ICOT Technical Memorandum: TM-1271

TM-1271

PIM 上の並列定理証明系 MGTP

長谷川　隆三、越村　三幸、
藤田　博（三菱）、藤田　正幸（MRI）

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

PIM 上の並列定理証明系 MGTP

MGTP: A Parallel Model Generation Theorem Prover on PIM

長谷川 隆三 Ryuzo HASEGAWA 越村 二幸 Miyuki KOSHIMURA
新世代コンピュータ技術開発機構 ICOT

藤田 博 Hiroshi FUJITA
三菱電機中央研究所 MELCO

藤田 正幸 Masayuki FUJITA
三菱総合研究所 MRI

We have implemented a model-generation based parallel theorem prover called MGTP in KL1 on a parallel inference machine PIM. We have developed several techniques to improve the efficiency of forward reasoning theorem provers. These include avoiding redundancy in conjunctive matching, suppressing over-generation of models and unnecessary case splitting, compiling input clauses into KL1 clauses, and parallelization on a distributed memory architecture. With MGTP, we have achieved almost linear speedup on a PIM/m consisting of 256 PEs and succeeded in solving hard mathematical problems in which an open problem in finite algebra is included. MGTP can cover wide AI application areas and is actually used to develop a legal reasoning system HELIC-II in ICOT.

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 について述べる．本研究の狙いは，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 上に高速かつ柔軟性に富む汎用的な推論エンジンを構築することである．

MGTP の証明方式は SATCHMO[1] のモデル生成法を基本としており，値域限定された節集合に対しては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．さらに，補題化，包摂テスト，削除戦略等の探索空間を狭める技法を導入することが容易である．

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 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[2]．MGTP は数学的問題の他，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．

MGTP の研究過程で種々の効率改善技法や機能拡張技術が開発された．これらは，1)SATCHMO に含まれていた連言照合の冗長性を除去する方法[3]，ケース分割の爆発を防ぐ手法[4]，アトムの過剰生成を抑止して計算量・記憶量を削減する手法[5]，入力節及び証明手続きの KL1 へのコンパイル技法や，2) 失敗による否定や仮説推論[6][7]，様相推論を MGTP に組み込む技術[8]，そして 3) 分散メモリアーキテクチャ上での並列実装技術[9] などである．

本稿では，この内 3) の MGTP の並列化方式とその実現手法に焦点をあて，MGTP の最近の話題を紹介する．

2 MGTP におけるモデル生成法

MGTP の入力節は次のような含意形式で与えられる．

$$A_1, A_2, \ldots, A_n \rightarrow C_1 : C_2 : \ldots : C_m \quad (n, m \geq 0)$$

ここで，$A_i (1 \leq i \leq n)$ および $C_j (1 \leq j \leq m)$ 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 $(n = 0)$ を正節と呼び，後件が $false$ である節 $(m = 0)$ を負節と呼ぶ．それ以外の節 $(m \neq 0, n \neq 0)$ は混合節と呼ばれる．以下では，混合節及び負節をそれぞれ，generator 節，tester 節とも呼ぶ．

モデル生成法は，以下の二つの規則を繰り返し適用し，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．規則中 $\mathcal{M}$ は構成途中のモデルの集合を表し，各要素をモデル候補と呼ぶ．$\mathcal{M}$ の初期値は $\{\phi\}$ である．

- モデル拡張規則: ある置換σの下で，混合節の前件が，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に Mに各 $C_j\sigma$ 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 $M \in \mathcal{M}$ で充足される時にその 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．

(0) $M := \phi$;
(1) $D := \{A \mid (true \rightarrow A) \in$ a set of given clauses$\}$;
(2) **while** $D \neq \phi$ **do begin**
(3) $D := D \setminus \{\Delta\}$;
(4) $new := CJM_{generator}(\{\Delta\}, M)$;
(5) $M := M \cup \{\Delta\}$;
(6) $new' := subsumption(new, M \cup D)$;
(7) **if** $CJM_{tester}(new', M \cup D) \ni false$
then return(unsat);
(8) $D := D \cup new'$;
(9) **end return**(sat)

