{H} = \mathcal{H} \cup \{a\}$ とする．a は任意に選択した節集合 $\mathcal{S}$ の個体記号であり，**代表個体記号**と呼ぶ．ただし，$\mathcal{S}$ に個体記号が含まれていないとき，a は仮想個体記号である．
3. $\mathcal{M}$ が空でない場合, $\mathcal{M}$ の最左の要素 $\beta^m\langle j\rangle$ を $\mathcal{N}$ に移し，$\mathcal{S}$ に現れる各 $app(\varepsilon, \beta^m\langle j\rangle)$ $(1 \leq j \leq m)$ にたいして，
(1) ε が定数であれば，$\mathcal{H} = \mathcal{H} \cup \{\varepsilon\}$ とする．
(2) ε が関数 $f(\tau_1, \ldots, \tau_t)$ であれば，$\mathcal{H} = \mathcal{H} \cup \mathcal{V}\langle f\rangle$ とする．ここでは，$\mathcal{V}\langle f\rangle$ は関数 $f(\tau_1, \ldots, \tau_t)$ のとりうる値の集合を表す（定義 3.3 参照）．
(3) ε が変数 X である場合，C を X が現れる節とし，C に現れる各 X はある u 引数述語または関数 γ の k 番目の引数の値となっている．つまり，$app(X, \gamma^u\langle k\rangle)$ が C に現れているものとする．そのとき，$\gamma^u\langle k\rangle \notin \mathcal{M} \cup \mathcal{N}$ であれば，$\gamma^u\langle k\rangle$ を $\mathcal{M}$ に加える．
4. *2* に戻る．

定義 3.1 において，$\mathcal{M}$ が計算待ちの引数集合であり，$\mathcal{N}$ が計算済みの引数である．また，$\mathcal{M}$ がステップ 3 の (3) の場合だけ拡張される．$\beta^m\langle j\rangle$ を処理中の引数とする．引数 $\gamma^u\langle k\rangle$ が $\mathcal{M}$ に加えられるのは，ある変数 X にたいして $app(X, \beta^m\langle j\rangle)$ と $app(X, \gamma^u\langle k\rangle)$ が同じ節 C に現れる場合に限られている．そのとき，任意の代入において，$\beta^m\langle j\rangle$ の値と $\gamma^u\langle k\rangle$ の値は同値である．このような引数 $\beta^m\langle j\rangle$ および $\gamma^u\langle k\rangle$ を**同ドメイン引数**と呼ぶ[☆]．以上の議論は $\mathcal{M} \cup \mathcal{N}$ にあるすべての要素に拡張できる．よって，$\mathcal{M} \cup \mathcal{N}$ にあるすべての引数は同ドメイン引数である．そのゆえ，定義 3.1 のアルゴリズムが停止したとき，$\mathcal{N}$ にあるすべての引数が同ドメイン引数である．

また，任意の節集合に含まれる節の数，述語記号と関数記号の数およびそれらの引数の数が有限であるため，上記のアルゴリズムは必ず有限ステップで終了することは明らかである．便宜のため，引数 $\alpha^n\langle i\rangle$ に対応する SHU を引数 $\alpha^n\langle i\rangle$ の SHU と呼び，$\mathcal{H}(\alpha^n\langle i\rangle)$ で表すこともある．

例 3.1 $\mathcal{S}$ を次の節集合とする．

☆ X を節 C に現れる変数とする．X を値とする引数を **X に対応する引数**と呼ぶ．そうすると，X に対応するすべての引数は同ドメイン引数である．

$$\neg p(f(a, X_1), X_2) \quad (1)$$
$$p(a, X_3) \vee p(X_4, X_3) \quad (2)$$
$$\neg p(X_5, X_6) \vee p(f(X_5, b), X_6) \quad (3)$$

定義 3.1 による，述語記号 p の 1 番目の引数 $p^2\langle 1\rangle$ の SHU（$\mathcal{H}_1$ で表す）は次のように計算できる．

(1) 最初に，$\mathcal{H}_1 = \emptyset$，$\mathcal{M}_1 = \{p^2\langle 1\rangle\}$，$\mathcal{N}_1 = \emptyset$ とする．

(2) $\mathcal{M}_1$ の一番左の要素 $p^2\langle 1\rangle$ を $\mathcal{N}_1$ に移す．そうすると，$\mathcal{M}_1 = \emptyset$，$\mathcal{N}_1 = \{p^2\langle 1\rangle\}$ となる．
$p^2\langle 1\rangle$ は $\mathcal{S}$ に 5 つのところに現れている．つまり，$app(f(a, X_1), p^2\langle 1\rangle)$，$app(a, p^2\langle 1\rangle)$，$app(X_4, p^2\langle 1\rangle)$，$app(X_5, p^2\langle 1\rangle)$ と $app(f(X_5, b), p^2\langle 1\rangle)$ である．
$app(f(a, X_1), p^2\langle 1\rangle)$ と $app(f(X_5, b), p^2\langle 1\rangle)$ にたいして，定義 3.1 においてステップ 3 の (2) にあたる．そのため，$\mathcal{H}_1 = \mathcal{H}_1 \cup \mathcal{V}\langle f\rangle = \mathcal{V}\langle f\rangle$ となる．
$app(a, p^2\langle 1\rangle)$ においては，定義 3.1 のステップ 3 の (1) にあたる．そのため，$\mathcal{H}_1 = \mathcal{H}_1 \cup \{a\} = \{a\} \cup \mathcal{V}\langle f\rangle$ となる．
$app(X_4, p^2\langle 1\rangle)$ および $app(X_5, p^2\langle 1\rangle)$ にたいして，定義 3.1 においてステップ 3 の (3) にあたる．変数 X_4 は $\mathcal{S}$ にほかのどこにも現れていないため，なにもしなくてもよい．一方，$\mathcal{S}$ には，$app(X_5, f^2\langle 1\rangle)$ が存在するため，$f^2\langle 1\rangle$ は $\mathcal{M}_1$ に加えられる．そうすると，$\mathcal{M}_1 = \{f^2\langle 1\rangle\}$ となる．

