論　文

部分例示に基づく定理自動証明

正　員　山本　雅人[†]　　非会員　大柳　俊夫[††]　　正　員　大内　東[†]

Theorem Proving Based on the Partial Instantiation Technique

Masahito YAMAMOTO[†], *Member*, Toshio OHYANAGI[††], *Nonmember*
and Azuma OHUCHI[†], *Member*

あらまし　定理自動証明に対するこれまでの研究として，1965年にRobinsonによって提案された導出原理に基づくさまざまな方法，戦略などが提案されており，現在の定理証明システムの多くはこの導出原理に基づいている．これに対して，Jeroslowは1988年に部分例示手法を提案している．しかしながら，Jeroslowの提案した方法は関数記号を含まない論理式のみを対象としていること，また，節形式を前提としていないために効率が悪いことなどの欠点があった．筆者らはこのJeroslowの方法を改善して，関数記号を含まない節形式の論理式の充足可能性を判定する手続きを提案して，その有効性を示している．本論文では，この手続きを拡張した定理証明手続きを提案する．すなわち，関数記号を含む論理式を扱えるようにしている．また，本手続きが完全であることの証明を与える．更に，比較実験を行い本手続きの有効性について検討する．

キーワード　部分例示，充足可能性問題，定理自動証明，導出原理，Davis-Putnamの方法

1. まえがき

定理の自動証明は人工知能の分野における重要な研究の一つであり，質疑応答システム，プログラムの合成，分析などの分野に応用されている(1)．定理自動証明に対するこれまでの研究として，1965年にRobinsonによって提案された導出原理に基づくさまざまな方法，戦略などが提案されており，現在の定理証明システムの多くはこの導出原理に基づいている(2)．これに対して，Jeroslowは1988年に部分例示手法を提案している(3)．部分例示手法は，論理式中のすべての変数にHerbrand空間の要素を代入するのではなく，一部の変数にのみHerbrand空間の要素を代入することによって，節数の増大を防ぐものである．また，その際に原子式への真理値の割当てを利用することも特徴の一つである．しかしながら，Jeroslowの方法は関数記号を含まない論理式のみを対象としていること，また，節形式を前提としていないために効率が悪いことなどの欠点があった．筆者らはこのJeroslowの方法を改善して，関数記号を含まない節形式の論理式の充足可能性を判定する手続きを提案して，その有効性を示している(4)．

本論文では，部分例示に基づいた定理証明手続きを提案する．本論文で提案する手続き（以下，本手続きと呼ぶ）は，文献(4)で提案した手続きを，関数記号を含む論理式を扱えるように拡張したものであり，このことによって，導出原理に基づく方法と同様の論理式を扱うことができる．また，本手続きが完全であることの証明を与える．更に，比較実験を行い本手続きの有効性について検討する．

以下，2.で本論文で用いる用語や記号を定義し，3.では，本手続きの詳細について述べ，4.において本手続きの完全性を証明する．5.では二つの例題を用いて，本手続きの実際の処理について説明する．6.で実験結果から本手続きの有効性について考察し，最後に7.で結論と今後の研究について述べる．

2. 諸定義

定理の自動証明は，以下のようなm個の節からなる節集合Sに対して，Sの充足可能性を調べることに帰着できる．

$$S=\{C_1,\cdots,C_m\}$$

[†] 北海道大学工学部情報工学科，札幌市
Faculty of Engineering, Hokkaido University, Sapporo-shi, 060 Japan
[††] 札幌医科大学保健医療学部一般教育科，札幌市
School of Health Sciences, Sapporo Medical University, Sapporo-shi, 060 Japan

```
1:  begin
2:    k ← 1, S_k ← S ;
3:    loop
4:      V_k ← φ, B_k ← φ ;
5:      V_k ← a variant independent valuation which makes S_k true ; (Fig. 2)
6:      if  V_k = φ  then  S is unsatisfiable ; (exit)
7:      B_k ← a set of all proper blockages in S_k for V_k ; (Fig. 3)
8:      if  B_k = φ  then  S is satisfiable ; (exit)
9:      S_{k+1} ← an expansion of S_k ; (Fig. 4)
10:     k ← k + 1 ;
11:   endloop ;
12: end ;
```

図1　定理証明手続きの概要
Fig. 1　The framework of the proposed theorem proving procedure.

