TM-1286

モデル生成型証明系 MGTP の機能拡張と高度問題解決への適用

長谷川　隆三

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191～5

Institute for New Generation Computer Technology

モデル生成型証明系 MGTP の機能拡張と高度問題解決への適用

長谷川 隆三　(新世代コンピュータ技術開発機構研究所)

1　はじめに

本稿では，ICOT で開発を進めてきた一階述語論理の並列定理証明系 MGTP[1] の応用技術について述べる．MGTP は，KL1 が提供する機能や論理プログラミング技術を駆使して，並列推論マシン PIM 上に構築された定理証明系であり，汎用的な推論エンジンと位置付けられる．

MGTP の証明方式は SATCHMO[2] のモデル生成法に基づいており，与えられた節集合に対して前向き推論を行なう．モデル生成法は超導出の一種と見なすことができ，値域限定された節集合に対しては，電子が基底であるので，ユニフィケーションが不要でマッチングで済むという利点を持っている．

現在，基底アトムのみを扱うグランド版 MGTP/G と変数含みのアトムを扱うノングランド版 MGTP/N の二つの版がある．両版とも，PIM/m-256PE 上でほぼ線形に近い台数効果を達成しており，有限代数の未解決問題を含むハードな数学的定理を解くことにも成功している [3]．MGTP は数学的問題の他，データベース，制約充足問題，設計・計画，検証等の様々な AI 的応用に適用することができ，実際に ICOT では法的推論システム HELIC-II の開発に使用されている．

本稿では，AI 的応用に向けて開発を進めてきた MGTP の機能拡張の内，ケース分割の爆発を防ぐためのノンホーン・マジックセット [4] 手法並びに，失敗による否定 [5]，仮説推論 [6] 及び様相推論 [7] を MGTP に組み込む技術を紹介する．

2　モデル生成法

モデル生成法は，与えられた節集合を充足するアトムの集合 (モデル) を構成的に求める証明手法である．

MGTP の入力節は，

$$A_1, A_2, \ldots, A_n \rightarrow C_{1,1}, \ldots, C_{1,k_1}; \ldots; C_{m,1}, \ldots, C_{m,k_m}.$$

の形の含意形式で与えられる．ここで，$A_i(1 \leq i \leq n)$ および $C_{i,j}(1 \leq i \leq m, 1 \leq j \leq k_i)$ はアトムであり，'→' の左辺を前件，右辺を後件という．',' は論理積を，';' は論理和を表す．前件が $true$ である節 ($n = 0$) を正節，後件が $false$ である節 ($m = 0$) を負節，それ以外の節 ($m \neq 0, n \neq 0$) は混合節と呼ぶ．

MGTP はモデル候補集合 $\mathcal{M} = \{\phi\}$ を初期値として，以下の二つの規則を繰り返し適用していく．

- モデル拡張規則: ある置換σの下で，負節でない節の前件が，あるモデル候補 $M \in \mathcal{M}$ で充足され，かつ後件部が充足されない時，Mに各 $\{C_{i,1}\sigma, \ldots, C_{i,k_i}\sigma\}$ を加えて m 個のモデル候補に分岐する．
- モデル棄却規則: ある置換σの下で，負節の前件が，あるモデル候補 $M \in \mathcal{M}$ で充足される時，その Mを棄却する．

上記の規則のいずれも適用できなくなった時点で，節集合の全モデルが $\mathcal{M}$ の要素として得られる．$\mathcal{M} \neq \phi$なら節集合は充足可能であり，$\mathcal{M} = \phi$なら充足不能である．

3　ノンホーン・マジックセット

モデル生成法の動作は，あるモデル候補の下で充足されない節 (違反節と呼ぶ) を検出し，これを充足するようにモデル候補を修正することと捉えられる．しかし，SATCHMO は違反節であればなんでも無原則にモデル拡張の対象として選んでしまうため，必須でない (証明に寄与しない) 非ホーン節が与えられた場合には，節の評価順序によってはモデル候補の爆発を引き起こす危険がある．

これを避けるため，Loveland[8] らは次のような関連性テスト (Relevancy Testing) を提案している．ホーン節の集合を HC，現時点のモデル候補を I としたとき，Prolog 実行によって $HC \cup I$ から $\perp$ (負節の否定) の 証明に失敗した場合，あるいは非ホーン節の前件リテラルの証明に失敗した場合，その探索過程で呼ばれたサブゴールを関連リテラルと呼ぶ．本方法では，違反非ホーン節の内，後件の全てのアトムが関連リテラルのインスタンスでなければ，その非ホーン節はモデル拡張に使用しない．しかし，負節による充足不能性のテスト及び違反非ホーン節の検出の度に Prolog を起動して動的に関連性テストを行なうので，このオーバヘッドはかなり大きいものと思われる．

