様相節変換に基づくボトムアップ型様相論理証明法

赤 埴 淳 一[†] 井 上 克 巳[††] 長谷川 隆 三[†††]

様相論理の高速な定理証明器の実現方法として，証明すべき様相論理式をモデル生成型定理証明器 MGTP の入力節に変換する様相節変換方式を提案する．本方式は，様相論理の標準的な証明法である様相タブロー法の書換え規則の部分計算に基づいている．本方式の特徴は，書換え規則の適用を遅延するように変換することによって，証明に無関係な分枝の生成を抑制する点にある．本稿では，まず，生成される分枝数を削減するために，閉条件のテストを前件の照合に変換する方式を述べる．さらにゴール情報を利用し，トップダウン推論を模擬するように変換することによって，高速化を図る．様相節変換のコストが入力論理式長に関して線形であることを示し，計算機実験によって本方式の有用性を示す．

Bottom-Up Modal Theorem Proving Based on Modal Clause Transformation

JUN-ICHI AKAHANI,[†] KATSUMI INOUE [††] and RYUZO HASEGAWA [†††]

This paper presents a technique to implement efficient modal theorem provers that transforms modal formulae into input clauses of the model generation theorem prover MGTP. The technique, called the *modal clause transformation method*, is based on partial evaluation of the rewriting rules for the modal tableau method. The modal clause transformation method has the advantages that it can avoid the generation of branches irrelevant to the proof by transforming input clauses so as to delay the invocation of the rewriting rules. To decrease the number of branches, we first propose a transformation method in which close condition testing is replaced with pattern matching in the antecedents of transformed clauses. Then, the method is extended by making use of goal information and by simulating top-down reasoning to reduce the search space. The cost of modal clause transformation is shown to be linear with respect to the length of the input modal formulae. Finally, the results of experiments illustrate the usefulness of the method.

1. はじめに

様相論理は時間や状況を表現・推論可能な論理であり，プログラムの検証・合成[15]や，分散システムの解析[9]，マルチエージェント環境における知識表現[13),19)]など計算機科学や人工知能において様々な応用が提案されている．このような応用では，様相論理式の推論が重要な位置をしめており，高速な定理証明器の実現が重要な研究課題である．これまで，様々な様相論理証明方式が提案されているが，多くは実現原理/技法の提案に留まっており，定量的な評価を試みた例は少ない[4),14)]．また，実際の応用では，与えられた定理の証明とは無関係な論理式を多く含んだ場合に対処しなければならない．そこで本稿では，より実際的な問題への適用を目指し，高速な様相論理証明器の実現をはかる．

様相論理の定理証明法は，直接 (direct) 法と，翻訳 (translation) 法とに大別できる[17]．直接法は，古典論理の証明法を様相論理に拡張したもので，様相タブロー法[7]や様相導出法[1),5)]が提案されている．それに対し，翻訳法は様相論理式を古典論理式に変換する方法[18]であり，様々な様相体系に適用できることが知られている[3),17)]．翻訳法には古典論理の様々な推論戦略が適用可能という利点もあるが，これまで推論制御に関する研究はなされていなかった．

本稿では，様相論理式を翻訳する際に解析を行い，モデル生成型定理証明器 MGTP[8] の入力節集合に変換する様相節変換方式を提案する．本方式は，様相タブロー法を MGTP 上にメタプログラミングする方式[14] に基づいている．文献 14) の方式は，様相タブロー法の書換え規則をスキーマとして MGTP の入力節で

† NTT コミュニケーション科学研究所
NTT Communication Science Laboratories
†† 豊橋技術科学大学情報工学系
Department of Information and Computer Sciences, Toyohashi University of Technology
††† 新世代コンピュータ技術開発機構
Institute for New Generation Computer Technology

記述し，様相タブロー法をMGTP上で模擬するものである．文献14)の方式は直接法に分類できるが，可能世界間の到達可能関係をプログラムすることで様々な様相体系に適用可能であり，翻訳法の利点も合わせもっている．我々は，メタプログラミング方式に部分計算法を適用し，推論制御を組み込んだMGTPの入力節集合を生成する．

一般に，様相タブロー法は多数の分枝を生成するという問題がある．そこで我々は，書換え規則の適用を遅延するように変換することによって，証明に無関係な分枝の生成の抑制を図る．我々はまず，生成される分枝数を削減するために，閉条件のテストを前件の照合に変換する方式について述べる．さらに，Non-Horn Magic Set (NHM) 法[10]を導入し，トップダウン推論を模擬するように変換することによって，高速化を図る．

これらの変換方式では，領域限定（range-restricted）[16]された節集合に変換される．領域限定された節とは，節に現れる変数はすべて前件に現れる節である．領域限定された節集合に対しては，完全な単一化は必要なく，照合で十分なので，効率的な定理証明が可能である．また，本稿の焦点は様相論理式の変換にあるので，最も簡単な様相体系 K を対象とする．

本稿の構成を以下に示す．まず2章で，MGTP上の様相タブロー証明器の概略を述べる．次に3章で様相節変換方式を述べ，4章でNHM法を導入する．さらに5章で，理論的な評価と計算機実験による評価を行い，本方式の有用性を示す．最後に6章で，他の研究との関連について議論する．

2. MGTP上の様相タブロー

本章では，様相論理，MGTPおよび，MGTP上の様相タブロー証明器の概要を述べる．

2.1 様相命題論理

様相命題論理の統語論を次のように定義する．命題記号の集合 Φ，論理結合子 $\neg$ と $\vee$，および様相演算子 $\Box$ が与えられているとする．

1. $p \in \Phi$ が命題記号のとき，p は論理式である．
2. φ と ψ が論理式のとき，$\neg\varphi$ および $\varphi \vee \psi$ は論理式である．
3. φ が論理式のとき，$\Box\varphi$ は論理式である．

論理結合子 $\wedge$ と $\supset$，および様相演算子 $\Diamond$ を通常のように導入する．例えば，$\Diamond\varphi$ は $\neg\Box\neg\varphi$ を表す．

様相命題論理の意味論は通常の可能世界意味論で与えられる．すなわち，モデル M は，可能世界の集合 W，可能世界間の到達可能関係 R，付値関数 V の3つ組 $\langle W, R, V \rangle$ で表される．ここで，付値関数 V は命題記号 $p \in \Phi$ に対し，W の部分集合 $V(p)$ を割り当てるものとする（直観的には，p が真となる世界の集合である）．

1. $p \in \Phi$ のとき，$M, w \models p$ iff $w \in V(p)$.
2. $M, w \models \neg\varphi$ iff $M, w \not\models \varphi$.
3. $M, w \models \varphi \vee \psi$ iff $M, w \models \varphi$ または $M, w \models \psi$.
4. $M, w \models \Box\varphi$ iff すべての wRw' なる w' に対し，$M, w' \models \varphi$.

