1D5-OS-11b-2in

CDCL ソルバーのための軽量動的包摂検査

Simple Dynamic Subsumption for CDCL Solver

杉本 拓也 *1　　鍋島 英知 *2
Takuya Sugimoto　　Hidetomo Nabeshima

*1山梨大学医学工学総合教育部コンピュータ・メディア工学専攻
Computer Science and Media Enginnering, Department of Education Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi

*2山梨大学医学総合研究部
Department of Research Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi

In this paper, we propose an extension of the dynamic subsumption algorithm for CDCL (conflict-driven clause learning) solvers proposed by Hamadi et al. The clause learning can be formalized as a resolution process. The dynamic subsumption efficiently checks whether a resolvent subsumes the parent clause or not. The subsumed clause can be removed. In this study, we extend the subsumption checking not only for a parent, but also for ancestors. Our approach can check the subsumption relation with almost no overhead. The experimental results show that our technique can improve the performace of a CDCL solver.

1. はじめに

近年，命題論理式の充足可能性判定問題である SAT 問題 (satisfiablity testing) を解くソルバーの性能は飛躍的に向上しており，ハードウェア・ソフトウェアの検証やプランニング，スケジューリングなどの様々な分野に応用されている．SAT 問題を効率よく解くための重要な技術の１つが簡単化である．SAT 問題の簡単化には，前処理として求解前に実行される静的簡単化と求解中に実行される動的簡単化の 2 種類がある．基本的な簡単化手法の 1 つである包摂検査は，静的および動的に実行する手法がこれまでにも提案されている．前者の例としては，定評あるプリプロセッサでもある SatElite [3] があり，後者の例としては，Hamadi らにより提案された動的包摂検査 [4] がある．

本稿では，Hamadi らの動的包摂検査手法を拡張する．最新 SAT ソルバーの基本アルゴリズムである矛盾からの節学習 (conflict-driven clause learning; CDCL) では，線形融合により節を導出して学習する．Hamadi らの手法では，この学習節の導出過程において，融合節が親節を包摂するかどうかを効率的に検査し，もし包摂する場合は，その親節を融合節と交換することにより簡単化を行う．Hamadi らの手法では各融合操作における親節のみを包摂検査の対象とするが，本稿ではそれを拡張し，祖先の節との包摂検査を行う手法を提案する．本手法は，すべての包摂関係を検出できるわけではないが，その一方で非常に高速に計算可能である．評価実験により，本手法がソルバーの性能改善に有用であること実証する．

2. SAT 問題と CDCL ソルバー

命題変数またはその否定を**リテラル**と呼び，リテラルの選言を**節**と呼ぶ．節の連言が CNF **式** (conjunctive normal form) であり，SAT 問題は通常この形式で与えられる．以下に例を示す．

$$(a \vee \neg b \vee c \vee d) \wedge (\neg a \vee c) \wedge (b \vee \neg c) \wedge (\neg b \vee \neg c)$$

ここで a, b, c, d は命題変数であり，a や $\neg a$ がリテラル，$(a \vee \neg b \vee c \vee d)$ が節である．論理式全体を充足する真偽値割り当てが存在するならば，その SAT 問題は**充足可能**であり，そうでない場合には**充足不能**である．

SAT ソルバーの基本動作の概略を述べる．まず，節内に未割り当てのリテラルを 1 つしか持たない**単位節**を充足するため，単位節内のリテラルに真を割り当てる．単位節への値割り当ては，新たな単位節を生むことがあるため，単位節が存在しなくなるまでこの割り当てを繰り返す．この割り当ての連鎖を**単位伝搬**と呼ぶ．単位節がなければ変数選択ヒューリスティクスを用いて未割り当てのリテラル選択し，真もしくは偽を割り当てる．SAT ソルバーはこの単位伝搬と変数選択を繰り返しながら動作する．単位伝搬において同じリテラルに対し，真と偽を同時に割り当てなくてはいけない状態が発生することを**矛盾**と呼ぶ．矛盾が発生した場合，以降の探索において同様の矛盾が繰り返し発生することを防ぐために，矛盾の原因を解析し，その原因を否定した節を節集合に加える [2, 5]．この節を**学習節** (learned clause) と呼ぶ．学習節は，矛盾した節から始めて，矛盾の発生に関わった節を順次（線形）融合することにより導出可能である．以下に具体例を示す．

$$\begin{aligned} C_1 &= (a \vee b) \\ C_2 &= (a \vee c) \\ C_3 &= (\neg b \vee d) \\ C_4 &= (\neg b \vee \neg c \vee \neg d) \end{aligned}$$

