有限区間制約を付加したモデル生成型定理証明系とその応用

白 井 康 之[†,†††] Reiner Hähnle[††] 長谷川 隆 三[†††]

モデル生成型定理証明システム MGTP（Model Generation Theorem Prover）は，ボトムアップ計算に基づく一階述語論理の定理証明系である．MGTP ならびにその拡張形である CMGTP は，制約充足問題に対しても有効な手続きを提供し，有限代数分野において，未解決問題の証明に成功するなど，高効率な証明系であることが実証されているが，半面，整数によって表された区間などの順序付けられた有限領域に関する制約伝搬においては，その記述の汎用性ゆえに，冗長なモデルを生成することが明らかとなった．本論文においては，こうした問題点を解決するために，全順序有限領域の部分集合をモデル候補として許す新しい定理証明系の枠組みを提案し，その形式的意味論と演繹手続きを明らかにする．また，いくつかの問題に対して適用した実験結果を示し，その有用性を示す．最後に，この新しい枠組みと制約論理プログラミングや多値論理，非単調論理との関連についても言及する．

Model Generation Theorem Proving with Finite Interval Constraints and Its Application

Yasuyuki Shirai,[†,†††] Reiner Hähnle[††] and Ryuzo Hasegawa[†††]

Model Generation Theorem Proving (MGTP) is a class of deduction procedures for first-order logic that were successfully used to solve hard combinatorial problems. For some applications the representation of models in MGTP and its extension CMGTP causes redundancy. Here we suggest to extend members of model candidates in such a way that a predicate p can have not only terms as arguments, but at certain places also subsets of totally ordered finite domains. The ensuing language and deduction system relies on constraints based on finite intervals in totally ordered sets and is called IV-MGTP. We show soundness/completeness of the procedure, and the experimental results that show considerable potential of the method.

1. はじめに

モデル生成型定理証明システム MGTP（Model Generation Theorem Prover）は，ボトムアップ計算に基づく一階述語論理の定理証明系である[6),7)]．Prolog などのトップダウン計算に基づく処理系と比較して，ホーン節に限定されない一階述語論理を対象とすることや，バックトラックなしに完全な証明を得ることができること，また，MGTP が前提とする問題節の値域限定条件により，高速な演算処理を可能としていることなどが主な特長としてあげられる．

MGTP ならびにその拡張形として負リテラルの表現形式を許した CMGTP [12)]は，制約充足問題に対して有効な手続きを提供し，有限代数分野において，未解決問題の証明に成功するなど，高効率な証明系であることが実証されているが[13)]，半面，整数によって表された区間などの順序付けられた有限領域に関する制約伝搬においては，その記述の汎用性ゆえに，冗長なモデルを生成することが明らかとなった．

たとえば，以下のように，整数によって表される有限領域 $\{1,2,3,4,5,6\}$ 上の変数 X，Y に関する制約を考えてみる．

$$1 \leq X, \quad Y \leq 6, \quad X+Y=6$$

このとき，$X \leq 3$ なる制約があるとすると，この制約が Y に伝搬して $3 \leq Y \leq 5$ となることは明らかである．しかしながら，MGTP においては，変数 Y の値がこのような領域上にあることを適切に表現する手段を持たない．$p(Y,N)$ を変数 Y が値 N をとることを表す述語であるとすれば，上記の Y に関する条件は，以下のような選言として表現せざるをえない．

† 株式会社三菱総合研究所情報技術研究部
Mitsubishi Research Institute, Inc.
†† Chalmers University of Technology
††† 九州大学大学院システム情報科学研究院
Graduate School of Information Science and Electrical Engineering, Kyushu University

$$p(Y,3); p(Y,4); p(Y,5) \qquad (1)$$

上記選言の要素はそれぞれ排他的なので，$p(Y,3)$, $p(Y,4)$, $p(Y,5)$ はそれぞれ3つの分岐したモデルとして表現されなければならない．Y が3より大きいときには，いずれも上記の制約を満足することが明らかであるにもかかわらず，このような分岐を生じることは，全体的な証明木を不必要に大きくし，証明を非効率にする．このような表現上の冗長性を回避するために，我々は，以下のようにリテラル表現の拡張を行った．

$$p(t_1,\ldots,t_r,S_1,\ldots,S_m)\ , \qquad (2)$$

ここで，t_i は通常の項を表し，S_i はある有限領域 N_i の部分集合を表す．この拡張は，述語 p の引数として有限領域の部分集合を許容するものである．この表記に従えば，(1) は，$p(Y,\{3,4,5\})$ と表すことができる．以下では，(2) の形式のリテラルを制約アトムと呼ぶ．

制約アトムは，$m=1$ の場合には，多値論理における符号アトム[1)]と同じ意味を持つ（通常，$S_1{:}p(t_1,\ldots,t_r)$ と書かれる）ので，(2) の表現は符号アトムの多次元拡張ともなっている．

有限領域 N_i の表現方法には，さまざまな方法がありうるが，順序性を活かした枝刈り効果を得るために，区間表現を採用することとした．この新しいMGTPの枠組みを以後，IV-MGTP と呼ぶこととする（IV は interval を表す）．

本論文の構成は以下のとおりである．2 章で MGTP について簡単に説明を行う．3 章で IV-MGTP の言語構造ならびに形式的意味論を定義する．IV-MGTP の推論手続きとその健全性，完全性については，4 章で説明を行う．また，5 章で実験の結果を示し，6 章で他の研究との関連，7 章でまとめと今後の課題について述べる．