すべてのモデル M，すべての世界 $w \in W$ に対し，$M, w \models \varphi$ なるとき φ は恒真であるといい，$M, w \not\models \varphi$ なるとき φ は充足不能であるという．

世界間の到達可能関係に制約を設けることにより，様々な様相体系が定義できる．本稿では，到達可能関係に制約を設けないもっとも基本的な様相体系である K を対象とする．

2.2 MGTP

MGTPはボトムアップにモデルを生成する定理証明器である．MGTPの入力節は次の含意形式で与えられる．

$$A_1, \cdots, A_n \rightarrow C_{1,1}, \cdots, C_{1,l_1} | \cdots | C_{m,1}, \cdots, C_{m,l_m}.$$

ここで，$n, m \geq 0$, $l_j \geq 1$ $(j=1, \cdots, m)$ であり，A_i $(i=1, \cdots, n)$ および $C_{j,k}$ $(j=1, \cdots, m; k=1, \cdots, l_j)$ はアトムである．変数は節の前で全称束縛されているものとする．α, β はアトム α と β の連言を表し，$\alpha|\beta$ は選言を表す．$\rightarrow$ は含意であり，$\rightarrow$ の左辺を前件，右辺を後件と呼ぶ．$n=0$ の節を正節と呼び，$true \rightarrow C_{1,1}, \cdots, C_{1,l_1} | \cdots | C_{m,1}, \cdots, C_{m,l_m}$ と表記する．同様に，$m=0$ の節を負節と呼び，$A_1, \cdots, A_n \rightarrow false$ と表記する．それ以外の節を混合節と呼ぶ．

MGTPは，与えられた節集合を充足するアトムの集合（モデル）を求めるために，以下の2つの操作をアトムの集合（モデル候補とよぶ）に繰り返し適用する．モデル候補の初期値は空集合である．

- モデル拡張操作：モデル候補 M に対して，混合節あるいは正節

  $$A_1, \cdots, A_n \rightarrow C_{1,1}, \cdots, C_{1,l_1} | \cdots | C_{m,1}, \cdots, C_{m,l_m}.$$

  と基礎置換 σ が存在して，アトム $A_1\sigma, \cdots, A_n\sigma$ がすべて M によって充足され，かつ，すべての $j=1, \cdots, m$ に対し，M によって充足されないようなアトム $C_{j,k}\sigma (1 \leq k \leq l_j)$ が存在するならば，各 $j=1, \cdots, m$ に対し，$\{C_{j,1}\sigma, \cdots, C_{j,l_j}\sigma\}$ を M に付け加えた m 個のモデル候補を生成する．
- モデル棄却操作：モデル候補 M に対して，負節

  $$A_1, \cdots, A_n \rightarrow false.$$

  と基礎置換 σ が存在して，アトム $A_1\sigma, \cdots, A_n\sigma$ が

すべて M によって充足されるならば，モデル候補 M を棄却する．

上記の規則が適用できなくなった時点で，節集合のすべての極小モデルが得られる．もしすべてのモデル候補が棄却されれば，入力節集合は充足不能であることがわかる．

前件および後件の各アトムの位置に，$\{pred\}$ を付け加えることにより，MGTP 組み込み述語 $pred$ の呼出しを記述することができる．組み込み述語は，前件におかれた場合は前件の充足可能性テストの際に，後件におかれた場合はモデル拡張の際に，それぞれ実行される．

節に現れるすべての変数がその前件に現れるとき，その節は領域限定 (range-restricted)[16] されているという．領域限定された節集合に対して，生成されるモデル候補は，変数を含まないアトムのみから構成される．したがって，上述の規則を適用する際および節の充足性を判定する際に，完全な単一化は必要なく，照合だけで十分である．すなわち，領域限定された節集合に対しては，効率的なモデル生成が可能となる．

2.3 MGTP 上の様相タブロー証明器

様相タブロー法[7] は，様相論理式に真偽値を付与した符号つき整式 (signed formula) を書き換え，反駁法にしたがって定理証明を行う方法である．符号つき整式は，様相論理式 φ と世界 w に対し，真偽値を表す記号 t および f を付与したものである．具体的には，様相論理式 φ と世界 w で真であることを $t(\varphi, w)$ で表し，偽であることを $f(\varphi, w)$ で表す．

越村ら[14] は，タブロー法の書換え規則と MGTP のモデル生成規則との類似性に着目し，様相タブロー法のメタプログラミングによる実現方式を提案している．本方式は，様相タブロー法の書換え規則をスキーマとして MGTP の入力節で記述し，様相タブロー法を MGTP 上でシミュレートするものである．本稿では，本方式をメタプログラミング方式と呼ぶ．

図 1 に，各書換え規則を記述した MGTP の入力節を示す．大文字で始まる記号は変数を表す．述語 $path(W, V)$ は世界 W から世界 V に到達可能であることを表す．$\{new\_world(V)\}$ は新たな世界に対応する記号を生成し，変数 V に代入する MGTP 組み込み述語の呼出しである．すべての節が領域限定されていることに注意されたい．

世界間の到達可能関係が満たすべき性質を入力節としてプログラムすることにより，K 以外の様々な様相体系の定理証明器に拡張可能である．実際，このような方法で，時相論理 PTL[15] の証明器が実現されている[14]．また，述語 $path$ の引数を増やすことにより，複数の様相演算子をもつ体系の証明器に，容易に拡張できる．

- α 変換規則．
  $t(P \wedge Q, W) \rightarrow t(P, W), t(Q, W).$
  $f(P \vee Q, W) \rightarrow f(P, W), f(Q, W).$
  $f(P \supset Q, W) \rightarrow t(P, W), f(Q, W).$
  $t(\neg P, W) \rightarrow f(P, W).$
  $f(\neg P, W) \rightarrow t(P, W).$
- β 変換規則．
  $t(P \vee Q, W) \rightarrow t(P, W) | t(Q, W).$
  $f(P \wedge Q, W) \rightarrow f(P, W) | f(Q, W).$
  $t(P \supset Q, W) \rightarrow f(P, W) | t(Q, W).$
- 閉分枝規則．
  $f(P, W), t(P, W) \rightarrow false.$
- π 変換規則．
  $f(\Box P, W) \rightarrow \{new\_world(V)\}, path(W, V), f(P, V).$
- ν 変換規則．
  $t(\Box P, W), path(W, V) \rightarrow t(P, V).$

図 1 様相タブロー法を実現した MGTP 節
Fig. 1 MGTP clauses used to implement the Modal Tableau Method.