(3) $\mathcal{M}_1$ の一番左の要素 $f^2\langle 1\rangle$ を $\mathcal{N}_1$ に移す．そうすると，$\mathcal{M}_1 = \emptyset$，$\mathcal{N}_1 = \{p^2\langle 1\rangle, f^2\langle 1\rangle\}$ となる．
$f^2\langle 1\rangle$ は $\mathcal{S}$ において 2 カ所に現れている．つまり，$app(a, f^2\langle 1\rangle)$ と $app(X_5, f^2\langle 1\rangle)$ である．
$app(a, f^2\langle 1\rangle)$ にたいして，定義 3.1 のステップ 3 の (1) にあたる．しかし，$a \in \mathcal{H}_1$ はすでに成立するから，$\mathcal{H}_1$ は変わらない．
$app(X_5, f^2\langle 1\rangle)$ にたいして，定義 3.1 のステップ 3 の (3) にあたる．$\mathcal{S}$ には $app(X_5, p^2\langle 1\rangle)$ が存在するが，$p^2\langle 1\rangle \in \mathcal{M} \cup \mathcal{N}$ のため，そのままでよい．

(4) $\mathcal{M}_1 = \emptyset$ となるから，定義 3.1 のステップ 2 により，$p^2\langle 1\rangle$ に対応する SHU の計算は終了する．ここで，$\mathcal{H}_1 = \{a\} \cup \mathcal{V}\langle f\rangle$，および $\mathcal{N}_1 = \{p^2\langle 1\rangle, f^2\langle 1\rangle\}$ となる．これによって，引数 $f^2\langle 1\rangle$ の SHU も $\mathcal{H}_1$ となる．

同様に，述語記号 p の第 2 引数 $p^2\langle 2\rangle$ に対応する SHU（$\mathcal{H}_2$ で表す）および関数記号 f の第 2 引数 $f^2\langle 2\rangle$ に対応する SHU（$\mathcal{H}_3$ で表す）はそれぞれ以下のようになる．

$$\mathcal{H}_2 = \{a\}$$
$$\mathcal{H}_3 = \{b\}$$

定義 3.2（すべての引数の SHU を計算するアルゴリズム）$\mathcal{S}$ を節集合とする．$\mathcal{S}$ に現れるすべての述語記号や関数記号の引数に対応する SHU は次のように求められる．

1. $\mathcal{T}$ を $\mathcal{S}$ に現れるすべての述語記号や関数記号の引数の集合とする．また，$j = 0$ とする．

2. $\mathcal{T}$ が空であれば，$\mathcal{H}_1, \ldots \mathcal{H}_j$ が導出した SHU である．$\mathcal{N}_k$ $(1 \le k \le j)$ に現れるすべての引数は同ドメイン引数であり，同一 SHU $\mathcal{H}_k$ を持つ．

3. $\alpha^n\langle i\rangle$ を $\mathcal{T}$ の 1 番目の要素とし，$j = j + 1$ とする．定義 3.1 によって，$\alpha^n\langle i\rangle$ の SHU $\mathcal{H}_j$ と $\alpha^n\langle i\rangle$ の同ドメイン引数集合 $\mathcal{N}_j$ を計算する．そこで，$\mathcal{T}$ から $\mathcal{N}_j$ に属する要素があれば除去する．

4. *2* に戻る．

例 3.2 例 3.1 に示した節集合 $\mathcal{S}$ を考える．$\mathcal{S}$ にあるすべての引数の SHU は次のように計算できる．

(1) 最初は，$\mathcal{T} = \{p^2\langle 1\rangle, p^2\langle 2\rangle, f^2\langle 1\rangle, f^2\langle 2\rangle\}$ であり，$j = 0$ となる．

(2) $j = 1$．$\mathcal{T}$ の最初の要素は，$p^2\langle 1\rangle$ である．例 3.1 により，$p^2\langle 1\rangle$ に対応する SHU，つまり，$\mathcal{H}_1$ は $\{a\} \cup \mathcal{V}\langle f\rangle$ である．また，$p^2\langle 1\rangle$ の同ドメイン引数集合 $\mathcal{N}_1$ は $\{p^2\langle 1\rangle, f^2\langle 1\rangle\}$ である．$\mathcal{N}_1$ の要素を $\mathcal{T}$ から削除すると，$\mathcal{T} = \{p^2\langle 2\rangle, f^2\langle 2\rangle\}$ となる．

(3) $j = 2$．$\mathcal{T}$ の最初の要素は，$p^2\langle 2\rangle$ である．例 3.1 により，$p^2\langle 2\rangle$ に対応する SHU，つまり，$\mathcal{H}_2$ は $\{a\}$ であり，$p^2\langle 2\rangle$ の同ドメイン引数集合 $\mathcal{N}_2$ は $\{p^2\langle 2\rangle\}$ である．$\mathcal{N}_2$ の要素を $\mathcal{T}$ から削除すると，$\mathcal{T} = \{f^2\langle 2\rangle\}$ となる．

(4) $j = 3$．$\mathcal{T}$ の最初の要素は，$f^2\langle 2\rangle$ である．例 3.1 により，$f^2\langle 2\rangle$ に対応する SHU，つまり，$\mathcal{H}_3$ は $\{b\}$ であり，$f^2\langle 2\rangle$ の同ドメイン引数集合 $\mathcal{N}_3$ は $\{f^2\langle 2\rangle\}$ である．$\mathcal{N}_3$ の要素を $\mathcal{T}$ から削除すると，$\mathcal{T} = \emptyset$ となる．

(6) $\mathcal{T} = \emptyset$ のため，計算は終了する．$\mathcal{H}_1$，$\mathcal{H}_2$ および $\mathcal{H}_3$ は導出した $\mathcal{S}$ の SHU である．

定義 3.3（エルブラン領域の形の SHU を計算するアルゴリズム）$\mathcal{S}$ を節集合，$f_1, \ldots, f_m$ を $\mathcal{S}$ に現れる関数記号とする．また，$\mathcal{H}_1, \ldots, \mathcal{H}_n$ を定義 3.2 によって導出した SHU とし，$\mathcal{V}\langle f_j\rangle$ $(1 \le j \le m)$ を $\mathcal{H}_1, \ldots, \mathcal{H}_n$ に現れる関数記号のとりうる値の集合とする．
$\mathcal{H}_i^*(0)$ $(1 \le i \le n)$ を $\mathcal{H}_i$ に現れる個体記号の集合と