2. MGTP

2.1 定　　義

MGTP は，直観的には，ルールベースのボトムアップ型推論システムである．MGTP の入力プログラムは，以下のようなルールの集合からなる．ルールは節とも呼ばれる．

$$A_1,\ldots,A_n \rightarrow C_1;\ldots;C_m \ (n \geq 0, m \geq 0)$$

ここで，$A_1,\ldots,A_n$ を前件部，$C_1;\ldots;C_m$ を後件部と呼ぶ．前件部の並びは連言であり，ルールが適用される条件を表す．また，後件部の並びは選言であり，ルールが適用されたときの帰結を表す．特に，$n=0$ の場合を正節と呼び，あらかじめ真であることが分かっているアトムを表す．また，$m=0$ の場合を負節と呼び，アトム間の制約条件を表す．

また，$A_1,\ldots,A_n$，$C_1,\ldots,C_m$ は述語記号を用いて表された単位節（アトム）で，n (≥ 0) 引数の述語記号 p と項 $t_1,\ldots,t_n$ により，$p(t_1,\ldots,t_n)$ と表される．項は，関数記号と定数，ならびに項変数により構成される．項変数を含まない項を基底項と呼び，基底項のみから構成される節を基底節という．上記のルール表現は，以下の一階述語論理表現と等価である．

$$\neg A_1 \vee \ldots \vee \neg A_n \vee C_1 \vee \ldots \vee C_m$$

また，MGTP のルールは，値域限定条件，すなわち，後件部に出現する項変数は，必ず前件部に出現するという条件が満たされなければならない．これゆえ，前件部が空であるような値域限定ルールは，基底でなければならない．

2.2 MGTP の推論手続き

MGTP の証明手続きは，正節から始めて，モデルとなる候補をボトムアップに生成していく過程である．MGTP の推論手続きを**図 1** に示す．

ここで，$\mathcal{M}$ をモデル候補集合と呼び，その要素をモデル候補と呼ぶ．モデル候補は，アトムの集合であり，以下，M と表記する．

モデル候補の拡張は，ルールの前件部が現在保持されているモデル候補 M により充足されるとき，後件部をモデル候補 M に追加することを意味する．後件部が選言である場合には，ケース分割による拡張が行われる．なお，終了したときに *result* が *sat* であれば，与えられたプログラムは充足可能であり，*unsat* であれば，プログラムは充足不能である．MGTP のモデル生成プロセス（データフロー）を**図 2** に示す．

モデル候補の拡張により生成された節（アトムまたは選言）は，モデル拡張候補（以下 D と表記）と呼ばれるバッファに蓄えられる．モデル拡張節 L は，モデル拡張候補から，適当な選択関数（通常は，単位節

(1)　初期状態として $\mathcal{M} := \{\emptyset\}$, $result := unsat$ とする．
(2)　$\mathcal{M} \neq \emptyset$ のとき，$M \in \mathcal{M}$ に対して：
　(a)　モデル候補の棄却
　　プログラム P 中に節，$A_1,\ldots,A_n \rightarrow .$ があり，また，$A_1\sigma,\ldots,A_n\sigma \in M$ を満たすような代入 σ があるとき，
　　$\mathcal{M} := \mathcal{M} - \{M\}$ とする．2 へ．
　(b)　モデル候補の拡張
　　プログラム P に節，
　　$A_1,\ldots,A_n \rightarrow C_1;\ldots;C_m.$ $(m > 0)$ があり，また，$A_1\sigma,\ldots,A_n\sigma \in M, C_1\sigma,\ldots,C_m\sigma \notin M$, なる代入 σ があるとき，$\mathcal{M} := (\mathcal{M} - \{M\}) \cup \{\{M \cup \{C_j\sigma\}\} \mid j = 1,\ldots,m\}$ として 2 へ．
　(c)　上記 (a), (b) がいずれも適用できないとき，M をモデルとして出力し，$result = sat$ とする．$\mathcal{M} := \mathcal{M} - \{M\}$ として，2 へ．
(3)　$\mathcal{M} = \emptyset$ のとき，終了．

図 1　MGTP の基本アルゴリズム
Fig. 1　The algorithm of the MGTP procedure.

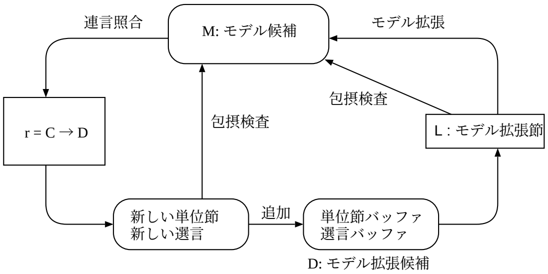

図 2 MGTP のモデル生成プロセス
Fig. 2 Model generation process in MGTP.