メタプログラミング方式では，証明すべき様相命題論理式 φ が与えられると，図 1 の節集合と節

$$true \rightarrow f(\varphi, w_0).$$

からなる節集合 $\Sigma(\varphi)$ を生成し，MGTP に入力する．様相タブロー法の完全性・停止性[6] から，本方式の完全性・停止性が保証される．すなわち，完全性に関して，以下の補題が成立する．

補題 1 節集合 $\Sigma(\varphi)$ が充足不能のとき，かつそのときに限り，論理式 φ が恒真となる．

3. 様相節変換

本章では，まず様相節を定義し，次に様相節変換による様相論理の定理証明方式の基本的アイデアを述べ，さらに 2 種類の変換方式を提案する．

3.1 様 相 節

様相節変換方式では，様相命題論理式を様相節集合として表現する．様相節の統語論は以下の構成規則で定義される．

1. $p \in \Phi$ が命題記号のとき，p は様相アトム (特に，命題アトムとよぶ) である．
2. $\varphi_1, \cdots, \varphi_n, \psi_1, \cdots, \psi_m$ が様相アトムのとき，$\neg\varphi_1 \vee \cdots \vee \neg\varphi_n \vee \psi_1 \vee \cdots \vee \psi_m$ は様相節である．ただし，$n, m \geq 0$.
3. φ が様相節のとき，$\Box\varphi$ は様相アトムである．

構成規則 2 において $n=0$，$m=1$ の場合からわかるように，様相アトムは様相節となる．構成規則 2 の様相節を $\varphi_1 \wedge \cdots \wedge \varphi_n \supset \psi_1 \vee \cdots \vee \psi_m$ のように表記する．こ

のとき，$\varphi_1 \wedge \cdots \wedge \varphi_n$ を前提部，$\phi_1 \vee \cdots \vee \phi_m$ を結論部と呼ぶ．例えば，$p, q, r \in \Phi$ を命題記号とすると，$\Box p \wedge \Box(p \supset q) \supset \Box q \vee \Box\Box r$ は様相節となる．

様相節集合は，様相節の連言を集合として表現したものである．すなわち，様相節集合は命題論理の連言標準形（節集合）を様相命題論理に拡張したものであり，その表現能力は前章で述べた様相命題論理と同等である．これは，次の命題によって保証される（証明の概要を付録に示す）．

命題 1 任意の様相命題論理式を様相節集合で表現することができる．

3.2 様相節変換方式

様相節変換方式の基本的アイデアは，メタプログラミング方式に部分計算法を適用することである．すなわち，α 規則と β 規則を部分計算して，π 規則および ν 規則が適用される可能性のある様相アトムを調べあげ，各様相アトムに対応した π および ν 規則の部分計算結果を MGTP の入力節として生成する点にある．様相節変換方式 I を図 2 に示す．簡単のため，証明すべき様相論理式として，様相節 $\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m$ が与えられたとしている．

様相節 $\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m$ が与えられた場合，メタプログラミング方式では，次の節が生成される．

$$true \rightarrow f(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m, w_0).$$

図 2 の変換ステップ 1 は，この節の後件に対し，様相タブロー法の α 規則を部分計算することによって得られる．ここで，φ_i および ϕ_j は様相アトム，すなわち，命題アトムあるいは様相論理式である．後者の場合には，π 規則あるいは ν 規則が適用可能である．

1. 前提部の各様相アトム φ_i に対して，以下の節を生成．
   $true \rightarrow t(\varphi_i, \omega_0)$.
   結論部の各様相アトム ϕ_j に対して，以下の節を生成．
   $true \rightarrow f(\phi_j, \omega_0)$.
2. 後件に，$f(\Box\varphi, W)$ あるいは $t(\Box\varphi, W)$ が現れる場合，以下を繰り返す．
   (a) $f(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W)$ が現れる場合，以下の節を生成（π 規則の特殊化）．
   $f(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W)$
   $\rightarrow \{new\_world(V)\}, path(W, V),$
   $t(\varphi_1, V), \cdots, t(\varphi_n, V), f(\phi_1, V), \cdots, f(\phi_m, V).$
   (b) $t(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W)$ が現れる場合，以下の節を生成（ν 規則の特殊化）．
   $t(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W), path(W, V)$
   $\rightarrow f(\varphi_1, V)|\cdots|f(\varphi_n, V)|t(\phi_1, V)|\cdots|t(\phi_m, V).$
3. 閉分枝規則を表すスキーマを生成．
   $t(P, W), f(P, W) \rightarrow false.$

図 2 様相節変換方式 I
Fig. 2 Modal Clause Transformation Method I.

さて，符号つき整式 $f(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m, W)$ が後件に現れたとしよう．この符号つき整式に，π 規則を適用すると，次の節が得られる．

$$f(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W)$$
$$\rightarrow \{new\_world(V)\}, path(W, V),$$
$$f(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m, V).$$

図 2 の変換ステップ 2(a) は，この節の後件に対し，様相タブロー法の α 規則を部分計算したものである．

同様に，符号つき整式 $t(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m, W)$ が後件に現れた場合は，ν 規則を適用して得られた節の後件に β 規則を部分計算することにより，図 2 の変換ステップ 2(b) が得られる．様相節変換方式 I の正当性は，次の補題によって保証される．

補題 2 様相論理式 φ を様相節変換方式 I によって変換した $MGTP$ 入力節集合 $\mathcal{T}_I(\varphi)$ が充足不能であるとき，かつそのときに限り，φ は様相体系 K において恒真となる．

様相節変換方式 I の変換ステップ 2(b) によって変換された節の後件はすべて選言なので，分枝を大量に生成するという問題が生じる．例えば，符号つき整式 $t(\Box(p \wedge q \wedge \Box r \supset s), W)$ に対して，次の節が ν 規則の特殊化として得られる．

$$t(\Box(p \wedge q \wedge \Box r \supset s), W), path(W, V)$$
$$\rightarrow f(p, V)|f(q, V)|f(\Box r, V)|t(s, V).$$

この節は，例えば，$t(\Box(p \wedge q \wedge \Box r \supset s), w_1)$ と $path(w_1, w_2)$ を含むモデル候補に対して適用可能となり，$f(p, w_2)$ と，$f(q, w_2)$, $f(\Box r, w_2)$, $t(s, w_2)$ に対応する 4 つのモデル候補が生成されてしまう．

そこで，閉条件のテストを前件の照合に変換し，生成される分枝数を削減することを考える．具体的には，図 2 の変換ステップ 2(b) を次の 2(b)′ で置き換える．得られた変換方式を様相節変換方式 II とよぶ．