本論文で提案する定理証明手続きでは，Herbrandの定理に基づき，S が充足不能であることを充足不能な S の節の基礎例を求めることによって示す．以下では，本手続きの記述に必要となる用語や記号について定義する．

F を論理式，θ を代入とするとき，$F\theta$ を F の例と言う．また，特に $F\theta$ が変数を含まないとき，$F\theta$ を F の基礎例と言う．ある原子論理式（以下，原子式と呼ぶ）A に対して，A の変数名を変えることによって得られる原子式を A の変種と言う†．

[定義 1]　節 C が節 C_i の例であるなら，C_i は C を被覆すると言う．また，C_i が C を被覆し，かつ C_i 自身を除く C_i の例が S に含まれないなら，C_i は C を直接被覆すると言う．

[定義 2]　変種であるすべての原子式に同じ真理値を割り当てる真理値の割当てを変種独立割当てと言う．

[定義 3]　節集合 S_k のすべての要素が S の例であり，かつ，$S_k \supseteq S$ であるとき，S_k を S の拡大と言う．

[定義 4]　V_k が S_k を真とする変種独立割当てであるとする．また，F, G を l_F, l_G にそれぞれ現れる原子式であるとする．但し，l_F, l_G は S_k 中の節 C_i, C_j にそれぞれ現れるリテラルである．このとき，以下の条件をすべて満足する原子式 F, G の組を純障害物と言い，$\langle F, G\rangle_{i,j}$ と表す．

（1）　F と G は V_k で異なる真理値を割り当てられている．

（2）　l_F と l_G の両方が V_k で真である．

（3）　$F\sigma = G\sigma$ となる最汎単一化作用素 σ が存在し，$F\sigma$ のある例が C_i' と C_j' の両方に現れる．但し，C_i', C_j' は C_i, C_j によってそれぞれ直接被覆されているとする．

3.　定理証明手続き

節集合 S の充足可能性を調べる本手続きの概要を図1に示す．なお，本手続きにおける k 回目の繰返し（図1の Step 4～Step 10）を第 k ループと呼び，第 k ループで扱う節集合を S_k とする．但し，$S_1 = S$ である．

本手続きは，以下の基本的な三つの処理を節集合 S の充足可能性が判定されるまで繰り返す．

（1）［変種独立割当て］　S_k を真とする変種独立割当て V_k を求める（Step 5）．

（2）［純障害物の発見］　V_k に対して，S_k 中のすべての純障害物を求める（Step 7）．

（3）［純障害物の解消］　S_k 中のすべての純障害物を解消した S_k の拡大 S_{k+1} を生成する（Step 9）．

上記の（2）および（3）に関して，論理式に関数記号が現れない場合は S の節の基礎例が有限であるため，文献(4)で提案した手続きのように第 k ループにおいて一つの純障害物を発見し，その一つの純障害物の解消のみを行うことでも完全性が保証できた．一方，論理式に関数記号が現れる場合は S の節の基礎例は有限ではないため，第 k ループにおいて一つの純障害物を発見しそれのみを解消する方法では，充足不能となる基礎例を生成する保証がない．しかし，本手続きのように，V_k に対する S_k 中のすべての純障害物を発見し解消することにより，**4.** で述べるように完全性を保証することができる．

以下では，(1)～(3) について順次，**3.1**, **3.2**, **3.3** で詳細に述べる．

† 特に断りがない限り，記号論理に関する用語および表記は文献(5)による．

```
1: begin
2:    transform S_k into a set of propositional clauses S_k^p ;
3:    I ← φ ;
4:    I ← an interpretation which satisfies S_k^p ;
5:    if  I = φ  then  exit;
6:    V_k ← a variant independent valuation obtained from I ;
7: end
```

図2 変種独立割当ての発見手続き
Fig. 2 The procedure for searching a variant independent valuation.