C_i $(1 \leq i \leq 4)$ とする．σ_i $(i < n)$ は節同士を融合することで得られる節である．変数選択の結果，a に偽が割り当てられたとする．C_1 が単位節となり，それを充足させるために b に真が割り当てられる．b に真が割り当てられたため，C_2 が新たな単位節となる．C_2 を充足させるために c に真が割り当

連絡先: 杉本 拓也，山梨大学大学院医学工学総合教育部コンピュータ・メディア工学専専攻，〒 400-8511 山梨県甲府市武田 4-3-1, E-mail:sugimoto@nabelab.org

てられる．この単位伝搬により，次に C_3 と C_4 が単位節になる．これらの節を充足させるためには d に真と偽を同時に割り当てをする必要があり，矛盾が発生する．この矛盾が起きた節と単位伝搬を利用し，解析を行う．最初に矛盾の原因となった C_3 と C_4 を d について融合し，$\sigma_2 = (\neg b \vee \neg c)$ を得る．次に C_3 と C_4 の節を単位節にした割り当ての中で c を先に辿る．c が割り当てられたのは C_2 であるため，C_2 を σ_2 と融合させ，$\sigma_1 = (a \vee \neg b)$ を得る．同じように b についての割り当てを辿っていく．b が割り当てられたのは C_1 であり，$\sigma_1 = (a \vee \neg b)$ と融合させ $\sigma_0 = (a)$ を得る．割り当てを辿ると次は a であるが，a は単位伝搬によって一意に値が決定されたリテラルではなく，変数選択によって真偽値を割り当てられたリテラルであるため，これ以上単位伝搬を辿ることはできない．そのため σ_0 を学習節として節集合に加え以下の式 1 に融合過程を示す.

$$\dfrac{C_1 \qquad \dfrac{C_2 \qquad \dfrac{C_3 \quad C_4}{\sigma_2}}{\sigma_1}}{\sigma_0}$$

式 1:節同士の融合過程

3. 動的包摂検査 [4]

SAT 問題を簡単化することで，ソルバーの求解時間は一般に短縮できる．簡単化手法の１つが包摂検査である．節 C が節 D を包摂するとは，C 中のリテラルが全て D に含まれるとき，すなわち $C \subseteq D$ が成立する場合をいう．例えば，$C = (\neg b \vee c)$, $D = (a \vee \neg b \vee c)$, のとき，$C$ は D を包摂する．このとき，両者の節を充足するためには $(\neg b \vee c)$ が真でなくてはならないため，D 中のリテラル a の真偽値は式 $C \wedge D$ の充足可能性に影響を及ぼさない．よって，節 D は削除可能である．

包摂を利用した簡単化において，節同士の融合を利用した検査が存在する．前述の $C = (\neg b \vee c)$ が, $D = (a \vee \neg b \vee c)$ と $E = (\neg a \vee \neg b)$ を融合して得られた融合節だとする．このとき，$C \subseteq D$ が成立するため，融合節 C を節集合に加えることで親節である D を削除することができる．これは，融合節との包摂検査によって D 冗長なリテラル a を削除できたともいえる．しかしあらゆる 2 つの節同士の組み合わせに対し，その融合節を求めることには計算コストがかかり，また必ずしも包摂関係が見つかるとは限らない．そこで Hamadi らが提案した動的包摂検査では矛盾からの節学習において行われる融合操作を利用することで，包摂検査に必要な計算コストを大幅に削減している．矛盾からの節学習では，式 1 に示したように，最初の融合ステップでは，実在する節同士を融合するが，以降の融合ステップでは，親節の一方のみが実在する節となる．本稿では，CNF 式中に存在する節を**実節**と呼ぶ．動的包摂検査では，融合節がその親実節を包摂するかどうかを検査する．以下では，Hamadi らの動的包摂検査手法について簡単に説明する．

融合節は親節が含むリテラルを 1 つ除いて全て持つ．実節がもう片方の親が持つリテラルを符号が異なるリテラルを除き持っていたならば，融合節は実節を包摂する．これらは親節の共通するリテラルを調べることで導出可能であり，これは節同士の融合操作と同時に行えるため余分な計算コストが必要ない．検査の結果，包摂関係が見つかった場合，親節からリテラルを削除し包摂結果と等しくする．

