補題と対偶ホーン節を用いたモデル生成木の刈込み

3P−2

松下 慎　長谷川 隆三　藤田 博
(九州大学大学院システム情報科学府 知能システム学専攻)

1 序論

モデル生成法は，一階述語論理の定理証明法である．モデル生成法の証明は公理 (正節) から始まり，推論規則 (混合節) を適用して次々と定理 (アトム) を生成していく過程を，目標とする定理 (負節) が得られるまで続ける，ボトムアップ実行に基づいている．そしてモデル生成法に基づいた定理証明器として，MGTP(Model Generation Theorem Prover) が開発されている．

本論文では MGTP に 2 つの手法を取り入れて探索空間の刈込みを狙う．

1 つ目の手法は，入力する節集合と共にそれらの対偶ホーン節を用いてモデル生成を行うものである．対偶ホーン節を用いることによって証明の分岐が減少し，またモデル候補の棄却を行い易くなることで探索空間を刈込むことが出来ると考えられる．

もう 1 つの手法は，負節によって証明が終了した枝から補題を作成し，他の枝に加えることによって，部分証明の重複を避けることができると考えられる．

以降の節では，モデル生成法，および 2 つの拡張方法について説明した後，拡張前後の MGTP について実験を行い，その結果について議論を行う．

2 モデル生成法

問題は，含意形式の節 $A_1 \wedge \ldots \wedge A_n \to B_1 \vee \ldots \vee B_m$ の集合で与えられる．ここで，$A_i (1 \le i \le n)$ および $B_j (1 \le j \le m)$ は原子論理式 (アトム) である．$\to$ の左側を前件部，右側を後件部という．$n = 0$ のとき，前件部を特に $true$ と書き，正節と呼ぶ．一方，$m = 0$ のとき，後件部を特に $false$ と書き，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．また，節が基礎アトムの集合 M に基礎置換 σ のもとで違反 (violated) しているとは，$\forall i (1 \le i \le n) A_i\sigma \in M \wedge \forall j (1 \le j \le m) B_j\sigma \notin M$ であることをいう．

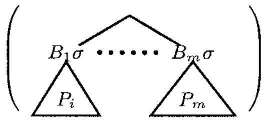
図 1 にモデル生成法による証明手続きを示す．手続き mg は，真であると考えられる基礎アトムの集合 Mc(モデル候補) と節集合 S を受け取り，S の (部分) 証明木を返す．Mc は空集合 $\emptyset$ に初期設定される．

procedure $mgtp(S) : P$;
　/* 入力 (S)：節集合，出力 (P)：S の証明木 */
　return$(mg(\emptyset, S))$;

procedure $mg(Mc, S) : P$;/*入力 (Mc):モデル候補*/

(1) (モデル棄却) 負節 $(A_1 \wedge \ldots \wedge A_n \to false) \in S$ が，Mc に基礎置換 σ のもとで違反している場合：
return $(\frac{|}{\perp})$

(2) (モデル拡張) 混合節もしくは正節 $(A_1 \wedge \ldots \wedge A_n \to B_1 \vee \ldots \vee B_m) \in S$ が，Mc に基礎置換 σ のもとで違反している場合：全ての $i(1 \le i \le m)$ について，$P_i = mg(Mc \cup \{B_i\sigma\}, S)$ とおいて，

return


(3) (モデル発見) (1) および (2) がいずれも適用できない場合：**return** $(\frac{|}{\top})$

図 1: モデル生成手続き

3 対偶ホーン節の作成：UR 化

節の UR 化 (Unit Resulting) は与えられた節の各リテラルをそれぞれ後件とし，元の節と同意義なホーン節，すなわち対偶ホーン節を作成する．

入力節	$p \wedge q \to r \vee s.$
UR 化された節	$p \wedge q \wedge \neg s \to r.$
	$p \wedge q \wedge \neg r \to s.$
	$q \wedge \neg r \wedge \neg s \to \neg p.$
	$p \wedge \neg r \wedge \neg s \to \neg q.$

入力節の UR 化による利点は，MGTP におけるモデル拡張はホーン節を優先して行うことから，枝分れが少ない効率的な証明が可能となることである．また，モデル候補に負リテラルが導入されるので，モデル候補が矛盾する可能性が増す．矛盾したモデル候補を棄却するために，図 2 に示す新しい規則が必要になる．

(0) (モデル矛盾) $\exists A (A \in Mc \wedge \neg A \in Mc)$ のとき：**return** $(\frac{|}{\perp})$