2(b)′ $t(\Box(\Box\varphi_1 \wedge \cdots \wedge \Box\varphi_n \wedge p_1 \wedge \cdots \wedge p_k \supset \phi_1 \vee \cdots \vee \phi_m), W)$ が現れる場合（ただし，$p_1, \cdots, p_k$ は命題アトム），以下の節を生成．

$$t(\Box(\Box\varphi_1 \wedge \cdots \wedge \Box\varphi_n \wedge p_1 \wedge \cdots \wedge p_k \supset \phi_1 \vee \cdots \vee \phi_m),$$
$$W), path(W, V), t(p_1, V), \cdots, t(p_k, V)$$
$$\rightarrow f(\Box\varphi_1, V)|\cdots|f(\Box\varphi_n, V)|t(\phi_1, V)|\cdots|t(\phi_m, V).$$

ここで，$f(\Box\varphi_i, V)$ を前件に移動しなかったのは，この符号つき整式に対し π 規則が適用される可能性があるからである．

例えば，上述の例では，以下の節が得られる．

$$t(\Box(p \wedge q \wedge \Box r \supset s), W),$$
$$path(W, V), t(p, V), t(q, V)$$
$$\rightarrow f(\Box r, V)|t(s, V).$$

このように変換すると，モデル拡張の前に $t(p, w_2)$ と $t(q, w_2)$ の充足可能性が調べられるので，ν 規則の適用が遅延され，無駄な分枝の生成を抑制できる．

この変換の正当性は，次の定理によって保証される(証明は，付録参照)．

定理 1 様相論理式 φ を様相節変換方式IIによって変換した *MGTP* 入力節集合 $\mathcal{T}_{II}(\varphi)$ が充足不能であるとき，かつそのときに限り，φ は様相体系 K において恒真となる．

メタプログラミング方式と同様に，世界間の到達可能関係が満たすべき性質を入力節としてプログラムすることにより，様相節変換方式を様々な様相体系の定理証明器に拡張可能である．

様相節集合 $\{\mathcal{C}_1, \cdots, \mathcal{C}_n\}$ が与えられた場合は，各様相節 $\mathcal{C}_i$ に対応する記号 $\mathscr{C}_i$ を生成し，次の節を生成する．

$$true \rightarrow \mathscr{C}_1|\cdots|\mathscr{C}_n.$$

さらに各様相節 $\mathcal{C}_i$ に対し，図 2 の 1 において *true* の代りに $\mathscr{C}_i$ とした様相節変換を適用する．これは，$true \rightarrow f(\mathcal{C}_1 \wedge \cdots \wedge \mathcal{C}_n, w_0)$ と β 規則を部分計算して得られる．

4. NHM 法による推論制御

実際の様相論理の証明では，証明に直接寄与しない部分論理式が多く含まれる場合が多いため，無駄な推論をしないように制御する必要がある．ボトムアップ型定理証明器に，トップダウン制御を取り入れる方法として Non-Horn Magic Set (NHM) 法[10] が提案されている．NHM 法は，証明すべきアトムをゴールとして表現し，後向き推論を模擬するように節集合を変換するものである．NHM 法には様々な方式があり，また様相節変換方式への様々な導入方法が考えられる．本稿では，ゴールを幅優先で探索する幅優先 NHM 法[10] の考え方を導入した NHM 様相節変換方式を紹介する．様々な NHM 法の導入方法およびその評価，変換の正当性に関しては，別稿で述べる．

幅優先 NHM 法では，入力節

$$A_1, \cdots, A_n \rightarrow C_1|\cdots|C_m.$$

を次の 2 節に変換する．

$$goal(C_1), \cdots, goal(C_m) \rightarrow goal(A_1), \cdots, goal(A_n).$$

$$goal(C_1), \cdots, goal(C_m), A_1, \cdots, A_n \rightarrow C_1|\cdots|C_m.$$

ただし，第一節は $n=0$ のときは生成しない．第一節は，後件のアトムを証明するためには，まず前件のアトムを証明しなければならないというゴール情報の伝播を表す．第二節は，後件のすべてのアトムがゴールとして要求されていなければならないという条件を，元の節に付加したものである．

様相節変換方式IIに，幅優先 NHM 法を単純に適用すると，閉分枝規則 (図 2 の 3) に対して，

$$true \rightarrow goal(t(P, W)), goal(f(P, W)).$$

が得られる．しかし，この節は領域限定性を満足せず，さらに，すべての $t(P, W)$ および $f(P, W)$ に対するゴールを生成してしまうため，ゴール情報を用いた制御の効果が全く得られないという問題が生じる．

そこで，$t(P, W)$ のみを対象とし，ν 規則の特殊化および閉分枝規則のみに幅優先 NHM 法を適用することを考える．このように NHM 法を適応することによって，領域限定性を保証し，ゴール情報を用いた制御が可能となる．

図 3 に NHM 様相節変換方式を示す．図で，$goal(P, W)$ は $goal(t(P, W)$ の略記である．例えば，符号つき整式 $t(\Box(p \wedge q \supset r \vee s), W)$ に対し，図 3 の 2(b)の第二節として，次の節が得られる．

$$goal(r, V), goal(s, V), t(\Box(p \wedge q \supset r \vee s), W),$$
$$path(W, V), t(p, V), t(q, V)$$
$$\rightarrow t(r, V)|t(s, V).$$

この節は，$goal(r, V)$ および $goal(s, V)$ に照合するアトムが存在しなければ，すなわち，後件のアトム $t(r, V)$ および $t(s, V)$ がゴールとして要求されていなければ，ν 規則を適用しないことを意味する．したがって，様相節変換方式IIよりもさらに ν 規則の適用が遅延され，証明に直接寄与しない不要な探索の抑制が可能となる．