また，SATCHMO や MGTP のようなボトムアップ型の証明法は，トップダウン型にみられるような重複計算を回避できるが，ゴールに無関係なアトムを生成してしまうという欠点も合わせ持っている．従来，演繹データベースの分野では，両者の利点を融合する方式として，マジックセット法や Alexander 法，およびこれらを一般化した逆メタ解釈法 (Upside Down Meta-Interpretation) が提案されている．これらの方法は，ゴールに関連する基礎アトムのみを外延データベース (EDB) として生成するように内延データベース (IDB) を変換するものであるが，対象が確定節の集合に制限されていた．

ノンホーンマジックセット (NHM) は，逆メタ解釈法を非ホーン節にも適用できるように拡張したものであり，任意の入力節 $(A_1, \ldots, A_n \rightarrow B_1; \ldots; B_m)$ を次のように変換する (幅優先 NHM)．

$$T1: \; goal(B_1), \ldots, goal(B_m) \rightarrow goal(A_1), \ldots, goal(A_n).$$
$$T2: \; goal(B_1), \ldots, goal(B_m), A_1, \ldots, A_n \rightarrow B_1; \ldots; B_m.$$

ここで $goal(A)$ は アトム A がゴールであることを表すメタ述語である．変換 $T1$ はトップダウン実行を模擬しており，入力節の前件 $A_1, \ldots, A_n$ を並列に解く．一方変換 $T2$ は関連性テストを模擬しており，$B_1, \ldots, B_n$が全てゴールになった場合に限り，元の節をモデル拡張の対象にする．ここに示し

た幅優先法と同様な原理で，$A_1, \ldots, A_n$ を逐次的に解いていく深さ優先の NHM が得られる．

上述の NHM 変換法は関連性テストと同等の能力を有しており，負節と関連しない非ホーン節の適用によるモデル候補の爆発を防ぐことができる．また，与えられたプログラムを静的に変換するアプローチをとっているので，Loveland らの動的アプローチに比し，関連性テストに伴うオーバヘッドを軽減することができる．

4 失敗による否定

失敗による否定は論理プログラミングが生み出した重要な技術の一つであり，知識表現に不可欠なツールとなっている．近年，失敗による否定 (not) と古典論理における否定 ($\neg$) の両方を含む一般論理プログラムに対する還言的意味論の研究が進められており，安定モデル意味論が有望視されている．しかし，一般プログラムの安定モデルは不動点において特徴づけられるので，プログラムの評価を局所的に行なうことができず，トップダウンアプローチだけによっては計算できない．そこで，MGTP の枠組を利用してこれをボトムアップに計算する．

ボトムアップに $notP$を計算するためには，Pが現時点でまだ証明されていなくても，それに続く推論で Pが証明されるかもしれないので，$notP$の評価を MGTP がモデルを計算した時点，即ち不動点において行なう．

このため，あるモデル候補で $not\,P$を評価しなければならなくなった場合に，(1)Pが証明されないことを仮定するモデル候補と (2)Pが (いずれ) 証明されなければならないことを要求するモデル候補，の二つに分岐させる，というのが本節で述べる手法のアイディアである．

以下のような not を含む節が与えられるものとする．

$$C \leftarrow A_1, \ldots, A_m, not\,A_{m+1}, \ldots, not\,A_n .$$

これを次のような MGTP 節に変換する．

$$A_1, \ldots, A_m, \rightarrow (\neg\mathrm{K}A_{m+1}, \ldots, \neg\mathrm{K}A_n, C) ; \mathrm{K}A_{m+1} ; \ldots ; \mathrm{K}A_n .$$

$\neg\mathrm{K}P$は Pが成立してはならないことを，$\mathrm{K}P$は Pが成立しなければならないことを，表す様相演算子付きアトムである．それぞれ，上記の (1)(2) に対応している．変換後の節では，$n - m + 1$ のモデル候補に分岐され，最初の分岐では，全ての $A_i (m < i \leq n)$ が証明されなけれないことを仮定している．この場合，変換前の節より Cが成り立つことに注意されたい．残りの分岐は，A_iが証明されることを仮定している枝である．