図 2: モデル矛盾の手続き

Pruning unnecessary branches in model generation trees by using contrapositive horn clauses and lemma,
Makoto Matsushita, Ryuzo Hasegawa, and Hiroshi Fujita, Department of Intelligent Systems, Graduate School of Information Science and Electrical Engineering, Kyushu University,

4　補題の導入：重複証明の削除 [1]

図3左の節集合に対してモデル生成法を用いると，図3右のような証明木が得られる．ここで，図3の証明木の網掛け部分は全く同じ証明になっている．このように証明が分岐した場合、それぞれの証明は全く独立に行われるため、各分枝で同じ証明を繰り返し行う可能性がある．このような重複証明があると証明木が余分に増えるとともに、モデル生成に時間がかかることになる．

C1:　$true \rightarrow t \vee p.$
C2:　$p \rightarrow q \vee s.$
C3:　$q \rightarrow r.$
C4:　$s \rightarrow r.$
C5:　$t \rightarrow p.$
C6:　$p \wedge r \rightarrow false.$

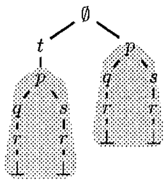


図 3: 証明分岐後の重複証明

ここで補題を導入し，図3での重複証明を削除することを考える．この証明木を作る過程で，左側の網掛け部分の部分の証明に用いられた節は C2，C3，C4，C6 である．これらの節より負節 $p \rightarrow false$ を導出し，これを補題とする．そして右側の網掛け部分の証明において p が現れた時点で補題 $p \rightarrow false$ を用いて即座に $false$ であると結論づけることで，重複証明を回避する．この証明木を図4に示す．図3の証明木と比べると，木の大きさが，ほぼ半減したことが分かる．

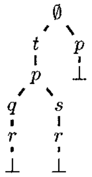


図 4: 図3の重複証明を除去した証明木

5　実験および結果

ある節集合を，無操作 (UR 化，補題無し)，UR 化のみ，補題のみ，UR 化+補題の 4 つの条件のもとで MGTP で解き，その結果を比較した．UR 化は全ての入力節について行った．比較する項目として枝数および証明時間を用いた．問題には The Second DIMACS Implementation Challenge [2] から 59 問を選択して用いた．

枝数

ほぼ全ての問題に関して枝数の減少が見られた．表1に無操作の場合の枝数に対する，各々の枝数の比率の平均，最小値，最大値を示す．

証明時間

UR 化のみを用いた場合，全 59 問中 36 問 (61.0%) の問題で証明時間の短縮が見られた．補題を用いた場合および補題と UR 化を同時に用いた場合では 47 問 (79.7%) の問題で証明時間が短縮した．表2に無操作の場合の証明時間に対する，各々の証明時間の比率の平均，最大値，最小値を示す．

表 1: 無操作に対する枝数の比率 (%)

条件	平均	最小値	最大値
UR 化	19.92	4.75×10^{-4}	50
補題	5.12	3.21×10^{-4}	100
UR 化+補題	1.81	2.73×10^{-4}	50

表 2: 無操作に対する証明時間の比率 (%)

条件	平均	最小値	最大値
UR 化	230.2	1.22×10^{-1}	1291
補題	37.9	3.91×10^{-2}	250
UR 化+補題	50.7	4.20×10^{-2}	283

6　考察

前節の結果から，MGTP に対する UR 化，補題の導入，およびそれら 2 つの相乗効果について考察する．

枝数については，どの条件下でも減少が見られる．特に，UR 化と補題を同時に用いると，より大きな枝数の減少が得られることが分かった．

証明時間に関しても，多くの問題で時間の短縮が得られた．とくに補題を用いた場合の効果は大きい．UR 化と補題を同時に用いると，補題による効果が大きく働き，それによって証明時間が短縮されている．しかし UR 化の影響によって，補題のみの場合より証明時間が長くなる場合がある．一方で，表2中には現れないが，両者がうまく働きあうような問題では，さらなる短縮が可能となっている．

以上より，本論文の目的であるモデル生成木の刈込みは達成されたと言える．証明時間に関しても，多くの問題で証明時間の短縮が得られたため，節の UR 化と補題の導入の有効性は十分に示すことが出来た．

今後の課題として，UR 化を行う節の選択や述語論理に対する効果の調査などがあげられる．

参考文献

[1] 越村 三幸, 長谷川 隆三: 証明の依存性解析による定理証明の冗長探索の削除, 人工知能学会誌, Vol.15, No.6, pp.1064-1073 (2000)

[2] D. S. Johnson and M. K .Trick: The Second DIMACS Implementation Chalenges, DIMACS Series in Discrete Mathematics and Theoretical Computer Sciense (1993)