3.1 変種独立割当て

本手続きの第 k ループにおいて，S_k を真とする変種独立割当てを求める．この手続きを図2に示す．

この処理はまず S_k 中のすべての原子式を以下の規則を用いて命題変数に変換し，命題論理式の節集合 S_k^p を得る．

- ある原子式 F の変種であるすべての原子式は F と同じ命題変数にする．
- ある原子式 F の変種でないすべての原子式は F と異なる命題変数にする．

次に，この命題論理式の節集合 S_k^p の充足可能性問題を解く．もし，S_k^p が充足不能なら，明らかに S_k を真とする変種独立割当ては存在しない．S_k^p を真とする解釈が存在すれば，S_k 中の原子式の真理値を，S_k^p 中の対応する命題変数の真理値とする．これが，S_k を真とする変種独立割当てである．

S_k を真とする変種独立割当ては一般には複数存在すると考えられるが，その中のどの割当てを用いても手続きの完全性は失われない．しかし，証明に要する繰返し回数 k は，用いる変種独立割当てによって大きく影響されることが予想され，高速に証明を行うためには，変種独立割当てを求める戦略が重要となる．

[例 1] 以下のような節集合を考える．

$$S_k = \{P(x), \sim P(a) \vee Q(f(y), b), \sim Q(f(z), b)\}$$

$P(x)$ と $P(a)$ は，変種となる原子式が S_k 中に存在しないので，それぞれ異なる命題変数 p_1，p_2 とする．また，$Q(f(y), b)$ と $Q(f(z), b)$ は互いに変種であるので，同じ命題変数 p_3 とする．従って，節集合 S_k は以下の節集合 S_k^p に変換できる．

$$S_k^p = \{p_1, \sim p_2 \vee p_3, \sim p_3\}$$

ここで，解釈 $\{p_1, \sim p_2, \sim p_3\}$ の下で S_k^p は真となり，S_k を真とする変種独立割当て V_k を得ることができる．この割当て V_k を以下のように表記する．ここで，V_k において，$P(x)$，$P(a)$，$Q(f(y), b)$，$Q(f(z), b)$ はそれぞれ，$True$，$False$，$False$，$False$ に割り当てられていることを意味する．

$$S_k = \{\underbrace{P(x)}_{True}, \sim \underbrace{P(a)}_{False} \vee \underbrace{Q(f(y), b)}_{False}, \sim \underbrace{Q(f(z), b)}_{False}\}$$

3.2 純障害物の発見

3.1 で，S_k を真とする変種独立割当て V_k が見つかれば，V_k に対して，S_k 中に存在するすべての純障害物を求めなければならない．この手続きを図3に示す．

S_k 中の異なる二つの節 C_i，C_j 中にそれぞれ現れる原子式 F，G が純障害物となるために，F と G が単一化可能であることは純障害物の定義から明らかである．しかし，実際に F と G の単一化を試みるのは F，G が V_k において異なる真理値を割り当てられていて，l_F と l_G がともに V_k で真となるときのみである．但し，l_F，l_G は F，G がそれぞれ現れているリテラルである．この結果，単一化可能であれば，F と G の最汎単一化作用素 σ を用いて，$C_i\sigma$ と $C_j\sigma$ がともに S_k の要素であるかを調べる．すなわち，$F\sigma$ の例の現れる節が C_i と C_j に直接被覆されているかどうかを調べる．このように，S_k 中のすべての異なる二つの節の組について純障害物を求め，その集合を B_k とする．