優先戦略）によって取りだされたもので，すでにモデル候補に含まれているかどうかのテスト（包摂検査）を行った後に，モデル候補に加えられる．すでにモデル候補に含まれている場合には，それ以降の処理は冗長となるため，新たに追加する必要はない．なお，モデル拡張候補として選言が選択された場合には，選言の要素を分解したケース分割が行われる．

連言照合は，モデル候補に含まれるアトムにより，前件部が充足されるルールを検出するプロセスである．連言照合において，モデル候補が棄却された場合，または，これ以上ルールが適用できないときは，その枝の探索は終了する．

2.3 CMGTP への拡張

MGTP の拡張である CMGTP は，MGTP の表現形式に，負のリテラル表現を許したものである．負リテラルが導出されることにより，相補リテラルによる棄却（unit refutation）や選言の簡約化（unit simplification）が行われ，有限代数分野などにおける制約伝搬問題において，冗長な枝の探索を回避し，制約伝搬を効率的に行うことができる[13]．

3. IV-MGTP の文法と意味論

以下では，IV-MGTP の文法と意味論（IV-MGTP 解釈）を定義する．

3.1 IV-MGTP の言語

IV-MGTP においては，アトムを以下のように拡張する．

$$p(t_1, \ldots, t_r, S_1, \ldots, S_m) \quad (r \geq 0, m \geq 0)$$

ここで，$t_1, \ldots, t_r$ は MGTP における項の定義と同一である．新たに導入された $S_1, \ldots, S_m$ を述語 p の制約部と呼び，各 S_i をそれぞれ制約（または符号）と呼ぶ．また，このような形式のアトムを制約アトムと呼ぶ．以下では，$p(t_1, \ldots, t_r, S_1, \ldots, S_m)$ の項部を $p(t_1, \ldots, t_r)$，また，制約部を $\langle S_1, \ldots, S_m \rangle$ で表記する．$m = 0$ の場合は，MGTP で扱われるアトムと同じである．また，$m = 1$ とし，S_1 を正負の 2 値からなる集合とすれば，CMGTP で扱う正負リテラルを表すことができる．したがって，IV-MGTP の言語は，MGTP および CMGTP の表現を含んでいる．

各 S_i に対しては，それぞれ領域として，プログラム中で定義された有限集合 N_i が対応し，S_i は N_i の部分集合である（$S_i \subseteq N_i$）．

これ以降の議論では，領域 N_i は全順序であることを前提とする☆．したがって，$|N_i| = n$ のとき，N_i は一般性を失うことなく，整数の集合 $\{1, \ldots, n\}$ として表現することができる．

ここで，領域 N を持つ制約 S の表現形式を以下のように与える．

(1) $\{i_1, \ldots, i_r\} : i_j \in N \ (1 \leq j \leq r)$
(2) $]I_1, I_2[\ : \ I_j \ (j = 1, 2)$ は，制約変数．ここで，$]I_1, I_2[\ = \{i \in N \mid i < I_1 \vee i > I_2\}$（この形式の表現を以下 extraval と呼ぶ）．
(3) $[I_1, I_2] : I_j \ (j = 1, 2)$ は，制約変数．ここで，$[I_1, I_2] = \{i \in N \mid I_1 \leq i \leq I_2\}$（この形式の表現を以下 interval と呼ぶ）．
(4) $U : U$ は領域変数

ここで，制約変数は，領域 N 上の値を示す変数であり，領域変数は，N の部分集合を示す変数である．制約の表現としては，当然上記以外にもありうるが，これは実装上の複雑さと有用性とのトレードオフである．

IV-MGTP ルールは，制約アトムによって定義されることを除いて，MGTP のルールと同様に定義される．また，項変数だけでなく，制約変数，領域変数に対しても値域限定条件が満足されなければならない．

制約アトムにおける制約は，それぞれ異なる領域に関連付けられているため，各述語記号に対する制約部の領域がそれぞれプログラム中で定義されている必要がある．すなわち，制約部を持つような述語 p に対して，IV-MGTP プログラムは，以下のような制約宣言を含む必要がある：

$$\text{declare } p(\underbrace{t, \ldots, t}_{r}, j_1, j_2, \ldots, j_m)$$

この宣言は，述語 p は，r 個の項引数と m 個の制約引数を持ち，各制約はそれぞれ，$j_1, \ldots, j_m$ 個の要素を持つ有限全順序集合を定義域とすることを表している．

3.2 IV-MGTP の形式的意味論

IV-MGTP プログラムには，3 つの変数，すなわち

☆ 全順序性は，順序関係に基づく枝刈りを効率的に行うために導入された条件である．なお，次章において展開する意味論，ならびに完全性，健全性に関する議論では，領域 N の有限性のみを仮定した場合でも同様にあてはまることに留意されたい．

項変数，制約変数，領域変数がある．基底代入 σ は，項変数から基底項へ，制約変数から領域 N 上へ，また領域変数から $2^N - \{\emptyset\}$ への関数として定義される．

述語 p が declare $p(t,\ldots,t,j_1,\ldots,j_m)$ のように定義されているとき，p の項部に対して，$\{1,\ldots,j_1\} \times \cdots \times \{1,\ldots,j_m\}$ の要素を対応させる関数を IV-MGTP 解釈 $\mathbf{I}$ と呼ぶ．すなわち，IV-MGTP 基底アトム $L = p(t_1,\ldots,t_r,S_1,\ldots,S_m)$ は，$\mathbf{I}(p(t_1,\ldots,t_r)) \in \langle S_1,\ldots,S_m\rangle$ となるとき，またそのときに限り，IV-MGTP 解釈 $\mathbf{I}$ によって充足される（$\mathbf{I} \models L$）という．IV-MGTP 基底アトムの集合 M は，すべての $L \in M$ に対して，$\mathbf{I} \models L$ が成り立つとき，またそのときに限り充足されるという．