して，$\mathcal{V}^*(f_j, 0) = \phi$ $(1 \leq j \leq m)$ とする．
f_j $(1 \leq j \leq m)$ を h_j 引数の関数記号，$\mathcal{H}_{u_t}$ $(1 \leq u_t \leq n)$ を $f_j^{h_j}\langle t \rangle$ $(1 \leq t \leq h_j)$ に対応する SHU とする．また，$k = 0, 1, 2, \ldots$ にたいして，

$$\mathcal{V}^*(f_j, k+1) = \{f_j(\alpha_1, \ldots, \alpha_{h_j}) | \alpha_1 \in \mathcal{H}^*_{u_1}(k), \ldots, \alpha_{h_j} \in \mathcal{H}^*_{u_{h_j}}(k)\}$$

さらに，$1 \leq i \leq n$ の各 i にたいして，

$$\mathcal{H}^*_i(k+1) = \{\mathcal{H}^*_i(k) \cup \mathcal{V}^*(f_v, k+1) | \mathcal{V}\langle f_v \rangle \in \mathcal{H}_i\}$$

このとき，$\mathcal{V}\langle f_j \rangle = \mathcal{V}^*(f_j, \infty)$（つまり，関数 f_j のとりうる値の集合である），$\mathcal{H}^*_i(\infty)$ は $\mathcal{H}_i$ のエルブラン領域の形である．

定義 3.3 によって導出した各 $\mathcal{H}^*_i(\infty)$ の各要素は与えられた節集合 $\mathcal{S}$ に現れる定数記号と関数記号だけを含んでいるため，各 $\mathcal{H}^*_i(\infty)$ は節集合 $\mathcal{S}$ のエルブラン領域の部分領域であることは明らかである．以下，SHU のエルブラン領域の形を SHU^* で表す．

例 3.3　$\mathcal{S}$ を例 3.1 に示した節集合とする．$\mathcal{H}_1$，$\mathcal{H}_2$ および $\mathcal{H}_3$ は例 3.2 に導出した各 SHU とする．$\mathcal{H}_1$，$\mathcal{H}_2$ および $\mathcal{H}_3$ にある定数記号の集合はそれぞれ $\mathcal{C}_1 = \{a\}$，$\mathcal{C}_2 = \{a\}$ と $\mathcal{C}_3 = \{b\}$ である．また，$f^2\langle 1 \rangle$ の SHU は $\mathcal{H}_1$ であり，$f^2\langle 2 \rangle$ の SHU は $\mathcal{H}_3$ である．
定義 3.2 により，エルブラン領域の形の各 SHU は次のように計算できる．

$$\begin{aligned}
&\mathcal{V}^*(f, 0) = \emptyset \\
&\mathcal{H}^*_1(0) = \mathcal{C}_1 = \{a\} \\
&\mathcal{H}^*_2(0) = \mathcal{C}_2 = \{a\} \\
&\mathcal{H}^*_3(0) = \mathcal{C}_3 = \{b\} \\
&\mathcal{V}^*(f, 1) = \{f(\alpha_1, \alpha_2) | \alpha_1 \in \mathcal{H}^*_1(0), \alpha_2 \in \mathcal{H}^*_3(0)\} \\
&\qquad = \{f(a, b)\} \\
&\mathcal{H}^*_1(1) = \mathcal{H}^*_1(0) \cup \mathcal{V}^*(f, 1) = \{a, f(a, b)\} \\
&\mathcal{H}^*_2(1) = \mathcal{H}^*_2(0) = \{a\} \\
&\mathcal{H}^*_3(1) = \mathcal{H}^*_3(0) = \{b\}
\end{aligned}$$

$$\begin{aligned}
&\mathcal{V}^*(f, 2) = \{f(\alpha_1, \alpha_2) | \alpha_1 \in \mathcal{H}^*_1(1), \alpha_2 \in \mathcal{H}^*_3(1)\} \\
&\qquad = \{f(a, b), f(f(a, b), b)\} \\
&\mathcal{H}^*_1(2) = \mathcal{H}^*_1(1) \cup \mathcal{V}^*(f, 2) \\
&\qquad = \{a, f(a, b), f(f(a, b), b)\} \\
&\mathcal{H}^*_2(2) = \mathcal{H}^*_2(1) = \{a\} \\
&\mathcal{H}^*_3(2) = \mathcal{H}^*_3(1) = \{b\} \\
&\qquad \vdots \\
&\mathcal{V}^*(f, \infty) = \{f(a, b), f(f(a, b), b), \\
&\qquad\qquad f(f(f(a, b), b), b), \ldots\} \\
&\mathcal{H}^*_1(\infty) = \{a, f(a, b), f(f(a, b), b), \\
&\qquad\qquad f(f(f(a, b), b), b), \ldots\} \\
&\mathcal{H}^*_2(\infty) = \{a\} \\
&\mathcal{H}^*_3(\infty) = \{b\}
\end{aligned}$$

$\mathcal{S}$ のエルブラン領域は $\{a, b, f(a, a)$, $f(a, b), f(b, a)$, $f(b, b)$, $f(f(a, a), a)$, $f(f(a, a), b)$, $f(f(a, b), a)$, $f(f(a, b), b)$, $f(f(b, a), a)$, $f(f(b, a), b)$, $f(f(b, b), a)$, $\ldots\ldots\}$ である．各 $\mathcal{H}^*_i(\infty)$ は $\mathcal{S}$ のエルブラン領域の部分集合であることは明らかである．

例 3.4　$\mathcal{S}$ を例 2.1 の節集合とする．このとき定義 3.2 と定義 3.1 によると，

$$\begin{aligned}
&\mathcal{T} = \{p_1^1\langle 1 \rangle, p_2^1\langle 1 \rangle, f^1\langle 1 \rangle\} \\
&\mathcal{H}_1 = \{c\} \cup \mathcal{V}\langle f \rangle \\
&\mathcal{N}_1 = \{p_1^1\langle 1 \rangle, p_2^1\langle 1 \rangle\} \\
&\mathcal{H}_2 = \{c\} \\
&\mathcal{N}_2 = \{f^1\langle 1 \rangle\}
\end{aligned}$$