1. 前提部および結論部の各様相アトムに対して，様相節変換方式 I と同様の節を生成．
2. 後件部に，$f(\Box\varphi, W)$ あるいは $t(\Box\varphi, W)$ が現れる場合，以下を繰り返す．
   (a) $f(\Box(\varphi_1 \wedge \cdots \wedge \varphi_n \supset \phi_1 \vee \cdots \vee \phi_m), W)$ が現れる場合，様相節変換方式 I と同様の節を生成．
   (b) $t(\Box(\Box\varphi_1 \wedge \cdots \wedge \Box\varphi_n \wedge p_1 \wedge \cdots \wedge p_k \supset \phi_1 \vee \cdots \vee \phi_m), W)$ が現れる場合(ただし，$p_1, \cdots, p_k$ は命題アトム)，以下の 2 節を生成．
   $goal(\phi_1, V), \cdots, goal(\phi_m, V), path(W, V)$
   $\rightarrow goal(\Box(\Box\varphi_1 \wedge \cdots \wedge \Box\varphi_n \wedge p_1 \wedge \cdots \wedge p_k \supset \phi_1 \vee \cdots \vee \phi_m), W),$
   $goal(p_1, V), \cdots, goal(p_k, V).$
   $goal(\phi_1, V), \cdots, goal(\phi_m, V), path(W, V), t(p_1, V), \cdots, t(p_k, V),$
   $t(\Box(\Box\varphi_1 \wedge \cdots \wedge \Box\varphi_n \wedge p_1 \wedge \cdots \wedge p_k \supset \phi_1 \vee \cdots \vee \phi_m), W),$
   $\rightarrow f(\Box\varphi_1, V)|\cdots|f(\Box\varphi_n, V)|t(\phi_1, V)|\cdots|t(\phi_m, V).$
3. 閉分枝規則を表すスキーマを生成．
   $f(P, W) \rightarrow goal(P, W).$
   $t(P, W), f(P, W) \rightarrow false.$

図 3 NHM 様相節変換方式
Fig. 3 NHM Modal Clause Transformation Method.

表 1 計算機による評価結果
Table 1 Evaluation results.

		meta	peval	nhm
例 1	分枝数	5	5	5
	平均証明長	35	21	39
	実行時間	1.0	0.3	0.5
例 2	分枝数	51	2	2
	平均証明長	184	106	210
	実行時間	14.0	3.2	7.0
例 3	分枝数	52	2	2
	平均証明長	187	108	214
	実行時間	16.0	3.5	7.2
例 4	分枝数	136	2	2
	平均証明長	31	13	34
	実行時間	9.8	0.2	0.6
例 5	分枝数	62	256	4
	平均証明長	31	20	30
	実行時間	6.4	8.9	0.4
例 6	分枝数	17900	486	2
	平均証明長	212	116	215
	実行時間	3347.0	41.0	9.0

5. 評　　価

本章では，様相節変換方式の理論的な評価および計算機実験による評価について述べる．

本稿で提案した方式は，様相節変換に要するコストが問題となる．様相節変換に要するコストは，様相論理式の解析に要するコストと入力節の生成に要するコストとに分けられるが，前者は与えられた様相論理式を一回走査するだけなので，与えられた様相論理式の文字列長に線形である．また後者は，様相節変換アルゴリズムからわかるように，与えられた様相論理式に含まれる様相演算子の個数に線形である．したがって，様相節変換に要するコストは与えられた様相論理式の文字列長に線形であることがわかる．

表 1 に計算機実験の結果を示す．表で，meta は 2 章で述べたメタプログラミング方式，peval は 3 章で述べた様相節変換方式II，nhm は 4 章で述べた NHM 様相節変換方式である．分枝数は探索木の幅を表し，平均証明長は探索木の深さを表す．実行時間の単位は秒である．計算機実験にあたっては，Prolog 版の MGTP コンパイラを用い，SparcStation 2 で計測を行った．評価に用いた様相論理式を**図 4** に示す．例 1 および例 5 において，$\Box_a$ などはエージェント a の信念を表す様相演算子である*．

例 1.　三賢人問題
$\Box_a(dot(a) \vee \Box_b \Box_c \neg dot(a)) \wedge$
$\Box_a \Box_b(dot(b) \vee \Box_c \neg dot(b)) \wedge$
$\Box_a \Box_b \Box_c(dot(a) \vee dot(b) \vee dot(c)) \wedge$
$\Box_a \Box_b \neg \Box_c dot(c) \wedge \Box_a \neg \Box_b dot(b) \supset \Box_a dot(a).$

例 2.　前件が長い例
$\Box(p_1 \wedge p_2 \wedge \cdots \wedge p_{50} \supset q \vee r) \wedge$
$\Box p_1 \wedge \Box p_2 \wedge \cdots \wedge \Box p_{50} \supset \Box(q \vee r).$

例 3.　推論チェーンが長い例
$\Box p_1 \wedge \Box(p_1 \supset p_2) \wedge \Box(p_2 \supset p_3) \wedge \cdots \wedge$
$\Box(p_{49} \supset p_{50}) \wedge \Box(p_{50} \supset q \vee r) \supset \Box(q \vee r).$

例 4.　無駄な分枝を生成する例
$\Box(p_1 \wedge p_2 \wedge \cdots \wedge p_5 \supset a_1 \vee b_1 \vee c_1) \wedge$
$\Box(p_6 \wedge p_7 \wedge \cdots \wedge p_{10} \supset a_2 \vee b_2 \vee c_2) \wedge$
$\Box(p_{11} \wedge p_{12} \wedge \cdots \wedge p_{15} \supset a_3 \vee b_3 \vee c_3) \wedge$
$\Box(p_1 \wedge p_2 \supset q \vee r) \wedge \Box p_1 \wedge \Box p_2 \supset \Box(q \vee r).$

例 5.　無駄な世界を生成する例
$\Box_a(\Box_c r \wedge \Box_a r \wedge \Box_e r \supset r_1) \wedge$
$\Box_a(\Box_f s \wedge \Box_g s \wedge \Box_h s \supset s_1) \wedge$
$\Box_a(\Box_i t \wedge \Box_j t \wedge \Box_k t \supset t_1) \wedge$
$\Box_a(\Box_b p \wedge \Box_b q \supset q \vee r) \wedge$
$\Box_a \Box_b p \wedge \Box_a \Box_b q \supset \Box_a(q \vee r).$

例 6.　前件が長くて，無駄な分枝を生成する例
$\Box(p_1 \wedge p_2 \wedge \cdots \wedge p_{50} \supset a_1 \vee b_1 \vee c_1) \wedge \cdots \wedge$
$\Box(p_1 \wedge p_2 \wedge \cdots \wedge p_{50} \supset a_5 \vee b_5 \vee c_5) \wedge$
$\Box(p_1 \wedge p_2 \wedge \cdots \wedge p_{50} \supset q \vee r) \wedge$
$\Box p_1 \wedge \Box p_2 \wedge \cdots \wedge \Box p_{50} \supset \Box(q \vee r).$

図 4 評価に用いた様相論理式
Fig. 4 Sample Modal Formulae.