基底 IV-MGTP ルールは，少なくとも前件部の 1 つが $\mathbf{I}$ によって充足されないか，または，後件部の少なくとも 1 つが $\mathbf{I}$ によって充足されるとき，またそのときに限り，$\mathbf{I}$ によって充足されるという．また，IV-MGTP プログラム P は，P に含まれるすべてのルール r に対して，$\mathbf{I} \models r$ が成り立つとき，またそのときに限り，$\mathbf{I}$ によって充足されるという．

4. IV-MGTP の推論手続き

この章では，IV-MGTP において拡張された推論手続きについて説明する．

4.1 モデル候補

MGTP のモデル候補は，基底アトムの集合として表されるが，IV-MGTP においては，これに加えて，述語に含まれる制約，すなわち，領域の部分集合も管理される．このため，MGTP の解釈では，基底アトムの無矛盾かつ極大な集合は，1 つの解釈に対応するのに対し，IV-MGTP においては基底化された制約部は一般に複数の IV-MGTP 解釈を含んでいる．

IV-MGTP におけるモデル候補 M は，declare $p(t,\ldots,t,j_1,\ldots,j_m)$ のように定義された制約アトムの項部の基底インスタンスを $(2^{\{1,\ldots,j_1\}} - \{\emptyset\}) \times \cdots \times (2^{\{1,\ldots,j_m\}} - \{\emptyset\})$ に写像する部分関数と見なすこともできる．したがって，以下では，基底化された項部 $p(t_1,\ldots,t_r)$ に対する制約部が M において $\langle S_1,\ldots,S_m\rangle$ と定義されているとき，$M(p(t_1,\ldots,t_r)) = \langle S_1,\ldots,S_m\rangle$ のように表すこととする．

4.2 連言照合

$r = A \rightarrow C \in P$ を IV-MGTP ルールとし，M をモデル候補とする．このとき，$p(t_1,\ldots,t_r,\kappa_1,\ldots,\kappa_m) \in A$ なる形式のアトムに対して，以下の条件を満足する基底代入 σ が存在するとき，ルール r の前件 A と M に対して，連言照合（conjunctive matching）が可能であるという：

（1） $M(p(t_1,\ldots,t_r)\sigma) = \langle S_1,\ldots,S_m\rangle$;

（2） すべての $i (1 \leq i \leq m)$ に対し：$S_i = \kappa_i\sigma$（κ_i が領域変数である場合），または，$S_i \subseteq \kappa_i\sigma$（それ以外の場合）;

4.3 モデル候補の棄却

モデル候補 M は，ある $L = p(t_1,\ldots,t_r,S_1,\ldots,S_m)$ において，$M(p(t_1,\ldots,t_r)) = \langle S'_1,\ldots,S'_m\rangle$ かつ，ある $i \in \{1,\ldots,m\}$ に対して $S_i \cap S'_i = \emptyset$ となるとき，L によって棄却されるという．明らかに，$\mathbf{I} \models M$ となるような IV-MGTP 解釈 $\mathbf{I}$ において，M が L によって棄却されるなら，$\mathbf{I} \not\models L$ が成り立つ．

4.4 包 摂 検 査

MGTP における包摂検査は，連言照合の結果生成された節を論理的に含むアトムがモデル候補中に含まれるか否かをチェックするプロセスであったが，IV-MGTP においては，制約部に関するテストも行われなければならない．IV-MGTP における包摂検査は，以下のように定義される．

定義 1 あるアトム $L = p(t_1,\ldots,t_r,S'_1,\ldots,S'_m)$ が，モデル候補 M に含まれるとは，$M(p(t_1,\ldots,t_r)) = \langle S_1,\ldots,S_m\rangle$ において，すべての i $(1 \leq i \leq m)$ に対して $S_i \subseteq S'_i$ となることをいう．

定義 2 C を，代入 σ により M で照合可能な IV-MGTP ルールの後件（一般には選言）とすると，$C\sigma$ が M により包摂されるとは，M に含まれるようなアトム L が $C\sigma$ 中に存在することをいう．

例 1 モデル候補 M において，代入 σ により，前件部の照合が可能な IV-MGTP ルールの後件 C を考える．今，C を $p(X,[1,2])$，$\sigma = \{X \leftarrow a\}$ とすると，$C\sigma = p(a,[1,2])$ となる．ここで，M 中にすでに $p(a)$ が存在していて，$M(p(a)) = \langle\{1,2,3\}\rangle$ であったならば，$C\sigma$ は，$M(p(a))$ を更新する制約情報を持つため，M に包摂されない．一方，もし，$M(p(a)) = \langle\{1\}\rangle$ であったならば，$C\sigma$ は，$M(p(a))$ を更新する制約情報を持たないので，M に包摂される．

4.5 モデル候補の管理

例 2 $M(p) = \langle\{2,3\}\rangle$ とし，$D = p([1,2])$ が新たにモデル候補として追加されることを考える．このとき，D は M と矛盾せず，包摂もされないが，M は，$M(p) = \langle\{2\}\rangle$ と書き換えられなければならない．

一般に，モデル候補の更新は，以下のように定義することができる：

$L = p(t_1,\ldots,t_r,S_1,\ldots,S_m)$ をモデル候補 M と無矛盾な制約基底アトムとする．また，$p' = p(t_1,\ldots,t_r)$