Pが成立してはならないと仮定していて，Pが証明されたり，さらに Pが成立することを仮定するのは矛盾であるから，

$$\neg\mathrm{K}P, P \rightarrow false.$$

$$\neg\mathrm{K}P, \mathrm{K}P \rightarrow false.$$

というスキーマが必要になる．また，$\mathrm{K}P$である場合，最終的に Pが証明されたかどうか確かめる必要がある．これは，MGTP によって得られたモデル Mに対して「もし $\mathrm{K}A \in M$ ならば $A \in M$」なる制約を設けることにより実現できる．

5 仮説推論

不完全な知識を用いた推論である仮説推論においては，理論と矛盾しない限り仮説を導入し，結論の集合が拡張される．本節では，仮説推論の内，特に，観測事実を説明することのできる仮説集合を計算する過程 (アブダクション) を MGTP 上に実現する手法について述べる．この手法は，前節で述べた失敗による否定と似た，節変換によるものである．

アブダクションは，以下に述べる Eを求めるものとして形式化される．いま，Σを論理式の集合，Γ(仮定可能) をリテラルの集合，G を閉論理式とする．EをΓの基礎例の集合として，もし，

(1) $\Sigma \cup E \models G$ であり，

(2) $\Sigma \cup E$ は無矛盾

であるならば，Eは，(Σ, Γ) からの G の説明であるという．

説明 Eが，条件 (1) を満たすかどうかはモデル生成によって確かめることができる．また条件 (2) については，モデル生成過程で検査することができる．

さて，Σの節に仮定可能な述語 $H_1, \ldots, H_m$ ($H_i \in \Gamma, m \geq 0$) が

$$A_1 \wedge \ldots \wedge A_l \wedge \underbrace{H_1 \wedge \ldots \wedge H_m}_{\text{仮定可能}} \supset C$$

のように含まれていたとしよう．これを以下のような形式の MGTP 節に変換する．

$$A_1, \ldots, A_l \rightarrow (H_1, \ldots, H_m, C) ; \neg\mathrm{K}H_1 ; \ldots ; \neg\mathrm{K}H_m .$$

後件部は，$m + 1$ 通りに分岐するが，その最初の分岐は，$H_1, \ldots, H_m$が証明されたものと仮定して，推論を進めていく枝である．つまり，仮定可能ならば証明なしに成り立つこととするわけである．また，証明されたとものとするので，元の節の結論 Cも成立することになる．残りの分岐は，H_iが証明されないと仮定する枝である．これに対応して，$\neg\mathrm{K}$が付加される．この場合，証明されてしまったら矛盾なので，これを表すスキーマ

$$\neg\mathrm{K}H, H \rightarrow false.$$

を加える必要がある．

以上のような変換の後に，MGTP によって得られたモデルは条件 (2) を満たしており，さらに G を含むものは条件 (1) を満たすことが分かる．

本方式では，構成されるモデル候補の組合せ爆発をさらにどのようにして減少させるかという課題が残されているが，並列性の観点からは有望であると考えられる．また，知識ベースに仮定可能な述語と 失敗による否定の両方が含まれているような場合，比較的容易に失敗による否定と統合できる利点を有する．

6 様相命題タブロー

MGTP は，正節によって初期値，混合節によって計算手続き，負節によって終了条件を与えることができる計算シス

テムと見なすことができる．一方，タブロー法による証明では，証明すべき定理の否定が計算の初期値を与え，拡張規則によって証明が進められ，閉規則によって証明が終了する．本節では，このようなMGTPの計算システムとタブロー法による証明手続きの類似性に着目した，MGTP上に命題様相タブロー証明器を作成する方法について述べる．

様相タブロー法は，定理の否定を満たすようなKripkeモデルが存在しないことを示すことにより，(否定される前の)定理を証明する手続きである．以下に様相システムKのタブロー拡張規則の混合節による表現を示す．$P(F)$は，論理式Fが可能世界Pで真であることを，$P\mathcal{R}Q$は世界Pから世界Qに到達可能であることを示す．また，$\{new(Q)\}$は世界Qを新たに作ることを表すメタ呼び出しである．

$$
\begin{array}{ll}
\alpha & P(\neg\neg F) \rightarrow P(F). \\
 & P(F \wedge G) \rightarrow P(F), P(G). \\
 & P(\neg(F \vee G)) \rightarrow P(\neg F), P(\neg G). \\
 & P(\neg(F \supset G)) \rightarrow P(F), P(\neg G). \\
\beta & P(F \vee G) \rightarrow P(F); P(G). \\
 & P(\neg(F \wedge G)) \rightarrow P(F); P(G). \\
 & P(F \supset G) \rightarrow P(\neg F); P(G). \\
\gamma & P(\Box F), P\mathcal{R}Q \rightarrow Q(F). \\
 & P(\neg\Diamond F), P\mathcal{R}Q \rightarrow Q(\neg F). \\
\pi & P(\neg\Box F) \rightarrow \{new(Q)\}, P\mathcal{R}Q, Q(\neg F). \\
 & P(\Diamond F) \rightarrow \{new(Q)\}, P\mathcal{R}Q, Q(F).
\end{array}
$$