図 1: MGTP のアルゴリズム

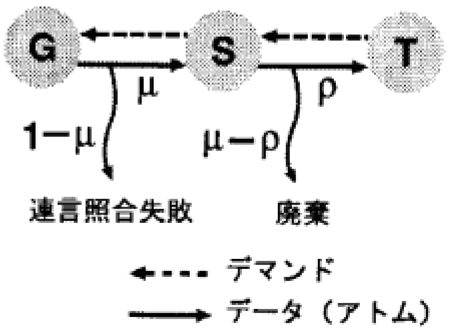

図 2: 遅延モデル生成

3 MGTP の処理手続き

前述のモデル生成法を実現する MGTP の処理手続きを図 1に示す．但し，分岐に対する手続きは含まれていない．ここで，Mはモデル候補を，Dはモデル拡張候補 (モデル拡張規則の適用の結果としてMに付加されるべきアトムの集合) を表す．

$CJM_{Cs}(\{\Delta\}, M)$ は，節集合 Cs の各節に対して，前件が $\{\Delta\} \cup M$で充足されるか否かを調べ，充足された場合その後件を出力する操作である．この操作では，前件アトムと $\{\Delta\} \cup M$の各要素の間の照合 (これを連言照合と呼ぶ) が必要になる．例えば前件が A_1, A_2 なる節の場合，A_1にΔを，そして A_2に Mの各要素を照合することを$\Delta \times M$と表すと、$\Delta \times M$，$M \times \Delta$，$\Delta \times \Delta$の照合が必要になる．

$subsumption(new, M \cup D)$ は，生成されたアトム集合newが $M \cup D$に包摂されるか否かを調べ，包摂されなかったアトムを出力する操作である．

MGTP は上記の手続きに対応して，(4) の generator 節を用いて新たなアトム集合を生成する「生成 (G) プロセス」，(6) の包摂テストを行なう「包摂テスト (S) プロセス」，(7) の tester 節を用いてモデル候補の棄却テストを行なう「棄却テスト (T) プロセス」，から構成される．

MGTP の計算機構は基本的に generate-and-test 型であるが，G プロセスの走り過ぎによるアトムの過剰生成の危険性があり，計算量の爆発と共にメモリ空間の爆発をも引き起こす可能性がある．この危険を回避するため我々は，「T プロセスが必要とする時のみ G プロセスを起動しアトムを生成する」という要求駆動的考え方に基づいた，遅延モデル生成法を開発した[5]．

図 2に遅延モデル生成法におけるデータ及び要求の流れを示す．図中のμ, ρの意味については，表 1を参照されたい．G プロセスで g通りの連言照合が行なわれたとすると，その内μg個のアトムが照合の成功により生成され，S プロセスに送られる．そして，包摂テストの結果 $(\mu - \rho)g$個のアトムが廃棄され，結局ρg個のアトムが棄却テストされる．

表 1: 計算量とメモリ量の解析

(前件リテラル数は混合節が 2 で負節が 1 の場合)

アルゴリズム	T 棄却	S 包摂	G 生成	M メモリ
基本	ρm^2	$\mu\rho^2 m^{4\alpha}$	$\rho^2 m^4$	$\rho^3 m^4$
遅延	ρm^2	$\mu m^{2\alpha}$	m^2	ρm^2
遅延+先読み	m^2	$(\mu/\rho) m^{\alpha}$	m/ρ	m

m : モデル候補 Mの要素数　μ : 連言照合の成功率
ρ : 生成アトムの生存率 $(0 \leq \rho \leq \mu \leq 1)$
α : 包摂テストの効率指数 $(1 \leq \alpha \leq 2)$

遅延モデル生成法では，アトムの過剰生成による無駄をなくすことができ，表 1に示すように各プロセス (S/G) の計算量及びメモリ消費量 (M) のオーダを，前件リテラル数 n に対して n 乗根の割合で大幅に削減することができる．ここで遅延+先読みは，より広い空間について棄却テストを行なう最適化を施したものである．