[例 2] 以下の節 C_1，C_2，C_3 からなる節集合 S_k を真とする以下の変種独立割当て V_k が与えられている．

$$C_1 = \underbrace{P(x)}_{True}$$

$$C_2 = \sim \underbrace{P(a)}_{False} \vee \underbrace{Q(f(y), b)}_{False}$$

$$C_3 = \sim \underbrace{Q(f(z), b)}_{False}$$

この V_k に対して，C_1 中の $P(x)$ と C_2 中の $P(a)$ は純障害物である．なぜなら，$P(x)$ と $P(a)$ はとも

```
1:  begin
2:    for   all pairs of C_i, C_j (i ≠ j) in S_k   do
3:      for   all pairs of atomic formulas F, G occur in C_i, C_j, respectively   do
4:        if   ( F and G have opposite truth value in V_k   and
5:               both l_F and l_G are true in V_k   and
6:               F and G are unifiable )
7:        then begin
8:             σ ← a most general unifier for F and G ;
9:             if   ( C_iσ isn't an element of S_k   and
10:                   C_jσ isn't an element of S_k )
11:            then   B_k ← B_k ∪ {⟨F, G⟩_{i,j}} ;
12:       end ;
13: end ;
```

図3　すべての純障害物の発見手続き
Fig. 3　The procedure for finding all proper blockages.

に V_k で異なる真理値を割り当てられており，それらが現れるリテラル $P(x)$ と $\sim P(a)$ は V_k でともに真である．更に，$P(x)\sigma = P(a)\sigma$ となる最汎単一化作用素 $\sigma = \{a/x\}$ が存在し，$P(x)\sigma$ が C_1 の直接被覆している節 $P(a)$ と C_2 の直接被覆している節 $\sim P(a) \vee Q(f(a), b)$ の両方に現れるからである．
S_k 中には V_k に対して，純障害物は他に存在しないから，$B_k = \{\langle P(x), P(a)\rangle_{1,2}\}$ である．

3.3　純障害物の解消

3.2 で求めた純障害物の集合 B_k 中のすべての純障害物を解消するために S_k の拡大 S_{k+1} を生成する．この手続きを図4に示す．

$\langle F, G\rangle_{i,j}$ を S_k 中の V_k に対する純障害物の一つ，すなわち，B_k の要素の一つとし，σ を F と G の最汎単一化作用素であるとする．本手続きは，節 $C_i\sigma$ と $C_j\sigma$ を生成し，S' に保存していく．但し，$C_i\sigma = C_i$ のとき，または，$C_j\sigma = C_j$ のときは，それぞれ $C_i\sigma$，$C_j\sigma$ の生成は明らかに無駄となるため行わない．この処理を B_k のすべての要素について行った後，S' と S_k の和集合を S_{k+1} とする．S_{k+1} が S_k の拡大になっていることは容易にわかる．従って，S の拡大になっている．

生成した S_{k+1} には $C_i\sigma$ と $C_j\sigma$ が存在する．この二つの節において，$F\sigma$ と $G\sigma$ は $F\sigma = G\sigma$ であるから，純障害物にはなり得ない．また，F と G も S_{k+1} 中では純障害物にはならない．なぜなら，$F\sigma$ の例が現れる任意の基礎例は，S_{k+1} において $C_i\sigma$ か $C_j\sigma$ に直接被覆されるからである．

[例 3]　例2における節集合 S_k を真とする変種独立割当て V_k に対して，$B_k = \{\langle P(x), P(a)\rangle_{1,2}\}$ を解消するために，$C_1\sigma = P(a)$ を生成する．また，$C_2\sigma$ に関しては $C_2\sigma = C_2$ であるから生成しない．そして，$S_k \cup \{P(a)\}$ を S_{k+1} とする．

```
1:  begin
2:    S' ← φ
3:    for   all elements ⟨F, G⟩_{i,j} of B_k   do begin
4:      σ ← a most general unifier for F and G ;
5:      S' ← S' ∪ {C_iσ, C_jσ} ;
6:    end;
7:    S_{k+1} ← S_k ∪ S' ;
8:  end;
```

図4　S_k の拡大 S_{k+1} の生成手続き
Fig. 4　The procedure for computing an extension S_{k+1} of S_k.