それぞれの規則は，論理結合子の(Kripke的)意味をそのまま表現していると考えられる．α及びβ規則は命題論理記号に関する規則であり，それぞれ，後件が連言及び選言からなる規則である．例えばβの最初の規則は，$F \vee G$が世界Pで真であれば，FがPで真であるか，GがPで真であるかのいずれかであることを表している．γ及びπ規則は，様相記号に関する規則である．ここで，$\Box$は必然性を，$\Diamond$は可能性を表す様相記号である．例えば，πの最初の規則は，Pで$\Box F$が偽であれば，Fが偽であるようなPから到達可能な世界Qが存在することを表している．

閉規則は，矛盾を表す負節

$$P(F), P(\neg F) \rightarrow false.$$

で表現できる．そして，Fが証明したい論理式である場合，その否定が真となるような世界pがあることを意味する正節

$$true \rightarrow p(\neg F).$$

を与えることにより，MGTPは証明を開始する．

本節で述べた手法は，MGTP上にタブロー証明器をメタプログラミングしていると考えることもできる．その柔軟な表現性により，Kの他にも$K4$，T，$S4$や$S5$といった様々な様相論理に対するタブロー証明器を容易に作成することが可能となった．また，メタプログラミングにおいてはその実行効率が問題となるが，解釈オーバヘッドは少なく，種々の実験により性能面でも良好な結果を得ている．

7 おわりに

本稿では，モデル生成型並列定理証明系MGTPの応用面に焦点をあて，トップダウンとボトムアップの融合を図ったノンホーン・マジックセット手法について言及し，次いで高度問題解決に必要とされる失敗による否定，仮説推論，様相推論等の枠組がMGTP上で簡単に実現できることを示した．

その基本的アイデアは，MGTPをメタインタープリタとして用い，これらの枠組に特有な性質(非単調性や様相)を古典論理で扱えるような形式(MGTP入力形式)に変換することである．こうすることによって，これらの特有な性質はモデル候補の生成検査問題に帰着でき，MGTPのケース分割や矛盾したモデル候補の棄却機能によって効果的に取り扱うことができる．

最初にとり上げたノンホーン・マジックセットはMGTPを実用的なものにするための核となる技術であり，選言的データベース，仮説推論，様相推論等さまざまな局面に適用することができる．本手法の本質はゴール(負)情報の伝搬が可能なようにプログラムを静的に変換して探索空間を狭めることにある．この考えは制約充足問題における制約伝搬と関連しており，本手法がかかる問題に適用可能かどうかを明らかにすることが今後の課題である．

参考文献

[1] H. Fujita and R. Hasegawa, A Model Generation Theorem Prover in KL1 Using Ramified-Stack Algorithm, In *Proc. of ICLP'91*, pages 535–548, 1991. ICOT TR-606 1990.

[2] R. Manthey and F. Bry, SATCHMO: a theorem prover implemented in Prolog, In *Proc. of 9th CADE*, Springer Verlag, 1988.

[3] M. Fujita, J. Slaney, and F. Bennett, Automatic Generation of Some Results in Finite Algebra, Proc. of 13th IJCAI に掲載予定, 1993.

[4] R. Hasegawa, Y. Ohta, and K. Inoue, Non-Horn Magic Sets and Their Relation to Relevancy Testing, TR 834, ICOT, 1993. Dagstuhl Seminor on Deduction in Germany.

[5] K. Inoue, M. Koshimura, and R. Hasegawa, Embedding Negation as Failure into a Model Generation Theorem Prover, In *Proc. of 11th CADE*, pages 400–415. Springer Verlag, 1992. LNAI 607.

[6] K. Inoue, Y. Ohta, H. Hasegawa, and M. Nakashima, Bottom-Up Abduction by Model Generation, TR 816, ICOT, 1992. Proc. of 13th IJCAI に掲載予定, 1993.

[7] 越村 三幸，長谷川 隆三，モデル生成型証明器上の様相命題タブロ，In *Proc. of LPC'91*, pages 43–52. ICOT, 1991.

[8] D. W. Loveland, D. W. Reed and D. S. Wilson, SATCHMORE: SATCHMO with RElevancy, CS-1993-06, Duke Univ., 1993.