証明に無関係な部分論理式が含まれない場合（例 1 から例 3）では，メタプログラミング方式（meta）に対する様相節変換方式II（peval）の優位性は明らかである．例 2 および例 3 の評価結果は，入力論理式が長大になった場合に，様相節変換方式IIによって，分枝数を大幅に削減可能であることを示している．様相節変換方式IIにはオーバーヘッドが存在しないので，分枝数削減の効果は，実行時間に見ることができる．それに対し，NHM 様相節変換方式（nhm）では，ゴール情報伝播のオーバーヘッドのため，様相節変換方式IIよりも実行時間を要する．しかし，メタプログラミング方式よりも実行時間は勝っており，このオーバーヘッドよりも分枝数削減の効果が上回っていることがわかる．問題の規模が増大し，入力論理式が長大になるにつれて，削減される分枝数も増大するので，様相節変換の効果が大きくなると考えられる．

一方，証明に無関係な部分論理式が含まれる場合（例 4 から例 6）では，様相節変換方式IIは，メタプログラミング方式に比べて，おおむね優位である．例 4 では，閉条件テストの前件照合への変換による ν 規則の適用遅延の効果を見ることができる．メタプログラミング方式では，入力論理式の前提部のすべての様相アト

* 複数の様相演算子をもつ様相論理体系に対する様相節変換方式は，述語 path に各様相演算子に対応する引数を付け加えることにより定義できる．

ムに対してν規則が適用されるため，無駄な分枝が多数生成されるが，様相節変換方式IIでは，証明に必要最小限の分枝が生成されるだけである．しかし，様相節変換方式IIは，複数の分枝を一度に生成するため，例5のように，メタプログラミング方式よりも悪化する場合がある．それに対し，NHM様相節変換方式の優位性は明らかである．例4に対しては，上述のオーバヘッドの影響が見られるが，他の問題では様相節変換方式IIよりも優れている．例5および例6の分枝数に，ゴール情報による推論制御の効果を見ることができる．実行時間も，分枝数の削減に応じて改善されており，特に例6ではメタプログラミング方式よりも3桁性能が向上している．

一般に，問題の規模が増大し，入力論理式が長大になると，証明に無関係な部分の割合が増大すると考えられる．したがって，実際の大規模な問題に対して，証明に無関係な探索を抑制可能なNHM様相節変換方式の有用性は明らかであると考えられる．

6. おわりに

様相タブロー法の書換え規則を部分計算し，証明すべき様相論理式をMGTPの入力節に変換する様相節変換方式を述べた．本方式では，様相節変換の際に様相論理式を解析するので，無駄な推論を行わないように制御することができることを示した．様相節変換のコストが様相演算子の個数に関して線形であることを示し，さらに計算機実験による評価を行い，本様相論理証明器の有用性を示した．

これまで，様相論理証明の効率的な戦略に関しては，ほとんど報告がなされていなかった．Auffrayらは，様相論理に対する，入力導出や線形導出などの導出戦略を提案している[2]．しかし，これらの戦略は，様相ホーン節という限られた様相論理式のクラスにしか適用できないという問題がある．それに対し，我々の方式は，任意の様相論理式に対して適用可能である．

様相節変換の立場から見ると，従来の翻訳法はπ規則とν規則を完全に部分計算したものと考えることができる．従来の翻訳法と比較すると，様相節変換方式には，次の利点がある．

1. 領域限定された節に翻訳するため，効率的な定理証明が可能となっている．
2. 新たな世界を生成するπ規則，および，新たな世界に情報を伝播するν規則の発火を制御可能である．したがって，証明に無関係な分枝の生成を抑制できる．

本稿では命題の例のみを示したが，領域述語[16]を導入することにより，一階様相論理を扱うことができる．また，本稿では，到達可能関係に性質を設けない様相体系Kを対象とした．本方式は翻訳法の一種なので，到達可能関係の計算手法[17),18]を取り入れることにより，様々な様相論理体系に拡張可能である．しかし，より効率的な定理証明器を実現するためには，各体系に特化した変換方式を検討する余地がある．本方式で用いた定理証明器MGTPは，並列計算機上で実現されている．本稿では，逐次型を対象としたが，様相論理定理証明からいかに並列性を引き出すかが今後の課題である．

様相論理は，人工知能，特に信念の非単調論理や，非標準論理を用いたアブダクションおよび信念翻意の形式化における重要性が認識されてきている．一方，論理プログラミングのデフォルト否定の計算[11]や一階述語論理に基づくアブダクション[12]において，MGTPの有用性が示されている．これらの研究と我々の研究とを合わせることにより，信念の様相論理に基づく非単調推論や仮説推論のより一般的な枠組がMGTP上で扱えることが期待される．

謝辞 本方式の計算機上での実現および実験に際して様々な点でご教示いただいたICOTの越村三幸氏，Prolog版MGTP処理系を提供していただいた三菱電機中央研究所の藤田博氏に感謝します．さらに，本研究にご支援いただいたNTTコミュニケーション科学研究所の西川清史前所長，河岡司所長，中野良平研究グループリーダ，ICOTの内田俊一所長に感謝します．

参考文献

1) Abadi, M. and Manna, Z.: Modal Theorem Proving, *Proc. the 8th Int. Conf. on Automated Deduction*, pp. 172-189 (1986).
2) Auffray, Y., Enjalbert, P. and Hebrard, J.-J.: Strategies for Modal Resolution: Results and Problems, *J. Automated Reasoning*, Vol. 6, pp. 1-38 (1990).
3) Caferra, R. and Demri, S.: Cooperation between Direct Method and Translation Method in Non Classical Logics: Some Results in Propositional S5, *Proc. IJCAI'93*, pp. 74-79 (1993).
4) Catach, L.: TABLEAUX: A General Theorem Prover for Modal Logics, *J. Automated Reasoning*, Vol. 7, pp. 489-510 (1991).
5) Farinas del Cerro, L.: Molog, a System that Extends Prolog with Modal Logic, *New Generation Computing*, Vol. 4, pp. 35-50 (1986).