4 並列化

MGTP には，(1) ケース分割 (複数のモデル候補の探索)，(2) 連言照合，(3) 包摂テストの三種の並列性の源がある．ノンホーン節を含む問題の場合には，ケース分割による OR 並列性のみで十分な並列効果が得られる．しかし，ホーン節のみからなる問題の場合はモデル候補は一つしかないので，(2)(3) の AND 並列性を引き出さなければならない．

4.1 OR 並列化

OR 並列性は分岐が生じるたびに各分枝の探索を異なる PE で行なうことにより抽出されるが，実際には親元の環境 (モデル候補) のコピーが必要であり，また一般には分枝数は爆発的に増加するので PE 間通信が問題となる．そこで，利用可能な PE を埋めるのに充分な分枝が得られた後は，他 PE にそれ以降の分枝を割り付けることなしに，自 PE 内で処理を続ける方法 (有界 OR 並列) を採っている．

分枝の PE への割り当て方法としては，ケース分割が

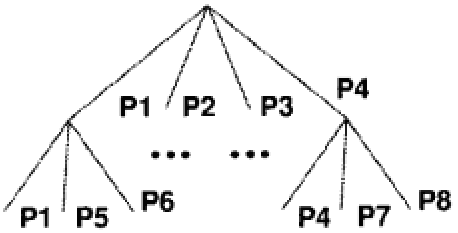

図 3: 縦型割り付け

行なわれた時点で，その中の一つの分枝を自 PE 内で解き，他の分枝は他 PE へ投げる縦型割り付け法 (図 3) を採用した．これは，深さ優先探索と幅優先探索の一種の混合である．さらに，分枝を渡す先の PE の決定法としては，(1) 各 PE で乱数によって確率的に決める方法や，マスタ PE を介し，(2) 割り当て PE を modulo 計算により巡回的に決める，(3) 各 PE の負荷を監視し軽負荷の PE を通知する，(4) アイドル状態の PE を管理し，その一つを通知する，等を検討した．次節で述べる Bennett 問題の実験によると，(2) 及び (3) が比較的良好な結果を示している．

4.2 AND 並列化

AND 並列化にあたり，(1)PE 数によらず証明が変わらない証明不変と PE 数に応じて変わる証明変動，(2) モデル共有 (分散メモリアーキテクチャでは各 PE が同じモデル候補をコピーして持つ) とモデル分散，(3) マスターありとマスターなし，の三項目について比較検討し，証明不変，モデル共有，マスター・スレーブを基本方式とした．ここで証明不変を採用した理由は，速度向上が並列化によるものか，あるいは戦略 (探索空間の変更) によるものかを，明確に区別したかったからである．

本 AND 並列化方式では，図 4のように，中央のマスター (M) プロセスを介して上部の G プロセス群と下部の T プロセス群が連結している．G プロセスは要求駆動的に，T プロセスはデータ駆動的に動作する．

本方式には，証明不変及び包摂テストに起因する二つの逐次性がある．前者は，生成されたアトムのとりだし順序を固定化するために起こるもので M プロセスの serializer を merger にするとこの問題は無くなる．ただし，こうすると証明は変動する．後者は，原理的な問題でありここでは，包摂テストを局所的包摂テスト (LS) と大域的包摂テスト (GS) に分離して，逐次性の低減を図った．生成されたアトムに対する包摂テストは，それ以前に生成されたアトム集合 A に対して行なわれなければならない．LS では，これを既に包摂テスト済みのアトム集合 $M \cup D$に対して行ない，GS ではその残り $A \setminus (M \cup D)$ に対して行なう．現方式はモデル共有型であるので，各生成アトム New_jに対する LS は，他に影響を与えることもなく，各 PE 内の G プロセスでそれぞれ独立に行なうことができる．GS には依