さらに，定義 3.3 によると，

$$\begin{aligned}
&\mathcal{V}^*(f, 0) = \emptyset \\
&\mathcal{H}^*_1(0) = \{c\} \\
&\mathcal{H}^*_2(0) = \{c\} \\
&\mathcal{V}^*(f, 1) = \{f(\alpha) \mid \alpha \in \mathcal{H}^*_2(0)\} = \{f(c)\} \\
&\mathcal{H}^*_1(1) = \mathcal{H}^*_1(0) \cup \mathcal{V}^*(f, 1) = \{c, f(c)\} \\
&\mathcal{H}^*_2(1) = \{c\} \\
&\mathcal{V}^*(f, 2) = \{f(\alpha) \mid \alpha \in \mathcal{H}^*_2(1)\} = \{f(c)\} \\
&\mathcal{H}^*_1(2) = \mathcal{H}^*_1(1) \cup \mathcal{V}^*(f, 2) = \{c, f(c)\} \\
&\mathcal{H}^*_2(2) = \{c\} \\
&\qquad \vdots \\
&\mathcal{V}^*(f, \infty) = \{f(c)\} \\
&\mathcal{H}^*_1(\infty) = \{c, f(c)\} \\
&\mathcal{H}^*_2(\infty) = \{c\}.
\end{aligned}$$

$p_1^1\langle 1 \rangle$ と $p_2^1\langle 1 \rangle$ は同ドメイン引数であり，それらの SHU^* は $\mathcal{H}^*_1(\infty)$ である．一方，$f^1\langle 1 \rangle$ の SHU^* は $\mathcal{H}^*_2(\infty)$ であり，両方とも有限集合である．なお，前述のように，$\mathcal{S}$ のエルブラン領域は $\{c, f(c), f(f(c))$, $f(f(f(c))), \ldots\ldots\}$ であり，無限集合である．

4. 妥　当　性

定義 4.1　（SHU^* 基礎節）$\mathcal{S}$ を節集合，C を $\mathcal{S}$ に現れる節とする．節 C に現れる各変数 X に対応する引数の SHU^* の要素を代入して得られる基礎節を C の SHU^* 基礎節 という．

前述のように，節集合 $\mathcal{S}$ のすべての SHU^* は $\mathcal{S}$ のエルブラン領域の部分集合であるから，節 C の SHU^* 基礎節は節のエルブラン基礎節であること，また，その逆は必ずしも成立しないことは明らかである．

定理 4.1　（妥当性）節集合 $\mathcal{S}$ が充足不能であるための必要十分条件は，$\mathcal{S}$ の SHU^* 基礎節のある有限

集合が充足不能であることである．

定理 4.1 の証明は付録 A.1 に示す．

5. モデル生成法による定理証明への応用

定理 4.1 により，節集合 $\mathcal{S}$ の充足不能性を調べるとき，$\mathcal{S}$ のエルブラン領域における基礎節集合のかわりに，$\mathcal{S}$ の SHU^* 基礎節集合を考えればよい．そのため，非領域限定節集合を領域限定の形に変換するとき，非領域限定変数をその変数に対応する SHU^* の要素に制限すればよい．

定義 5.1 （SHU 領域限定変換）$\mathcal{S}$ を非領域限定節集合とする．定義 3.2 により $\mathcal{S}$ から導出したすべての SHU は $\mathcal{H}_1, \ldots, \mathcal{H}_t$ であるものとする．各 $\mathcal{H}_l$ $(1 \le l \le t)$ に対して，補助述語 dom_l を導入する．節集合 $\mathcal{S}$ は次のように SHU 領域限定節集合 $\mathcal{S}^*$ に変換する．

1. 領域限定節にたいしては，同形変換を行う．

2. $\mathcal{A} \rightarrow \mathcal{C}$ を $\mathcal{S}$ に現れる非領域限定節とする．また，$X_1, \ldots, X_n$ をその節に現れる非限定変数とする．さらに，$\mathcal{H}_{s_1}, \ldots, \mathcal{H}_{s_n}$ $(1 \le s_k \le t, 1 \le k \le n)$ をそれぞれ変数 $X_1, \ldots, X_n$ に対応する SHU とし，その節を $\mathcal{A}, dom_{s_1}(X_1), \ldots, dom_{s_n}(X_n) \rightarrow \mathcal{C}$ に変換する．ただし，$\mathcal{A}$ が $true$ である場合，$\mathcal{A}$ を省略してもよい．

3. 各 $\mathcal{H}_j$ （$1 \le j \le t$）に対して，

(1) $\mathcal{H}_j$ に現れる各個体記号 c に対して，節 $true \rightarrow dom_j(c)$ を加える．

(2) $\mathcal{V}\langle f\rangle$ を $\mathcal{H}_j$ に現れる記号とする．このとき，f は m 引数関数記号とし，$\mathcal{H}_{h_1}, \ldots, \mathcal{H}_{h_m}$ は f の各引数 $f^m\langle 1\rangle, \ldots, f^m\langle m\rangle$ に対応する SHU とする．節 $dom_{h_1}(Y_1), \ldots, dom_{h_m}(Y_m) \rightarrow dom_j(f(Y_1, \ldots, Y_m))$ を加える．

定義 5.1 に示した変換手続きを行うと，非領域限定変数がその変数に対応する SHU^* 領域の要素のみに基礎化されることは明らかである．

例 5.1 $\mathcal{S}$ を例 2.1 に示した節集合とする．例 3.4 によると，引数 $p^1_1\langle 1\rangle$ と $p^1_2\langle 1\rangle$ 対応する SHU は $\{c\} \cup \mathcal{V}\langle f\rangle$ であり，引数 $f^1\langle 1\rangle$ に対応する SHU は $\{c\}$ である．