4.　手続きの正当性と完全性

本手続きが正当でかつ完全であることを示すために，節集合 S が充足不能であることと S_p を真とする変種独立割当てが存在しないような有限の p が存在することが同値であることを示す．以下の定義および補題は定理1を示すために必要である．

[定義 5]　$T = \{D_1, \cdots, D_n\}$ を充足不能な S の節の基礎例の集合であるとする．また，$U_k = \{E_1, \cdots, E_n\}$ を多重集合とする．但し，各 E_i $(i = 1, \cdots, n)$ は D_i を直接被覆する S_k 中の節であるとする．

[補題 1]　T を充足不能な S の節の基礎例の集合であるとする．また，S_k を真とする変種独立割当て V_k が存在するとする．このとき，V_k に対して U_k 中の節に純障害物 $\langle F, G\rangle_{i,j}$ が存在する．但し，F，G はそれぞれ U_k 中の節 E_i，E_j $(i \neq j)$ に現れるとする．

(証明)　この補題が成り立たないとすると，U_k 中には純障害物が存在しない．このとき，以下のような割当てを考える．U_k の節の任意の基礎例 E_i' について，

E'_i を直接被覆する U_k 中の節 E_i が存在し，E_i は V_k において真である．従って，E_i には V_k で真となるリテラル l が存在する．また，E'_i は E_i の例であるから，E'_i には l の例 l' が存在する．このとき，l に現れる原子式に割り当てられている真理値を l' に現れている基礎原子式の真理値とする．この割当てによって，U_k の節の任意の基礎例 E'_i を真とすることができる．但し，ある基礎例 E'_i 中で真理値を割り当てられた基礎原子式 P が別の基礎例 E'_j 中で異なる真理値を割り当てられる場合がある．この場合，上記の割当て方法から，P が現れるリテラルは V_k において，E'_i と E'_j でともに真となっている．従って，P に異なる真理値が割り当てられるのは，E_i，E_j 中に例示すると P となる二つの原子式 F，G が存在し，F，G がそれぞれ現れるリテラル l_F，l_G が V_k においてともに真となるときのみ起こる．すなわち，純障害物 $\langle F, G\rangle_{i,j}$ が存在するときである．しかし仮定より，V_k において純障害物は存在しないのでこのような場合は生じない．従って，このような割当てにより U_k 中の節の基礎例からなる任意の節集合は充足可能となる．T は，U_k 中の節の基礎例からなる集合であるから T は充足可能であり仮定に反する．　(証明終)

[定義 6]　節 C に対して，$\|C\|$ はその節での定数，関数記号，述語記号の出現の個数を表すとする．また，節集合 $S = \{C_1, \cdots, C_m\}$ に対して，$\|S\|$ を $\|S\| = \|C_1\| + \|C_2\| + \cdots + \|C_m\|$ と定義する．

[補題 2]　T を充足不能な S の節の基礎例の集合であるとする．また，S_k を真とする変種独立割当て V_k が存在するとする．このとき，本手続きにおける S_k の拡大により，S_{k+1} に対して $\|U_k\| < \|U_{k+1}\|$ となる．

(証明)　補題 1 より，U_k 中に純障害物 $\langle F, G\rangle_{i,j}$ が存在する．但し，F，G は U_k 中の節 E_i，E_j $(i \neq j)$ にそれぞれ現れるとする．このとき，σ を F と G の最汎単一化作用素とすると，S_k の拡大により，S_{k+1} には $E_i\sigma$ と $E_j\sigma$ が含まれる．E_i，E_j に直接被覆されていた節は，それぞれ $E_i\sigma$，$E_j\sigma$ に直接被覆されるから，$U_{k+1} = U_k \cup \{E_i\sigma, E_j\sigma\} - \{E_i, E_j\}$ となる．また，σ は，F か G に現れる変数を変数以外の記号（定数記号，関数記号）で置き換える代入であるから，$\|E_i\| + \|E_j\| < \|E_i\sigma\| + \|E_j\sigma\|$ である．従って，$\|U_k\| < \|U_{k+1}\|$ である．　(証明終)

[定理 1]　S が充足不能であることと S_k を真とする変種独立割当てが存在しないような有限の p が存在することは同値である．

(証明)