を L の基底項部分とする．このとき，L によるモデル候補 M の更新 $M+L$ は以下のように定義される：

$$(M+L)(q)=\begin{cases}\langle S_1,\ldots,S_m\rangle: & q=p' \text{かつ} \\ \quad M(p') \text{が未定義のとき} & \\ \langle S_1\cap S'_1,\ldots,S_m\cap S'_m\rangle: & \\ \quad q=p', & \\ \quad M(p')=\langle S'_1,\ldots,S'_m\rangle & \\ M(q):\text{それ以外のとき} & \end{cases}$$

この定義より，明らかに IV-MGTP 解釈 $\mathbf{I}$，モデル候補 M，制約基底アトム L に対して，$\mathbf{I}\models M$ かつ $\mathbf{I}\models L$ ならば $\mathbf{I}\models M+L$ であることが帰結される．

例 3　例 2 で示した例を MGTP で考えてみる．制約アトム $D=p([1,2])$ は，MGTP の選言 $D'=p(1);p(2)$ に対応し，M は 2 つのモデル候補 $M'=\{p(2)\}$，$M''=\{p(3)\}$ に対応する．したがって，モデル候補の更新としては，4 つの可能性が存在することになるが，このうち，D' は M' に包摂されるため，M' は変化しない．一方，M'' は D' によって棄却される．したがって，M' のみが残ることになるが，この M' は，$M+p([1,2])=p(\{2\})$ によって示される制約情報と一致する．つまり，IV-MGTP のモデル候補更新手続きは，MGTP 上の包摂と棄却を内包するものであるということができる．

4.6　IV-MGTP の手続き

IV-MGTP の基本アルゴリズムを**図 3** に，また IV-MGTP のモデル生成プロセスを**図 4** に示す．

4.7　推論手続きの健全性と完全性

この節では，前章で示した IV-MGTP 手続きの健全性と完全性について言及する．なお，本論文では，補題，定理，系の厳密な証明は省略する☆．

4.7.1　健　全　性

補題 1　充足可能な IV-MGTP プログラム P における IV-MGTP 解釈を $\mathbf{I}$ とする．このとき，$\mathbf{I}\models M$ となるようなモデル候補 M に対して，M が P のルールによって棄却されずかつ拡張可能であるならば，P のルールによって棄却されない $\mathbf{I}\models M'$ なるモデル候補 M の更新 M' が存在する．

証明：背理法による．■

IV-MGTP 手続きに関しては，以下のような健全性が成り立つ：

定理 1（健全性）　IV-MGTP プログラム P が充足可能であるとき，P は IV-MGTP モデルを持つ．

証明：補題より，モデル候補 M の更新列（$M_\emptyset,\ldots,M_\infty$）を構成することができる．更新は単調であり，かつ領域は有限であるから，更新列は明らかに有限回で収束する．このとき，M_∞ は棄却されない IV-MGTP モデルとなる．■

☆ 証明の詳細については，文献 14) を参照のこと．

(1)　初期状態として $\mathcal{M}:=\{\emptyset\}$, $result:=unsat$ とする．
(2)　$\mathcal{M}\neq\emptyset$ のとき，$M\in\mathcal{M}$ に対して：
　(a-1) モデル候補の棄却
　　$p(t_1,\ldots,t_r,S_1,\ldots,S_m)\in M$ に対して，ある i が存在して，$S_i=\emptyset$ となるとき，$\mathcal{M}:=\mathcal{M}-\{M\}$ として 2 へ．
　(a-2) モデル候補の棄却
　　プログラム P 中にルール，$A_1,\ldots,A_n\rightarrow.$ があり，また，$A_1\sigma,\ldots,A_n\sigma\in M$ を満たすような代入 σ があるとき，$\mathcal{M}:=\mathcal{M}-\{M\}$ として 2 へ．
　(b) モデル候補の拡張
　　プログラム P にルール，
　　$A_1,\ldots,A_n\rightarrow C_1;\ldots;C_m.$ $(m>0)$ があり，また，$A_1\sigma,\ldots,A_n\sigma$ が M により連言照合可能で，かつ $C_1\sigma,\ldots,C_m\sigma$ のいずれも M に包摂されないような代入 σ があるとき，
　　$\mathcal{M}:=(\mathcal{M}-\{M\})\cup\{\{M+C_j\sigma\}\mid j=1,\ldots,m\}$ として 2 へ．
　(c) 上記 (a-1),(a-2),(b) がいずれも適用できないとき，M をモデルとして出力し，$result=sat$ とする．$\mathcal{M}:=\mathcal{M}-\{M\}$ として，2 へ．
(3)　$\mathcal{M}=\emptyset$ のとき，終了．

図 3　IV-MGTP の基本アルゴリズム
Fig. 3　The algorithm of the IV-MGTP procedure.

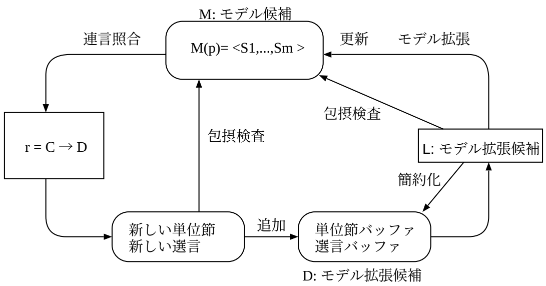

図 4　IV-MGTP のモデル生成プロセス
Fig. 4　Model generation process in IV-MGTP.