定義 5.1 に示した手続きにより，$\mathcal{S}$ は次の SHU 領域限定の形 $\mathcal{S}^*$ に変換できる．

$$p_1(c) \rightarrow false. \quad p_2(f(c)) \rightarrow false.$$
$$dom_1(X) \rightarrow p_1(X); p_2(X). \quad true \rightarrow dom_1(c).$$
$$true \rightarrow dom_2(c). \quad dom_2(Y) \rightarrow dom_1(f(Y)).$$

SATCHMO は $\mathcal{S}^*$ が充足可能であることをただちに証明できる．

非領域限定節集合 $\mathcal{S}$ の非領域限定変数に対応する SHU は，$\mathcal{S}$ のエルブラン領域の真部分集合になることがあるため，非領域限定変数をその変数に対応する SHU の要素に限定して束縛することによって，基礎節の数が著しく減少することがある．これは，モデル生成法による定理証明を行うとき，前向き推論に用いられる節の数が減ることを意味する．

そのため，節集合が非領域限定であり，モデル生成法による定理証明を行った場合，提案手法によって領域限定変換を行えば，証明の効率が改善されることがある．

定理証明のベンチマーク問題集 TPTP library [13] version 2.5.0 には，5181 問が存在する．その中の 2881 問は非領域限定かつ非ホーン節を含む問題である．本稿で提案した手法が適用できる問題，すなわち，2 つ以上の SHU が導出される問題の数は 451 である．導出された最大の SHU の数は 271 である（問題 SYN837-1）．

提案手法を適用可能な 451 の問題にたいして，領域特定あり（$*$ が付いている）と領域特定なしのそれぞれの場合において，計算機（Intel PentiumIII/1 GHz，RAM256 MB）上で SATCHMO と UNSEARCHMO [6]☆を用いて実験を行った．なお，制限時間を 300 秒とした．実行時間については，領域限定変換，コンパイルおよび推論時間の合計である．

表 1 はそれぞれの場合において SATCHMO と UNSEARCHMO が解けた問題数を示す．また，領域特定なしで解けたすべての問題は領域特定ありでも解けることが判明している．

表 2 に一部の問題の実験結果についての詳細を示す．ここでは，実行時間の単位は秒である．*t.o.* はタイムアウト，すなわち，証明が制限時間の 300 秒を超えても終了しなかったことを意味する．表 2 により，提案手法は非領域限定問題に有効であることが分かる．

6. おわりに

本稿では，節集合 $\mathcal{S}$ の充足不能性を調べるとき，$\mathcal{S}$ のエルブラン領域における基礎節の集合のかわりに，$\mathcal{S}$ のそれぞれの変数に，その変数に対応するサブエルブラン領域の要素を代入して得られた基礎節の集合だけを考えればよいことを示した．これによって，モデ

☆ UNSEARCHMOはlevel-cut [2] という効率化法をSATCHMOに取り込んだ証明器である．

表 1 実験結果（I）
Table 1 The experiment results for different provers (I).

Number of problems	SATCHMO	SATCHMO*	UNSEARCHMO	UNSEARCHMO*
451	79	136	89	142

表 2 実験結果（II）
Table 2 The experiment results for different provers (II).

Problem	Number of derived *SHU*s	SATCHMO	SATCHMO*	UNSEARCHMO	UNSEARCHMO*
COM002-2	12	*t.o.*	0.63	0.34	0.28
KRS016-1	2	*t.o.*	0.25	*t.o.*	0.27
MSC004-1	3	*t.o.*	*t.o.*	*t.o.*	85.88
NLP048-1	4	*t.o.*	1.41	*t.o.*	1.43
NLP059-1	6	*t.o.*	1.67	*t.o.*	1.65
NLP060-1	7	*t.o.*	*t.o.*	*t.o.*	92.34
NLP067-1	5	*t.o.*	2.12	*t.o.*	1.38
NLP131-1	6	*t.o.*	1.95	*t.o.*	1.93
NLP134-1	6	*t.o.*	1.57	*t.o.*	1.56
NLP138-1	7	*t.o.*	1.92	*t.o.*	1.85
NLP162-1	7	*t.o.*	44.26	*t.o.*	85.03
NLP221-1	5	*t.o.*	0.89	*t.o.*	0.86
NLP230-1	5	*t.o.*	2.73	*t.o.*	1.66
NLP238-1	5	*t.o.*	1.85	*t.o.*	1.73
PUZ017-1	6	*t.o.*	3.79	*t.o.*	8.15
PUZ018-2	2	*t.o.*	108.89	2.76	0.29
PUZ019-1	2	*t.o.*	236.24	0.80	0.33
PUZ035-5	3	*t.o.*	0.10	*t.o.*	0.18
PUZ044-1	2	*t.o.*	0.16	*t.o.*	0.15
SET004-1	2	*t.o.*	*t.o.*	0.11	0.19
SET008-1	2	*t.o.*	*t.o.*	3.97	2.66
SET777-1	2	*t.o.*	0.17	*t.o.*	0.11
SYN423-1	2	*t.o.*	138.23	*t.o.*	143.27
SYN425-1	2	*t.o.*	107.80	*t.o.*	127.78
TOP004-2	3	*t.o.*	0.32	*t.o.*	0.35

ル生成法による非領域限定の節集合に対して定理証明を行う場合，前向き推論に用いられる節の数が減少し，探索空間が制限され，証明の効率化が実現できる．

今後の課題として，非領域問題における既存の効率手法[2]との比較や，サブエルブラン領域を構成する際に，節の本体と頭部での出現を区別して計算することによる，さらなる領域限定の可能性についての検討があげられる．また，本稿で提案したアルゴリズムの実装では，与えられた節集合に含まれるリテラルの数およびその節集合のサブエルブランの数が多い場合，サブエルブラン領域の特定の計算だけで 300 秒を超える場合がある．効率的な実装手法の開発も今後の課題となる．

謝辞 本研究を助成していただいた大幸財団に深く感謝する．

参 考 文 献

1) 有川節夫，原口 誠：述語論理と論理プログラミング，オーム社，東京 (1990).

2) Baumgartner, P., Furbach, U. and Niemela, I.: Hyper Tableaux, *Proc. European Workshop: Logics in Artificial Intelligence*, *JELIA*, LNAI 1126, pp.1–17 (1986).