(⇒)

S が充足不能であるとする．Herbrand の定理より，S の節の有限個の基礎例からなる充足不能な節集合 T が存在する．このとき，ある k において，S_k を真とする変種独立割当てが存在しない場合は証明は完了しているので，S_k を真とする変種独立割当て V_k が存在する場合を考える．補題 1 より，V_k に対して S_k 中には純障害物が存在するから，補題 2 より，S_k の拡大により，S_{k+1} に対して $\|U_k\| < \|U_{k+1}\|$ となる．T に含まれる各節における定数，関数記号，述語記号の出現の個数の総和は有限であるから，ある有限の p において $T = U_p$ となる．T は充足不能な S の節の基礎例の集合であるから，U_p を真とする変種独立割当ては存在しない．U_p のすべての要素は S_p の要素であるから，S_p を真とする変種独立割当ては存在しない．

(⇐)

ある有限の p に対して，S_p を真とする変種独立割当てが存在しないとする．S の Herbrand 空間の要素の一つを a とし，節集合 S_p 中のすべての変数に a を代入した節集合を S'_p とする．このとき，S_p 中のすべての変種は S'_p 中ですべて同じ基礎原子式になる．従って，仮定より，S_p を真とする変種独立割当ては存在しないから，S'_p は充足不能である．なぜなら，もし，S'_p が充足可能なら，その真理値の割当てから S_p を真とする変種独立割当てを得ることができるからである．明らかに，S'_p 中のすべての節は S 中の節の基礎例になるから，S'_p には，充足不能な S 中の節の有限個の基礎例が存在し，Herbrand の定理より，S は充足不能である．　(証明終)

この定理から，以下の系は容易に導ける．この系は本手続きの終了条件の一つになっている．

[系 1]　任意の変種独立割当てに対して S_k 中に純障害物が存在しなければ，S は充足可能である．

5. 例　　題

本手続きにおける処理を二つの例題を用いて説明する．例題 1 は，充足不能である節集合に対して，例題 2 は，充足可能な節集合に対して適用した例である．それぞれの例題において，変種独立割当ての求め方，純障害物の発見や解消，などの手続きの詳細については **3.** で述べているので省略し，ここでは全体の流れや問題点について説明する．

[例題 1] 三つの節 C_1, C_2, C_3 からなる節集合 S が以下のように与えられているとする.

$$C_1 = P(x)$$

$$C_2 = \sim P(a) \vee Q(f(y), b)$$

$$C_3 = \sim Q(f(z), b)$$

但し, a, b は定数, x, y, z は変数, f は関数記号, P, Q は述語記号であるとする.

$S_1 = S$ を真とする以下の変種独立割当て V_1 が存在する.

$$C_1 = \underbrace{P(x)}_{True}$$

$$C_2 = \sim \underbrace{P(a)}_{False} \vee \underbrace{Q(f(y), b)}_{False}$$

$$C_3 = \sim \underbrace{Q(f(z), b)}_{False}$$

V_1 に対して, S_1 中には一つの純障害物が存在する. すなわち, $B_1 = \{\langle P(x), P(a)\rangle_{1,2}\}$ である.

図 1 の Step 9 では, 純障害物 $\langle P(x), P(a)\rangle_{1,2}$ の原子式の最汎単一化作用素 $\sigma_1 = \{a/x\}$ を用いて, 以下の節 C_4 を生成する. $C_2\sigma_1 = C_2$ であるから, $C_2\sigma_1$ は生成する必要はない.

$$C_4 = C_1\sigma_1 = P(a)$$

B_1 中のすべての純障害物について処理したので, 新しく生成した節の集合 $\{C_4\}$ と S_1 との和集合を S_2 とする.

第 2 ループにおいて, S_2^p は以下のようになる.

$$S_2^p = \{p_1, \sim p_2 \vee p_3, \sim p_3, p_2\}$$

この S_2^p は充足不能なので, S_2 を真とする変種独立割当ては存在しない. 従って, 定理 1 より S は充足不能である.