4.7.2　完　全　性

完全性に関しては，ここまで用いてきた標準的な意味論に基づけば，IV-MGTP は完全ではない．これは，任意の選択関数に基づく導出法やハイパータブローが完全ではないのと同じ理由である[5]．たとえば，以下のようなプログラムを考えてみる．

$$P=\{\top\rightarrow p,\ \neg q\rightarrow\neg p,\ q\rightarrow\bot\} \qquad (3)$$

このとき，P は明らかに充足不能であるが，前件部または後件部のみから選択する演繹手続きにおいては，充足不能であることを判定できない．たとえば，CMGTP においては，モデル $\{p\}$ を導出する．IV-MGTP においても同様である．

Lu[10]はこのような問題に対し，非標準的な解釈として，拡張解釈と呼ばれる意味論を符号論理プログラ

ミング（SFLP）に対して導入している．本節では，この拡張解釈の考え方を，IV-MGTP に対して拡張した IV-MGTP 拡張解釈を定義し，そのもとでの完全性を示すこととする．

IV-MGTP 拡張解釈の基本的な考え方は，制約部の領域情報を解釈自身に持たせることにある．すなわち，IV-MGTP 解釈が項部から制約部の要素への関数であるのに対し，IV-MGTP 拡張解釈は，項部から制約部の部分領域への（部分）関数となる．これは，前述のモデル候補の定義と同一なものである．以下では，モデル候補として，M，M' の記号を，また，IV-MGTP 拡張解釈として $\hat{\mathbf{I}}$，$\hat{\mathbf{I}}'$ の記号を用いるが，混同の恐れがない限り同一の記号を用いることとする．

IV-MGTP 拡張解釈 $\hat{\mathbf{I}}$ が，IV-MGTP 基底アトム $L{=}p(t_1,\ldots,t_r,S_1,\ldots,S_m)$ を拡張充足する（$\hat{\mathbf{I}} \models L$）とは，$\mathbf{I}(p(t_1,\ldots,t_r))$ が定義されており，かつ，その値を $\langle S'_1,\ldots,S'_m\rangle$ とすると，すべての $1 \le i \le m$ に対して $S'_i \subseteq S_i$ となることをいう．

例 4 (3) の P に対応する IV-MGTP プログラム $P' = \{\top \to p(\{2\}), q(\{1\}) \to p(\{1\}), q(\{2\}) \to \bot\}$ は，充足不能であるが $\hat{\mathbf{I}}(p) = \langle\{2\}\rangle$，$\hat{\mathbf{I}}(q) = \langle\{1,2\}\rangle$ なる $\hat{\mathbf{I}}$ によって拡張充足可能である．

例 5 $\top \to p(\{1\}); p(\{2\})$ なる単一のルールからなるプログラム P_1 と，$\top \to p([1,2])$ なる単一のルールからなるプログラム P_2 を考える．これらは，IV-MGTP 解釈においては，区別することができない．すなわち，ともに，$\mathbf{I}_1 = \{p(1)\}$ と $\mathbf{I}_2 = \{p(2)\}$ によって充足される．一方，$\mathbf{I}_1$ と $\mathbf{I}_2$ に対応する IV-MGTP 拡張解釈 $\hat{\mathbf{I}}_1(p) = \langle\{1\}\rangle$，$\hat{\mathbf{I}}_2(p) = \langle\{2\}\rangle$ は，P_1，P_2 をともに充足するが，P_2 に関しては，さらに別な IV-MGTP 拡張解釈 $\hat{\mathbf{I}}_3(p) = \langle\{1,2\}\rangle$ が存在し，これは，P_1 を拡張充足しない．

IV-MGTP 解釈と IV-MGTP 拡張解釈の関係は，次の補題によって示される．ここで，IV-MGTP 拡張解釈（あるいはモデル候補）$\hat{\mathbf{I}}$ が確定的であるとは，$\hat{\mathbf{I}}(p(t_1,\ldots,t_r)) = \langle S_1,\ldots,S_m\rangle$ のとき $|S_1| = \cdots = |S_m| = 1$ であることをいう．

補題 2 $\mathbf{I}$ を IV-MGTP プログラム P を充足する IV-MGTP 解釈とする．このとき，以下のように定義される IV-MGTP 拡張解釈 $\hat{\mathbf{I}}$ は P を拡張充足する．

$$\hat{\mathbf{I}}(p(t_1,\ldots,t_r)) = \begin{cases} \langle\{i_1\},\ldots,\{i_m\}\rangle & \mathbf{I}(p(t_1,\ldots,t_r)) = \langle i_1,\ldots,i_m\rangle \\ \text{未定義} & \text{上記以外} \end{cases}$$

また，$\hat{\mathbf{I}}$ が P を拡張充足する確定的 IV-MGTP 拡張解釈であるとき，以下のように定義される IV-MGTP 解釈 $\mathbf{I}$ は，P を充足する．

$$\langle t_1,\ldots,t_r,i_1,\ldots,i_m\rangle \in \mathbf{I}(p) \quad \text{iff } \mathbf{I}'(p(t_1,\ldots,t_r)) = \langle\{i_1\},\ldots,\{i_m\}\rangle$$