6) Fitting, M.: *Proof Methods for Modal and Intuitionistic Logic*, Vol. 169 of *Synthese Library*, D. Reidel Publishing Company (1983).
7) Fitting, M.: First-Order Modal Tableaux, *J. Automated Reasoning*, Vol. 4, pp. 191-213 (1988).
8) Fujita, H. and Hasegawa, R.: A Model Generation Theorem Prover in KL1 Using Ramified-Stack Algorithm, *Proc. ICLP'91*, pp. 535-548 (1991).
9) Halpern, J. Y. and Moses, Y.: Knowledge and Common Knowledge in a Distributed Environment, *JACM*, Vol. 37, pp. 549-587 (1990).
10) Hasegawa, R., Ohta, Y. and Inoue, K.: Non-Horn Magic Sets and Their Relation to Relevancy Testing, Technical Report834, ICOT (1993).
11) Inoue, K., Koshimura, M. and Hasegawa, R.: Embedding Negation as Failure into a Model Generation Theorem Prover, *Proc. the 11th Int. Conf. on Automated Deduction*, pp. 400-415 (1992), LNAI 607.
12) Inoue, K., Ohta, Y., Hasegawa, R. and Nakashima, M.: Bottom-up Abduction by Model Generation, *Proc. IJCAI'93*, pp. 102-108 (1993).
13) Konolige, K.: *A Deduction Model of Belief*, Morgan Kaufmann (1986).
14) 越村, 長谷川：モデル生成型証明器上の様相命題タブロ, *Proc. LPC'91*, pp. 43-52 (1991).
15) Kroger, F.: Temporal Logic of Programs, in *EATCS Monographs on Theoretical Computer Science*, pp. 1-148, Springer-Verlag (1987).
16) Manthey, R. and Bry, F.: SATCHMO: A Theorem Prover Implemented in Prolog, *Proc. the 9th Int. Conf. on Automated Deduction*, pp. 415-434 (1988).
17) Nonnengart, A.: First-Order Modal Logic Theorem Proving and Functional Simulation, *Proc. IJCAI'93*, pp. 80-85 (1993).
18) Ohlbach, H. J.: A Resolution Calculus for Modal Logics, *Proc. the 9th Int. Conf. on Automated Deduction*, pp. 500-516 (1988).
19) Shoham, Y.: Agent-Oriented Programming, *Artif. Intell.*, Vol. 60, pp. 51-92 (1993).

付　録

［命題1の証明］　任意の様相命題論理式を様相節の連言で表現可能であることを，構造帰納法を用いて示す．

1. 様相命題論理式が命題記号 p の場合，様相節の構成規則1から p は様相節であり，様相節の連言で表現可能である．
2. 様相命題論理式 φ と ψ が様相節の連言で表現されていると仮定する．φ が $\mathcal{C}_1 \wedge \cdots \wedge \mathcal{C}_n$（ただし，$\mathcal{C}_i$ は様相節）と表現されたとする．
   - 様相命題論理式 $\varphi \vee \psi$ は，様相節の連言の選言なので，命題論理の連言標準形への変換規則を用いて，様相節の連言で表現できる．
   - 様相命題論理式 $\neg\varphi$ は，$\neg\mathcal{C}_1 \vee \cdots \vee \neg\mathcal{C}_n$ と表現できる．
     (a) 様相節 $\mathcal{C}_i$ が様相アトムの場合，様相節の構成規則2から $\neg\mathcal{C}_i$ は様相節である．
     (b) 様相節 $\mathcal{C}_i$ が様相アトムでない場合，すなわち，$\neg\varphi_1 \vee \cdots \vee \neg\varphi_n \vee \psi_1 \vee \cdots \vee \psi_m$（ただし，$\varphi_1, \cdots, \varphi_n, \psi_1, \cdots, \psi_m$ が様相アトム）と表現できる場合，$\neg\mathcal{C}_i$ は様相節の連言 $\varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\psi_1 \wedge \cdots \wedge \neg\psi_m$ で表現できる．

     したがって，$\neg\mathcal{C}_i$ は様相節の連言で表現可能である．$\neg\varphi$ は様相節の連言の選言となるので，命題論理の連言標準形への変換規則を用いて，様相節の連言で表現できる．
   - 様相命題論理式 $\Box\varphi$ は，様相節の連言 $\Box\mathcal{C}_1 \wedge \cdots \wedge \Box\mathcal{C}_n$ で表現できる．

以上から，様相命題論理式の構造帰納法により，任意の様相命題論理式を様相節の連言で表現可能であることが示された．

［定理1の証明］　補題2より，節集合 $\mathcal{T}_{II}(\varphi)$ が充足不能であるとき，かつそのときに限り，節集合 $\mathcal{T}_I(\varphi)$ が充足不能となることを示せばよい．節集合 $\mathcal{T}_I(\varphi)$ において，変換ステップ2(b)が適用された節 $C_1, \cdots, C_M$ を，変換ステップ2(b)′を適用した節 $C'_1, \cdots, C'_M$ に順次置き換えていくことを考える．ここで，M は変換規則2(b)が適用された節の総数である．i 番目の節まで置き換えた節集合を $\mathcal{T}^{(i)}(\varphi)$ と記述すると，この置換によって，次の節集合の列ができる．

$$\mathcal{T}_I(\varphi) = \mathcal{T}^{(0)}(\varphi) \Rightarrow \mathcal{T}^{(1)}(\varphi) \Rightarrow \cdots \Rightarrow \mathcal{T}^{(M)}(\varphi) = \mathcal{T}_{II}(\varphi)$$

$i=1, \cdots, M$ に対して，節集合 $\mathcal{T}^{(i)}(\varphi)$ が充足不能であるとき，かつそのときに限り，節集合 $\mathcal{T}^{(i-1)}(\varphi)$ が充足不能となることを示せばよい．

議論を簡単にするため，変換ステップ2(b)′において $n=k=m=2$ の場合を考える．節集合 $\mathcal{T}^{(i-1)}(\varphi)$ に，変換規則2(b)が適用された節 C_i

$$t(\Box(\Box\varphi_1 \wedge \Box\varphi_2 \wedge p_1 \wedge p_2 \supset \psi_1 \vee \psi_2), W), \; path(W, V) \rightarrow f(\Box\varphi_1, V) | f(\Box\varphi_2, V) | f(p_1, V) | f(p_2, V) | t(\psi_1, V) | t(\psi_2, V).$$

が含まれるとき，節 C_i を節 C'_i

$$t(\Box(\Box\varphi_1 \wedge \Box\varphi_2 \wedge p_1 \wedge p_2 \supset \psi_1 \vee \psi_2), W), \; path(W, V), t(p_1, V), t(p_2, V)$$

$$\to f(\Box\varphi_1, V)|f(\Box\varphi_2, V)|t(\phi_1, V)|t(\phi_2, V).$$

に置換して，節集合 $\mathcal{T}^{(i)}(\varphi)=\{C_i'\}\cup\Sigma$（ただし，$\Sigma=\mathcal{T}^{(i-1)}(\varphi)\backslash\{C_i\}$）を得たとする．

まず，節集合 $\mathcal{T}^{(i)}(\varphi)=\{C_i'\}\cup\Sigma$ が充足不能ならば，節集合 $\mathcal{T}^{(i-1)}(\varphi)=\{C_i\}\cup\Sigma$ も充足不能であることを，$\{C_i'\}\cup\Sigma$ の証明木から $\{C_i\}\cup\Sigma$ の証明木を構成して示す．ここで，証明木の定義は以下のとおりである．