[例題 2] 以下の二つの節からなる節集合 S を考える.

$$C_1 = P(f(a))$$

$$C_2 = \sim P(x) \vee P(f(x))$$

但し, x は変数, f は関数記号, P は述語記号である. $S_1 = S$ を真とする以下の変種独立割当て V_1 が存在する.

$$C_1 = \underbrace{P(f(a))}_{True}$$

$$C_2 = \sim \underbrace{P(x)}_{True} \vee \underbrace{P(f(x))}_{True}$$

この V_1 では, すべての原子式が真に割り当てられているため, 純障害物は存在しない. 従って, 系 1 より, S は充足可能である.

しかし, S_1 には以下で示す別の変種独立割当て V_1' も存在する.

$$C_1 = \underbrace{P(f(a))}_{True}$$

$$C_2 = \sim \underbrace{P(x)}_{False} \vee \underbrace{P(f(x))}_{True}$$

この割当てに対しては, 純障害物 $\langle P(f(a)), P(x)\rangle_{1,2}$ が存在する. この純障害物を解消するために節 $C_3 = C_2\{f(a)/x\} = \sim P(f(a)) \vee P(f(f(a)))$ を生成する. このとき, C_1, C_2, C_3 を真とする以下の変種独立割当てに対しても純障害物 $\langle P(f(f(a))), P(x)\rangle_{2,3}$ が存在するため, 処理を続けなければならない. このように, どの変種独立割当てを用いるかによって計算効率が大きく変わる.

$$C_1 = \underbrace{P(f(a))}_{True}$$

$$C_2 = \sim \underbrace{P(x)}_{False} \vee \underbrace{P(f(x))}_{True}$$

$$C_3 = \sim \underbrace{P(f(a))}_{True} \vee \underbrace{P(f(f(a)))}_{True}$$

6. 実験および考察

本手続きの有効性を示すために, いくつかの問題に本手続きを適用した. 本手続き中で変種独立割当てを求めるアルゴリズムとして Davis-Putnam の手続きを用いた[8]. 但し, Davis-Putnam の手続きにおける分割規則では, 最初の節に現れる最初のリテラルを真とすることを優先する戦略をとった.

実験は, Sun SPARC Station IPX (主記憶 32 MByte, 28.5 MIPS) を用いて行い, プログラムは, Sun Common Lisp により開発した. また, 比較のために OTTER での計算時間も併せて示した.

表1 実験結果1

問題	充足可能性	本手続き		OTTER
		Time	Loop	Time
35	*F*	0.01	2	0.04
36	*F*	0.98	4	0.08
37	*F*	0.05	3	0.09
39	*F*	0.01	1	0.06
40	*F*	20.10	8	0.17
41	*F*	0.15	3	0.08
42	*T*	-	-	0.27
43	*F*	4.76	8	1.36
44	*F*	0.03	2	0.12
45	*F*	0.49	3	0.13
46	*F*	0.62	4	0.18

表2 実験結果2

問題	充足可能性	本手続き		OTTER
		Time	Loop	Time
35′	*T*	0.02	1	0.03
36′	*T*	9.41	6	0.07
37′	*T*	0.25	5	191.31
39′	*T*	0.01	1	0.03
40′	*T*	0.01	1	0.03
41′	*T*	0.01	1	0.05
42′	*T*	0.02	2	-
43′	*T*	-	-	0.09
44′	*T*	0.01	1	0.07
45′	*T*	0.23	2	0.10
46′	*T*	0.56	3	0.09

OTTER は，McCune によって開発された導出原理を基礎とした定理証明プログラムであり，さまざまな手法，戦略などがサポートされている著名な定理証明システムの一つである．プログラムは C 言語で書かれ，多種のコンピュータで動作するシステムである．実験では，Ver. 3.0.3 を用い，戦略等はすべての問題においてオート（auto）に設定した[6]．