連言照合や矛盾性のチェックにおいて述べた操作の意味的な性質は，IV-MGTP 拡張解釈に対しても同様に成り立つ．IV-MGTP 拡張解釈に基づき，以下の完全性が成り立つ．

定理 2（IV-MGTP 拡張解釈に基づく完全性） IV-MGTP モデル M を持つ IV-MGTP プログラム P は M により拡張充足可能である．

証明： P に含まれるすべてのルールの基底インスタンスがモデル M により拡張充足されることを示す． ■

また，IV-MGTP が確定的なモデル候補を持つときは，通常の意味での完全性が成り立つ．すなわち，

系 確定的な IV-MGTP モデル M を持つ IV-MGTP プログラム P は，充足可能である．

また，IV-MGTP 拡張解釈においては，以下の健全性も定理 1 と同様に証明される．

定理 3（IV-MGTP 拡張解釈に基づく健全性） IV-MGTP プログラム P が拡張充足可能であるとき，P は IV-MGTP モデルを持つ．

5. 実　験

IV-MGTP の推論手続きは，Java 上で実現されている．以下では，区間表現を用いることによる証明木の縮約とその実行効率を確認するために，チャネルルーティング問題を例題とした行った実験の結果を示す．

5.1 定 式 化

VLSI 設計におけるチャネルルーティング問題は，異なるネットが互いに異なる経路をとるという条件のもとで，ターミナル間の接続要求を満足する制約充足問題として考えることができる．

ここでは，問題を単純化するために，経路はドッグレッグを含まないことを仮定する．この仮定により，問題は，各ネットに対してトラックとレイヤを決定することに帰着する．ここでの制約は，各ネットの経路が互いに異なるということを示す *not equal* の制約と，同一レイヤ上に 2 つの経路が存在する場合の順序関係を定義する *above* の制約の 2 種類がある．たとえば，ネット $N1$，$N2$ に関する *not equal* 制約は，以下のように記述することが可能である：

$$p(N1,[L,L],[T1,T1]), p(N2,[L,L],[T21,T22]),$$
$$neq(N1,N2) \to p(N2,[L,L],]T1,T1[).$$

ここで，述語 p は，2 つの制約領域，すなわち，レイヤを表す制約領域とトラックを表す制約領域を持っ

ている．$neq(N1, N2)$ は，$N1$ と $N2$ が同一レイヤ上で同じトラックを共有しないことを表す制約である．

5.2 実験結果

レイヤ数が 2，トラック数が 3 の領域において，ネット数がそれぞれ 6，8，10，12 個からなる場合について実験を行った結果を**表 1** に示す．

表 1 から，IV-MGTP が証明木を縮約させていることが観察できる．たとえば，6 ネットの場合で，以下のようなモデルが導かれている：

$$\{\, p(1,[1,1],[3,3]), p(2,[1,1],[1,1]),\ p(3,[1,1],[2,2]), p(4,[2,2],[2,3]),\ p(5,[2,2],[1,2]), p(6,[1,1],[2,3]) \,\}$$

これは IV-MGTP においては，1 つのモデルとして導出されるが，CMGTP における 8（$= 1 \times 1 \times 1 \times 2 \times 2 \times 2$）つのモデルを内包しているものである．すなわち，IV-MGTP 表現の利点は，複数の異なるトラックを区間制約を用いて同時に表すことができる点にある．たとえば，ネットの集合を N ($|N| = n$) とし，各ネットがそれぞれ m_i ($i \in N$) 個の CMGTP モデルを持つとすると，$\prod_{i \in N} m_i$ 個の組合せを 1 つの IV-MGTP モデルとして表現することが可能になる．

なお，ここで示したチャネルルーティングのほか，覆面算についても実験を行っている．詳細な結果は省略するが，問題規模に応じて，ほぼ同様の実験結果が得られている．

表 1　チャネルルーティング問題に対する実験結果
Table 1　Experimental results on channel routing problems.

Number of Nets = 6

	IV-MGTP	CMGTP
models	250	840
branches	286	882
runtime(msec)	168	95

Number of Nets = 8

	IV-MGTP	CMGTP
models	1560	10296
branches	1808	10302
runtime(msec)	706	470

Number of Nets = 10

	IV-MGTP	CMGTP
models	4998	51922
branches	6238	52000
runtime(msec)	2311	3882

Number of Nets = 12

	IV-MGTP	CMGTP
models	13482	538056
branches	20092	539982
runtime(msec)	7498	31681

6. 関連研究

- 制約論理プログラミング：IV-MGTP は，有限領域の制約伝搬処理に関して，制約論理プログラミング（CLP）[4),11)]と共通する部分があるが，CLP は，トップダウン計算に基づいているのに対し，IV-MGTP はボトムアップ計算に基づいており，バックトラックなしに証明を行う点，また，それゆえ，同一リテラルの導出において冗長性が排除できる点が大きな利点である．また，CLP はホーン節に限定されているのに対し，IV-MGTP では，非ホーン節も扱うことができるため実応用上の表現能力が高い．
- 符号論理プログラミング（SFLP）：1 章で見たように，IV-MGTP アトムは，1 つの基底アトムに対する制約を複数個許した符号アトムの多次元拡張となっている．また，CLP 同様，SFLP もトップダウンの推論手続きである．
- パラコンシステントロジック：IV-MGTP 拡張解釈においては，領域が空集合であることを許していないが，パラコンシステントロジック[9)]においては，矛盾する情報を表すものとして，空集合を表現することができる．IV-MGTP 拡張解釈，拡張充足性をこれにあわせて再定義し，領域として空集合を含めてモデル化することは可能であり，これによりパラコンシステントロジックを IV-MGTP 上で実現することは容易である．