定義 1　節集合 $\mathcal{S}$ の証明木は，以下で定義される木である．

1. 根は，空集合でラベル付けされる．
2. 節点は，モデル候補でラベル付けされる．
3. 節点 N に対して $\mathcal{S}$ 中の節 C が適用され，C の後件に対応して，N がモデル候補 $N_1, \cdots, N_l$ に拡張されるとき，N から $N_1, \cdots, N_l$ に辺を張る．

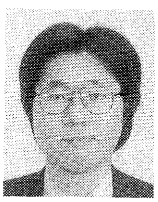
$\{C_i'\}\cup\Sigma$ の証明木を根からたどり，初めて節 C_i' が適用された任意のモデル候補を S とする．適用の際の置換を σ とすると，S は $C_i'\sigma$ の前件の 4 つのアトムを含み，$C_i'\sigma$ の後件に対応する 4 つのモデル候補を子としてもつ．さて，S は Σ のモデル候補でもあるので，$\{C_i\}\cup\Sigma$ のモデル候補となる．節 C_i の前件は節 C_i' の前件の部分列なので，同じ置換 σ によって S に節 C_i も適用可能となる．適用の結果，$C_i\sigma$ の後件に対応して 6 つのモデル候補が生成されるが，$f(p_1, V)\sigma$ と $f(p_2, V)\sigma$ に対応するモデル候補は，S に $t(p_1, V)\sigma$ と $t(p_2, V)\sigma$ が含まれることから棄却される．他の 4 つのモデル候補は C_i' を適用した場合と同一のものである．この操作を $\{C_i'\}\cup\Sigma$ の証明木の葉まで繰り返すと，$\{C_i'\}\cup\Sigma$ の証明木のすべての節点に対応する $\{C_i\}\cup\Sigma$ の証明木の節点が得られる．対応しない $\{C_i\}\cup\Sigma$ の証明木の節点（例えば，上述の $f(p_1, V)\sigma$ に対応するモデル候補）は，すべて棄却された葉である．$\{C_i'\}\cup\Sigma$ の証明木の葉はすべて棄却されるので，$\{C_i\}\cup\Sigma$ が充足不能であることを示す証明木が得られた．

次に，$\{C_i\}\cup\Sigma$ が充足不能ならば，$\{C_i'\}\cup\Sigma$ も充足不能となることを，対偶を用いて示す．$\{C_i'\}\cup\Sigma$ が充足可能として，$\{C_i'\}\cup\Sigma$ の証明木から $\{C_i\}\cup\Sigma$ の証明木を構成する．上述の議論と同様に，$\{C_i'\}\cup\Sigma$ の証明木の葉のすべてに対応する $\{C_i\}\cup\Sigma$ の証明木の節点が構成できる．$\{C_i'\}\cup\Sigma$ が充足可能なので，$\{C_i'\}\cup\Sigma$ の証明木の棄却されていない任意の葉 M は，$\{C_i'\}\cup\Sigma$ のモデルである．M が $\{C_i\}\cup\Sigma$ でも棄却されず，モデルとなることを示せばよい．

1. M に C_i' が置換 σ で既に適用されていた場合，$\{C_i\}\cup\Sigma$ の証明木でも，M に C_i が置換 σ で既に適用されている．したがって，$C_i\sigma$ の後件のアトムのいずれかが M に含まれているため，C_i を用いて M を拡張することは不可能である．また，M は $\{C_i'\}\cup\Sigma$ のモデルなので，Σ に含まれる節によって M を拡張できない．したがって，$\{C_i\}\cup\Sigma$ の証明木でも M は棄却されず，モデルとなる．
2. M に C_i' が適用不可能な場合．
   (a) M に C_i も適用不可能な場合は，M はモデルとなる．
   (b) M に C_i が置換 σ によって適用可能な場合，$C_i\sigma$ の後件に対応して 6 つのモデル候補が生成される．さて，M が $t(p_1, V)\sigma$ と $t(p_2, V)\sigma$ の両方を含むとすると，M に C_i' が適用可能となり，場合わけの条件に反するので，M は $t(p_1, V)\sigma$ と $t(p_2, V)\sigma$ のうち少なくとも 1 つを含まない．以下，$t(p_1, V)\sigma$ を含まないと仮定し，モデル候補 $\{f(p_1, V)\sigma\}\cup M$ が棄却されないことを示す．まず，$\{C_i\}\cup\Sigma$ の中で，$f(p_1, V)\sigma$ に照合可能な前件アトムをもつ節は
   $$t(P, W), f(P, W)\to false.$$
   のみであるが，M は $t(p_1, V)\sigma$ を含まないという仮定より，この節は適用不可能である．また，別の置換 θ で C_i が M に適用可能としても，$t(p_1, V)\sigma$ を導き得ない．よって，$\{f(p_1, V)\sigma\}\cup M$ は棄却されず，モデルとなる．

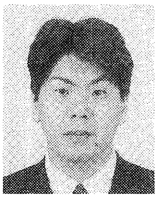
したがって，$\{C_i\}\cup\Sigma$ が充足不能ならば，$\{C_i'\}\cup\Sigma$ も充足不能となる．

（平成 6 年 5 月 16 日受付）
（平成 7 年 2 月 10 日採録）



赤埴　淳一（正会員）

1960 年生．1983 年京都大学工学部数理工学科卒業．1985 年京都大学大学院工学研究科数理工学専攻修士課程修了．同年日本電信電話（株）入社．1989 年～1990 年米国 Stanford 大学客員研究員．現在，NTT コミュニケーション科学研究所主任研究員．探索，信念推論，様相論理の定理証明に関する研究に従事．エージェント指向パラダイムに興味を持つ．AAAI，ソフトウェア科学会会員．



井上　克已（正会員）

1982年京都大学工学部数理工学科卒業．1984年京都大学大学院工学研究科数理工学専攻修士課程修了．同年松下電器産業(株)入社．同社東京情報システム研究所勤務，ならびに（財）新世代コンピュータ技術開発機構への出向を経て，1993年豊橋技術科学大学情報工学系講師．1994年同助教授．京都大学博士(工学)．人工知能，論理プログラミング，言語理論，オートマトン等計算機科学に関する教育研究に従事．人工知能学会会員．

長谷川隆三（正会員）

1949年11月17日生.1972年九州大学工学部通信工学科卒業．1974年九州大学大学院工学研究科通信工学専攻修士課程修了．同年日本電信電話公社入社．同社武蔵野電気通信研究所勤務．1987年（財）新世代コンピュータ技術開発機構へ出向，現在に至る．九州大学博士(工学)．ポリプロセッサシステム，データフローマシン，関数型言語，論理プログラミングおよび定理証明に関する研究に従事．電子情報通信学会会員．