結果を表 1，表 2 に示す．表中の Time は各問題に対する実行時間（単位 [秒]）を，Loop は変種独立割当てを求めた回数を表す．また，表中の「−」は，30 分経過しても解が得られなかったため，または，節の爆発的増加により計算続行が不能になったために，計算を中止したことを表す．

表 1 において使用した問題は，文献(7)中の等式を含まない節形式の論理式のみからなる問題 35〜46（但し，38 は問題の不備のため除いた）である．これらの問題の充足可能性は，問題 42 のみ充足可能であり，残りの問題はすべて充足不能である．表中では，それぞれ *T* および *F* で示す．表 2 における問題は，表 1 で使用した問題に対して，最後の節を一つ取り除いた節集合を用い，問題番号に「′」を付加して表した．表 2 に示すようにすべての問題の充足可能性は充足可能であった．

この実験の結果から，本手続きは OTTER に比べて問題 40，42，43′ などで極端に計算時間が長くなっている．これらの問題は，他の問題に比べて簡約節が数多く生成する問題である．本手続きは，このような簡約節が多く生成する問題に対しては，計算時間が長くなる傾向にあると考えられる．しかしながら，問題 37′ や 42′ のように，OTTER よりも本手続きの方が極端に計算時間が短くなる問題が存在する．これらの問題は，OTTER で多重の入れ子が現れる節が多数生成されるような問題である．本手続きでは，変種独立割当てによってこのような節の生成を避けることができると考えられる．

7. むすび

本論文で，筆者らは文献(4)で提案した手続きを，純障害物の処理方法を変えることによって拡張し，関数記号を含む任意の節形式の論理式に対して適用可能な定理証明手続きを得た．また，その手続きが完全であることの証明を与えた．更に，著名な定理証明システムの一つである OTTER との比較実験より，本手続きの有効性について検討した．

今後の研究として，本手続きの各ループで求められる変種独立割当てに対し，有効な割当てを用いるための戦略について検討すること，計算実験を通して本手続きが有効となる問題についての特徴を明らかにすること，などが挙げられる．

文　献

(1) Chang C.-L. and Lee R.C.-T.: "Symbolic Logic and Mechanical Theorem Proving", Academic Press (1973).

(2) Robinson J.A.: "A Machine-oriented Logic based on the Resolution Principle", J. Assoc. Comput. Mach., **12**, pp.23–41 (1965).

(3) Jeroslow R.G.: "Computation-Oriented Reductions of Predicate to Propositional Logic", Decision Support Systems, **4**, pp.183–197 (1988).

(4) 山本雅人，大柳俊夫，大内　東："部分例示手法に基づく充足可能性判定手続き"，信学論（A），**J77-A**, 11, pp.1577–1584 (1994-11).

(5) 長尾　真，辻井潤一："コンピュータによる定理の証明"，日本コンピュータ協会 (1983).

(6) McCune W.W.: "OTTER 3.0 Reference Manual and Guide", Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois (1989).

(7) Pelletier F.J.: "Seventy-five Problems for Testing Automatic Theorem Provers", J. Automated Reasoning, **2**, pp.191–216 (1986).

(8) Davis M. and Putnam H.: "A Computing Procedure for Quantification Theory", J. Assoc. Comput. Mach., **7**, pp.201–215 (1960).

山本　雅人

平3北大・工・情報卒．平5同大大学院情報工学専攻修士課程了．現在，同博士後期課程在学中．システム工学，自動推論などに興味をもつ．情報処理学会，人工知能学会各会員．

大柳　俊夫

昭60北大・工・電気卒．工博．現在，札幌医科大学保健医療学部講師．システム工学，遺伝情報処理の研究に従事．情報処理学会，日本OR学会，日本生物物理学会，ACM，SMBE各会員．

大内　　東

昭49北大大学院工学研究科博士課程了．工博．現在，北海道大学工学部情報工学科教授．システム情報工学，応用人工知能システム，医療システムの研究に従事．情報処理学会，人工知能学会，電気学会，計測自動制御学会，日本OR学会，医療情報学会，病院管理学会，IEEE-SMC各会員．