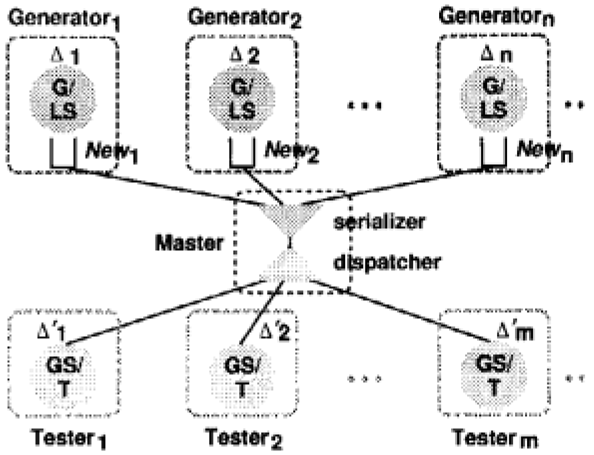

図 4: AND 並列化方式の概要

然逐次性があるので，これを T プロセス内で行なうことにし，各 T プロセス間のパイプライン実行により並列性を抽出している．

M プロセスの基本的な仕事は，G プロセスが生成した LS 済みのアトムを T プロセスに分配することである．各スレーブプロセスは，暇になると次の仕事をもらうために M プロセスに要求を出す．M プロセスは，1) 要求に先着順に応答し，2) 各スレーブプロセスが同じ仕事をしないように仕事を排他的に割り当て，3)G プロセスの生成したアトム数と T プロセスに分配したアトム数の差 (生成したが T プロセス によって消費されていないアトム数) を一定に保つことにより，G プロセスの先行し過ぎによるアトムの過剰生成を抑制する．1) により自然に動的負化分散も行なわれる．G 及び T プロセスの処理は前節で述べた通りであり，配布されたアトム一つに対して連言照合を行なう．

要求一回あたりの M プロセスの仕事量は，基本的には生成アトムΔの配分しか行なわないので，非常に軽い．包摂テストを LS と GS に分けることは，その逐次性を低減する他に，LS において生成アトムのふるい落としが行なわれることによる PE 間通信の低減も計れる利点を持つ．

5 性能評価

5.1 OR 並列性能

図 5に pigeon hole 問題と Bennett 問題[2] の OR 並列性能を示す．前者は典型的な定理証明のベンチマークであり，後者は有限準群に関する未解決問題 (ラテン方陣の数え上げ) である．いずれも生成されるアトムは基底アトムのみであるが，OR 分岐の数が組合せ的に増大する問題である．

証明図の形状は，pigeon hole の場合良く均衡しており，Bennett では均衡の度合が悪い．したがって Bennett は pigeon hole に比べ負荷の均等割り付けが困難であるが，それでも 170 倍程度の良好な速度向上が得られている．