3) Bry, F. and Yahya, A.: Positive Unit Hyperresolution Tableaux and Their Application to Minimal Model Generation, *J. of Automated Reasoning*, Vol.25, pp.35–82 (2000).

4) Chang, C.L. and Lee, K.C.T.: *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York (1973).

5) Fujita, H. and Hasegawa, R.: A Model Generation Theorem Prover in KL1 Using a Ramified-Stack Algorithm, *Logic Programming: Proc. 8th International Conference*, Paris, France, pp.535–548 (1991).

6) He, L.: UNSEARCHMO: Eliminating Redundant Search Space on Backtracking for Forward Chaining Theorem Proving, *Proc. Intl. Joint Conf. on Artificial Intelligence*, Seatle, U.S.A (2001).

7) He, L.: I-SATCHMO: an Improvement of

SATCHMO, *J. of Automated Reasoning*, Vol.27, pp.313–322 (2001).

8) He, L., Chao, Y., Simajiri, Y., Seki, H. and Itoh, H.: $\mathcal{A}$-SATCHMORE: SATCHMORE with Availability Checking, *New Generation Computing*, Vol.16, pp.55–74 (1998).

9) Loveland, D.W., Reed, D.W. and Wilson, D.S.: SATCHMORE: SATCHMO with RElevancy, *J. of Automated Reasoning*, Vol.14, pp.325–351 (1995).

10) Manthey, R. and Bry, F.: SATCHMO: a theorem prover implemented in Prolog, *Proc. 9th Intl. Conf. on Automated Deduction*, Argonne, Illinois, USA, pp.415–434 (1988).

11) Schutz, H. and Geisler, T.: Efficient Model Generation Through Compilation, *Information and Computation*, Vol.162, pp.138–157 (2000).

12) Stickel, M.E.: Upside-Down Meta-Interpretation of the Model Elimination Theorem-Proving Procedure for Deduction and Abduction, *J. of Automated Reasoning*, Vol.13, pp.189–210 (1994).

13) Sutcliffe, G. and Suttner, C.: http://www.cs.miami.edu/~tptp/

付　　録

A.1　定理 4.1 の証明

定義 A.1.1　（基礎項の深さ）α を基礎項とし，α の深さ $dep\langle\alpha\rangle$ を次のように定義する．

1. $dep\langle c\rangle = 1$．ただし，c は個体記号である．
2. $dep\langle f(\beta_1,\ldots,\beta_n)\rangle = h+1$．ただし，$f$ は n 引数の関数記号であり，h は $dep\langle\beta_1\rangle, \ldots, dep\langle\beta_n\rangle$ の中の最大値とする．

たとえば，$dep\langle f(a,b)\rangle = 2$，また，$dep\langle f(a,g(b,h(c)))\rangle$=4．任意の基礎項の長さは有限であるため，その項の深さは必ず計算できる．

定義 A.1.2　（導出原理による充足不能な基礎節集合の導出）$\mathcal{S}$ を充足不能な節集合とする．$\mathcal{S}$ の充足不能な基礎節集合（$\mathcal{S}_R$ で表す）は次のように求められる．

1. 節 C の因子 $C\sigma$ を生成する際，$C\sigma$ から重複したリテラルを削除するかわりに，ボックスで囲んでおく☆．
2. 2つの節から導出節を導く際，導出に使われるリテラルを削除するかわりに，ボックスで囲んでおく．

ボックスに囲まれたリテラルは処理済みであり，その後の導出に使わない．ただし，代入による変数の置換は普段と同様に行う．また，ボックスを無視すれば，導出節は与えられた節集合 $\mathcal{S}$ の節のインスタンスの選言と考えられる．すべてのリテラルがボックスに囲まれた導出節は空節を意味し，**拡張空節**と呼ぶ．このような拡張空節 R が導出されたとき，R に残った各変数 X（もしあれば）において，R にある引数 $\alpha^n\langle i\rangle$ において，$app(X,\alpha^n\langle i\rangle)$ が成立する．X に引数 $\alpha^n\langle i\rangle$ に対応する SHU に属するある定数記号を代入する．そうすると，R に現れるすべての $\mathcal{S}$ の節のインスタンスは $\mathcal{S}$ の基礎節となる．それらの基礎節の集合を $\mathcal{S}_R$ とする．明らかに，$\mathcal{S}_R$ は充足不能である．

例 A.1.1　$\mathcal{S}$ を例 3.1 に示した節集合，つまり，次の集合とする．

$$\neg p(f(a,X_1),X_2) \tag{1}$$
$$p(a,X_3) \vee p(X_4,X_3) \tag{2}$$
$$\neg p(X_5,X_6) \vee p(f(X_5,b),X_6) \tag{3}$$

定義 A.1.2 により，拡張空節 R は導出原理によって次のように導出できる．

$$\underbrace{\boxed{p(a,X_3)} \vee p(a,X_3)}_{\text{節 (2) のインスタンス}} \tag{4}$$

ここで，節 (4) は節 (2) の因子である．

$$\underbrace{\boxed{p(a,X_3)} \vee \boxed{p(a,X_3)}}_{\text{節 (2) のインスタンス}} \vee \underbrace{\boxed{\neg p(a,X_3)} \vee p(f(a,b),X_3)}_{\text{節 (3) のインスタンス}} \tag{5}$$

ここで，節 (5) は節 (4) と節 (3) の導出節である．

$$\underbrace{\boxed{p(a,X_3)} \vee \boxed{p(a,X_3)}}_{\text{節 (2) のインスタンス}} \vee \underbrace{\boxed{\neg p(a,X_3)} \vee \boxed{p(f(a,b),X_3)}}_{\text{節 (3) のインスタンス}} \vee \underbrace{\boxed{\neg p(f(a,b),X_3)}}_{\text{節 (1) のインスタンス}} \tag{6}$$