7. まとめ

本論文では，ボトムアップ型の定理証明システム系において，区間情報で伝搬されるような制約充足問題を解くための新しい枠組みと拡張解釈に基づく意味論的性質を示した．この枠組みは，同時に符号論理や多値論理の多次元拡張ともなっている．また，プロトタイプシステムを用いて，その有用性を検証した．特に一般の述語論理表現では，複数のリテラルによって表現される選言が，単一の表現によって表現されること，また，それによって証明木の縮約が行うことが可能なことを示した．今後の課題としては以下があげられる．

- 制約表現能力の拡張：部分領域の補集合を表す表現形式など，今後の応用領域を考え，制約表現形式の拡張を行う．
- 実行効率の向上：表 1 から観察されるように，枝数の削減に比した実行時間の削減効果が見られていない．これは，モデル候補に対するアトムの所属性検査およびモデル候補の更新処理が，CMGTP

ではフラグ処理で実装されているのに対し，IV-MGTP では，つねに領域計算をともなう処理となっているためである．領域計算に関する実装上の効率化を図ることによって全体効率を向上させることが可能と考えられる．

- 非単調推論への拡張：パラコンシステントロジックや多値論理など，非単調推論への応用を検討する．

参 考 文 献

1) Beckert, B., Hähnle, R. and F. Manyá: The SAT problem of signed CNF formulas, *Labelled Deduction*, Applied Logic Series, Kluwer, Dordrecht (1999).
2) Benhamou, F.: Interval constraint logic programming, *Constraint programming: basics and trends*, Vol.910 of LNCS, Springer-Verlag (1995).
3) Fujita, H. and Hasegawa, R.: A model generation theorem prover in KL1 using a ramified-stack algorithm, *Proc. 8th Intl. Conf. on Logic Programming*, The MIT Press (1991).
4) Gervet, C.: Conjunto: constraint logic programming with finite set domains, *Proc. Intl. Symposium on Logic Programming*, The MIT Press (1994).
5) Hähnle, R.: Tableaux and Related Methods, *Handbook of Automated Reasoning*, The MIT Press (2002).
6) Hasegawa, R. and Fujita, H.: A new implementation technique for a Model-Generation Theorem Provers to solve constraint satisfaction problems, *Research Reports on Information Science and Electrical Engineering*, Vol.4, No.1, Kyushu University (1999).
7) Hasegawa, R., Fujita, H. and Koshimura, M.: MGTP: a model generation theorem prover—its advanced features and applications, *Proc. Intl. Conf. on Automated Reasoning with Analytic Tableaux and Related Methods*, Vol.1227 of LNCS, Springer-Verlag (1997).
8) Hasegawa, R., Koshimura, M. and Fujita, H.: MGTP: A parallel theorem prover based on lazy model generation, *Proc. 11th Intl. Conf. on Automated Deduction*, LNAI 607, Springer-Verlag (1992).
9) Kifer, M. and Lozinskii, E.L.: A logic for reasoning with inconsistency, *Journal of Automated Reasoning*, Vol.9, No.2, pp.179–215 (1992).
10) Lu, J.J.: Logic programming with signs and annotations, *Journal of Logic and Computation*, Vol.6, No.6, pp.755–778 (1996).
11) Montanari, U. and Rossi, F.: Finite Domain Constraint Solving and Constraint Logic Programming, *Constraint Logic Programming: Selected Research*, The MIT Press (1993).
12) Shirai, Y. and Hasegawa, R.: Two approaches for finite-domain constraint satisfaction problem: CP and CMGTP, *Proc. 12th Intl. Conf. Logic Programming*, The MIT Press (1995).
13) 白井康之，長谷川隆三：モデル生成型定理証明システムによる制約充足問題の解決とその並列化，電子情報通信学会論文誌 D-II，Vol.J80-D-II, No.1 (1997).
14) http://ss104.is.kyushu-u.ac.jp/~shirai/iv-mgtp

(平成 13 年 4 月 4 日受付)
(平成 14 年 9 月 5 日採録)

白井　康之

1987 年東京工業大学理学部数学科卒業．1989 年同大学大学院修士課程修了．同年，(株) 三菱総合研究所入社．(財) 新世代コンピュータ技術開発機構出向を経て，現在，同社主任研究員．演繹学習，定理証明に関する研究に従事．ソフトウェア科学会，人工知能学会各会員．

Reiner Hähnle

1992 年 Ph.D.（カールスルーエ大学），1997 年 Habilitation（ウィーン工科大学）．現在，チャルマーズ工科大学教授，Safelogic 社シニアコンサルタント．自動定理証明，ソフトウェア検証，非古典論理に関する研究に従事．Gesellschaft für Informatik（GI），IEEE 各会員．

長谷川隆三（正会員）

1972 年九州大学工学部通信工学科卒業．1974 年同大学大学院修士課程修了．日本電信電話公社，(財) 新世代コンピュータ技術開発機構出向を経て，現在九州大学教授．工学博士．論理プログラミング，定理証明に関する研究に従事．電子情報通信学会，人工知能学会各会員．