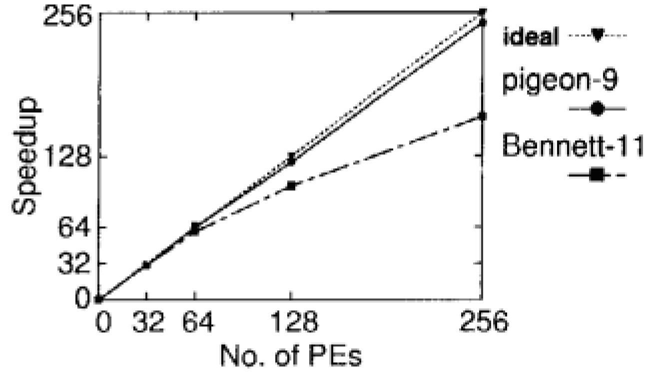

図 5: PIM/m-256PE 上での OR 並列性能

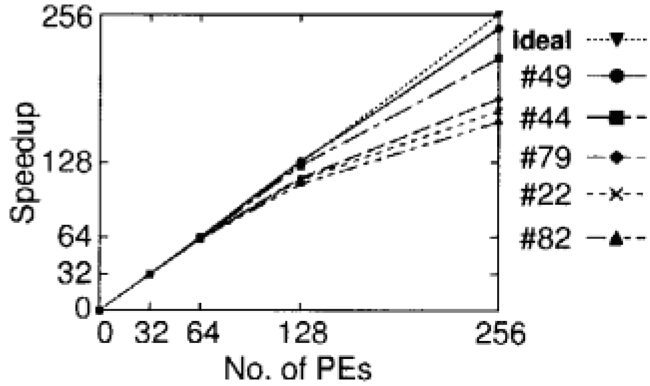

図 6: PIM/m-256PE 上での AND 並列性能

5.2 AND 並列性能

図6は，condensed detachment 問題 (分離の規則に関する問題) [10]に対する AND 並列性能を示したものである．

condensed detachment 問題は，米国 ANL が挑戦中の数学的定理であり，いくつかの単位正節 $true \rightarrow p(A)$，三段論法を表す混合節 $p(X), p(i(X,Y)) \rightarrow p(Y)$ と一つの単位負節 $p(G) \rightarrow false$ からなる．この問題はホーン節のみからなるので，枝分かれによる OR 並列性は全くない．

証明に要する時間 (32PE 上) は，#49:18600 秒，#44:9700 秒，#22:8600 秒，#79:2500 秒，#82:1900 秒である．また，$|M \cup D|$ の大きさは，#49:20600，#44:15100，#22:36500，#79:14200，#82:15100 である．台数効果をみると，#44，#49 では 230 倍以上，#82，#22，#79 では 170 から 180 倍程度の速度向上が得られている．

#44，#49 の台数効果が他の問題に比べて優れていることから，時間を要する問題ほど，また $|M \cup D|$ の成長速度が遅い問題ほど良好な台数効果が得られていることが分かる．$|M \cup D|$ の成長速度が速いと，並列性能が劣化する原因としては，大域的包摂テストの量が増加するためこの逐次性が顕著になるためと考えられる．

6 おわりに

ICOT で開発した並列定理証明系 MGTP について，並列化方式とその実現手法を中心に紹介した．本研究では，当初より世界最高速をスローガンに掲げてきたが，PIM 上で「線形に近い台数効果」が得られ，性能面に関しては概ねこの目標は達成した．主に，分離の規則に関する問題や準群問題のような数学的定理証明の問題に取り組んできたが，未解決問題を含めこれまで逐次マシンでは不可能であった問題を解くことに成功するなど，大規模並列定理証明系の優位性を示し得たと考える．

しかし，並列化方式の観点からは，解決すべき課題もいくつか浮き彫りにされてきている．現在の AND 並列化方式は 256PE 上で 200 倍以上の並列性能を示しているが，包摂テストの逐次性の問題は完全には解決しておらず，スケーラビリティの点で限界がある．項の重さによるソーティング等の機能を組み込む予定であるが，同種の問題が予想される．今後，これらの問題の解決を図ると共に，AI 応用向けに MGTP の機能拡張を進め，MGTP の特長を生かした応用領域を開拓していく予定である．

◇ 参 考 文 献 ◇

(1) R. Manthey and F. Bry. SATCHMO: a theorem prover implemented in Prolog. In *Proc. of CADE'87*. Springer Verlag, 1987.

(2) M. Fujita, J. Slaney, and F. Bennett. Automatic Generation of Some Results in Finite Algebra. Proc. of IJCAI'93 に掲載予定, 1993.

(3) 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.

(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) R. Hasegawa, M. Koshimura, and H. Fujita. Lazy Model Generation for Improving the Efficiency of Forward Reasoning Theorem Provers. In *Proc. of IWAR'92*, pages 191–202, 1992. ICOT TR-751.

(6) K. Inoue, M. Koshimura, and R. Hasegawa. Embedding Negation as Failure into a Model Generation Theorem Prover. In *Proc. of CADE'92*, pages 400–415. Springer Verlag, 1992. LNAI 607.

(7) K. Inoue, Y. Ohta, H. Hasegawa, and M. Nakashima. Bottom-Up Abduction by Model Generation. TR 816, ICOT, 1992. Proc. of IJCAI'93 に掲載予定.

(8) 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.

(9) 長谷川 隆三，越村 三幸. モデル生成型証明器 MGTP の AND 並列化方式. ICOT TR 申請中, 1993.

(10) W. McCune and L. Wos. Experiments in Automated Deduction with Condensed Detachment. In *Proc. of CADE'92*, pages 209–223. Springer Verlag, 1992. LNAI 607.