ここで，節 (6) は節 (5) と節 (1) の導出節である．導出節 (6) のすべてのリテラルはボックスに囲まれているため，節 (6) は拡張空節 R である．節 (6) に残った変数 X_3 に定数記号 a（ここで，節 (6) において，$app(X_3,p^2\langle 2\rangle)$ が成立する．また，例 3.3 により，$a \in \mathcal{H}(p^2\langle 2\rangle)$ が成立する）を代入すると，

$$\underbrace{\boxed{p(a,a)} \vee \boxed{p(a,a)}}_{\text{節 (2) の基礎節}} \vee \underbrace{\boxed{\neg p(a,a)} \vee \boxed{p(f(a,b),a)}}_{\text{節 (3) の基礎節}} \vee \underbrace{\boxed{\neg p(f(a,b),a)}}_{\text{節 (1) の基礎節}} \tag{7}$$

が得られる．

節 (7) から導出した $\mathcal{S}$ の充足不能な基礎節集合 $\mathcal{S}_R$ は次のようになる．

$\neg p(f(a,b),a)$	節 (1) の基礎節
$p(a,a) \vee p(a,a)$	節 (2) の基礎節
$\neg p(a,a) \vee p(f(a,b),a)$	節 (3) の基礎節

☆ 本稿では，議論を簡単にするために，節の因子を独立した節と見なす．

補題 A.1.1 $\mathcal{S}$ を節集合，T を導出原理による証明中に導出された因子または導出節，X を T に現れる変数とする．X に対応するすべての引数は同ドメイン引数である．

証明： 補題 A.1.1 を以下のように帰納法で示す．

$I(n)$：T_n を導出原理による証明において第 n ステップで導出した因子または導出節，X を T_n に現れる変数とする．このとき，X に対応するすべての引数は $\mathcal{S}$ において同ドメイン引数である．

Base case：$I(0)$ を示す．T_0 は $\mathcal{S}$ に含まれる節である．定義 3.1 により（1337 ページ脚注を参照），X に対応するすべての引数は $\mathcal{S}$ において同ドメイン引数である．

Induction step：$I(0)$, ..., $I(n)$ が成立することとし，$I(n+1)$ を示す．

T_{n+1} は節 C_1 と C_2 の導出節である．ここでは，C_1 と C_2 のいずれにおいても，$I(i)$ $(0 \le i \le n)$ において得られた節である．帰納法の仮定により，C_1 と C_2 に現れる任意の変数に対応するすべての引数は同ドメイン引数である．

変数 X が T_{n+1} に現れているため，C'（ただし，C' は C_1 か C_2 のいずれか）にも現れているはずである．T_{n+1} に存在し，C' にも存在する $app(X, \alpha_1^{n_1}\langle i_1\rangle)$, ..., $app(X, \alpha_r^{n_r}\langle i_r\rangle)$ において，帰納法の仮定により，$\alpha_1^{n_1}\langle i_1\rangle$, ..., $\alpha_r^{n_r}\langle i_r\rangle$ は同ドメイン引数である．

T_{n+1} に存在する他の X の出現は，X を C_1 もしくは C_2，あるいは C_1 かつ C_2 のいずれかにおいて存在する別の変数 Y_1, ..., Y_t に代入したものである．そのとき，各 Y_k（$1 \le k \le t$）において，$app(X, \beta_1^{m_1}\langle j_1\rangle)$，$app(Y_{s_1}, \beta_1^{m_1}\langle j_1\rangle)$，$app(Y_{s_1}, \beta_2^{m_2}\langle j_2\rangle)$，$app(Y_{s_2}, \beta_2^{m_2}\langle j_2\rangle)$，$app(Y_{s_2}, \beta_3^{m_3}\langle j_3\rangle)$，...，$app(Y_{s_u}, \beta_u^{m_u}\langle j_u\rangle)$，$app(Y_k, \beta_u^{m_u}\langle j_u\rangle)$ のような列が，C_1 もしくは C_2，あるいは C_1 かつ C_2 のいずれかに必ず存在する．ここで，$1 \le s_v \le t$，$1 \le v \le u$ である．また，$\beta_l^{m_l}\langle j_l\rangle$（$1 \le j_l \le m_u$，$1 \le l \le u$）は C_1 あるいは C_2 に現れる引数である．そのため，C_1 と C_2 において，変数 X に対応する引数と変数 Y_{s_1}，...，Y_{s_u} および Y_k に対応する引数が同ドメイン引数である．そのゆえ，T_{n+1} において，変数 X に対応するすべての引数は同ドメイン引数である．

T_{n+1} は節 C の因子の場合も，導出節と同様に示すことができる．

よって，$I(n+1)$ が成立する．

節 C を $p(X,Y,Y,Z) \vee q(Z) \vee p(U,U,V,V)$ とする．$p(X,X,X,X) \vee q(X) \vee \boxed{p(X,X,X,X)}$ は節 C の因子である．C の変数 Z は因子化において変数 X に代入された．C において，列 $app(X, p^4\langle 1\rangle)$，$app(U, p^4\langle 1\rangle)$，$app(U, p^4\langle 2\rangle)$，$app(Y, p^4\langle 2\rangle)$，$app(Y, p^4\langle 3\rangle)$，$app(V, p^4\langle 3\rangle)$，$app(V, p^4\langle 4\rangle)$ および $app(Z, p^4\langle 4\rangle)$ が存在する．定義 3.1 により，C において，変数 X，Y，Z，U と V に対応するそれぞれの引数は同ドメイン引数である．それら引数の集合は導出した因子において，変数 X に対応する引数の集合と同一である．

補題 A.1.2 $\mathcal{S}$ を充足不能の節集合，$\mathcal{S}_R$ を定義 A.1.2 により導出した $\mathcal{S}$ の充足不能の基礎節集合とする．$\mathcal{S}_R$ のすべての基礎節が SHU^* 基礎節である．

証明： $\mathcal{S}$ に現れる各定数記号 c にたいして，$\mathcal{S}$ に $app(c, \alpha^n\langle i\rangle)$ が存在すれば，定義 3.1 により，$c \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．ここで，$\mathcal{H}(\alpha^n\langle i\rangle)$ は引数 $\alpha^n\langle i\rangle$ の SHU を表す．

$\mathcal{S}$ に現れる各関数項 $f(\tau_1, \ldots, \tau_n)$ にたいして，$\mathcal{S}$ に $app(f(\tau_1, \ldots, \tau_n), \alpha^n\langle i\rangle)$ が存在すれば，定義 3.1 により，$\mathcal{V}\langle f\rangle \subset \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．

$\mathcal{S}$ に現れる変数 Y は導出原理による証明中に別の変数 X に代入されたとき，補題 A.1.1 により，変数 Y に対応する引数と変数 X に対応する引数は同ドメインである．

$\mathcal{S}$ に現れる変数 X は関数項 $f(\tau_1, \ldots, \tau_n)$ に代入されたとき，ある引数 $\alpha^n\langle i\rangle$ において，$app(X, \alpha^n\langle i\rangle)$ と $app(f(\tau_1, \ldots, \tau_n), \alpha^n\langle i\rangle)$ が $\mathcal{S}$ に存在する．定義 3.1 により，$\mathcal{V}\langle f\rangle \subset \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．

さらに，変数 X は定数記号 c に代入されたときを考える．この代入は導出原理の証明中に発生した場合，ある引数 $\alpha^n\langle i\rangle$ において，$app(X, \alpha^n\langle i\rangle)$ と $app(c, \alpha^n\langle i\rangle)$ が $\mathcal{S}$ に存在する．定義 3.1 により，$c \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．一方，X は導出原理による導出した拡張空節に現れる変数であり，充足不能の基礎節集合を導出するために，定数記号 c に代入された場合，定数記号 c の選び方により，その拡張空節に現れるある引数 $\alpha^n\langle i\rangle$ において，$app(X, \alpha^n\langle i\rangle)$ と $c \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．

$\mathcal{S}_R$ は基礎節の集合であるため，$\mathcal{S}_R$ に現れるすべての項は基礎項である．$\mathcal{S}_R$ の任意の基礎項 τ において，もし $app(\tau, \alpha^n\langle i\rangle)$ が $\mathcal{S}_R$ に存在すれば，$\tau \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立することを $dep\langle\tau\rangle$ における帰納法で示す．

$dep\langle\tau\rangle = 1$ のとき，τ は定数記号である．以上の議論により，$\tau \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．

$dep\langle\tau\rangle = i$（$1 \le i \le t$）のとき成立すると仮定する．このとき，$dep\langle\tau\rangle = t+1$ のときも成立することを示す．

一般性を失うことなく，$\tau = f(\rho_1, \ldots, \rho_m)$ とする．ただし，$dep\langle\rho_j\rangle \leq t$，$1 \leq j \leq m$．上の議論により，$\mathcal{V}\langle f\rangle \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．帰納法の仮定により，$\rho_j \in \mathcal{H}(f^m\langle j\rangle)$．また，定義 3.3 により，$f(\rho_1, \ldots, \rho_m) \in \mathcal{H}(\alpha^n\langle i\rangle)$．そのため，$dep\langle\tau\rangle = t+1$ のときも，$\tau \in \mathcal{H}(\alpha^n\langle i\rangle)$ が成立する．

上の議論および定義 4.1 により，$\mathcal{S}_R$ の各基礎節は SHU^* 基礎節である☆．

定理 4.1 は次のように証明できる．

証明：(⇒) $\mathcal{S}$ を充足不能の節集合，$\mathcal{S}_R$ を定義 A.1.2 により導出した充足不能の基礎節集合とする．補題 A.1.2 により，$\mathcal{S}_R$ のすべての節は SHU^* 基礎節である．$\mathcal{S}_R$ を $\mathcal{S}^*$ とすると，$\mathcal{S}^*$ が節集合 $\mathcal{S}$ の充足不能の SHU^* 基礎節集合である．

(⇐) 充足不能の SHU^* 基礎節集合 $\mathcal{S}^*$ が存在すると仮定する．$\mathcal{S}^*$ の各基礎節は $\mathcal{S}$ の基礎節でもあるため，エルブラン定理により，$\mathcal{S}$ は充足不能である．

(平成 15 年 9 月 11 日受付)
(平成 16 年 3 月 5 日採録)

何　立風（正会員）
1997 年名古屋工業大学工学研究科博士後期課程修了．博士（工学）．現在，愛知県立大学情報科学部助教授．マルチエージェント，定理証明，知識データベース，画像処理等に興味を持つ．

巣　宇燕
2000 年名古屋大学大学院人間情報文化工学研究科博士後期課程修了．2000 年 4 月から 2002 年 9 月まで，名古屋工業大学研究員．2002 年 9 月から名古屋産業大学講師．博士（学術）．定理証明，画像処理および図面理解，CAD 等に興味を持つ．

川那　宜充
2002 年名古屋工業大学知能情報システム学科卒業．同年電気情報工学専攻に入学，2004 年 4 月修了予定．人工知能，定理証明の分野に興味を持つ．

加藤　昇平（正会員）
1993 年名古屋工業大学工学部電気情報工学科卒業．1998 年同大学院博士後期課程修了．同年豊田工業高等専門学校助手，1999 年同講師．2002 年名古屋工業大学講師，2003 年同助教授．博士（工学）．推論・探索処理，分散・並列アルゴリズム，画像処理，知能ロボット等に興味を持つ．人工知能学会，電子情報通信学会各会員．

中村　剛士
1993 年名古屋工業大学工学部電気情報工学科卒業．1998 年同大学大学院博士後期課程修了．同年名古屋工業大学知能情報システム学科助手．2003 年同大学大学院工学研究科情報工学専攻助教授．博士（工学）．画像表現，感性情報処理，ソフトコンピューティング等に興味を持つ．電子情報通信学会，日本知能情報ファジィ学会，IEEE 各会員．

伊藤　英則（正会員）
1974 年名古屋大学大学院工学研究科博士課程満了．工学博士号取得．同年 NTT 入社，横須賀研究所勤務．1985 年（財）新世代コンピュータ技術開発機構出向．1989 年より名古屋工業大学教授．感性情報，自動推論，マルチメディア，マルチエージェント等に興味を持つ．電子情報通信学会，人工知能学会，ファジィ学会各会員．

☆ たとえば，例 A.1.1 に導出した各基礎節は SHU^* 基礎節である．