4. 拡張動的包摂検出手法

まず，節 σ を導出する線形融合において，n 世代前の親節（実節）を P^n_σ, n 世代前の融合節 (内接) を R^n_σ と表記する．P^1_σ までの動的包摂検査を P^2_σ まで拡張し，更に後術する条件を満たした場合，任意の祖先 $(P^n_\sigma (n \geq 3))$ との包摂検査まで行う．これらをまとめて**拡張動的包摂検査**と呼ぶ．計算コストを抑えた検査を実現するため，拡張動的包摂検査では親節とその融合節に含まれるリテラル数の変化に注目する．2 世代前の親実節 P^2_σ と融合節 R^0_σ が包摂関係にある場合，

$$|P^2_\sigma| > |R^0_\sigma|$$

が必要条件にあたる．P^2_σ に含まれる節の数が n ならば，節融合で得られる $|R^0_\sigma|$ の数は最小でも $n-2$ である．なぜなら，１回の融合操作において融合節がその親節より小さくなる場合は，前章で述べたとおり，高々１つであるためである．特定の親節と融合節に注目した場合，融合節は親節からリテラルを１つ取り除いたものでなければ小さくはならず，それは親節と融合節の間の包摂関係を証明できる．これらを前提として，P^2_σ と R^0_σ の必要条件が満たされる場合を考えると，2 パターンしか存在しない．n は任意の整数である.

パターン 1

- $|P^2_\sigma| = n$
- $|R^1_\sigma| = n-1$
- $|R^0_\sigma| = n-2$

パターン 2

- $|P^2_\sigma| = n$
- $|R^1_\sigma| = n$
- $|R^0_\sigma| = n-1$

パターン 1 について考える．融合節 R^1_σ が 実節 P^2_σ より小さいため R^1 は P^2 を包摂する．さらに $|R^0_\sigma|$ も親節である R^1_σ より小さいため同様である．従って，連続した包摂関係が存在するため， R^0 は P^2 を包摂する. このとき，P^2_σ からは 2 つのリテラルを削除可能であり， $|P^1_\sigma| = n-1$ ならば $|R^0_\sigma|$ が包摂する 2 つ目の節になるので，このとき，P^1 からはリテラルを 1 つ除去可能であり，P^2 は削除可能である.

パターン 2 については，包摂関係が成り立つ場合と成り立たない場合がある．すなわち，節の大きさの変化だけでは包摂関係を証明できない場合がある．包摂を証明するためには， P^2_σ と R^0_σ の間で共通するリテラル数を調べる必要があるが，それには計算コストがかかるため，ここでは融合過程の情報を利用する．パターン 2 では，$|P^2| = |R^1|$ であるため，P^2_σ と R^1_σ の間には共通しないリテラルが 1 つだけ存在している．それは R^2_σ に含まれているリテラルであり，それが P^1_σ との融合で削除されていれば，R^0_σ は P^2 を包摂する．そこで R^2_σ 由来のリテラルを調べるために，各リテラルに対し，新たに *latest source* という要素を付属される．リテラル L の *latestsource* は L を融合節に追加した最後の親節を表す．すなわち，全てのリテラルに対して，どの節から融合されたかの情報を簡単に記憶する．これは 1 度削除されたリテラルがその学習中に現れることがない，矛盾からの節学習における線形融合のみで適用可能は方法である．この値はリテラル同士の融合と同時に更新された

め，計算コストがほとんどかからない．$P_\sigma^2 = (a \vee \neg b \vee c \vee d)$ が 3 回目の融合された節としたとき，P_σ^2 に含まれる融合で消えるリテラルを除いた全てのリテラル (例えば，b, c, d) に 3 という情報が付加される．そして $P_\sigma^1(\neg e \vee c \vee d)$ ならば，そのうちのリテラル (例えば，c, d) に 4 回目なので 4 という情報が上書きされる．この値を用いることで，融合によって消えるリテラルがどの節由来なのかを知ることが可能である．$|R_\sigma^0|$ において，もしも親節 P_σ^2 由来のリテラルであれば *latest source* は 3 である．包摂関係が成立する $|R_\sigma^2|$ 由来のリテラルでならば，P_σ^2 よりも小さい値であるため 2 以下である．

P_σ^1 で削除されるリテラルを a，現在の融合回数を r とし，リテラル a の $latestsource$ を $a_{latestsource}$ と表す．パターン 2 について以下の条件が必要である．

$$a_{latestsource} < r - 1$$

従ってパターン 2 は節の大きさの変化と特定のリテラルを 1 つ調べるだけで，包摂関係を証明可能である．

ここまでで P_σ^2 との包摂関係をわずかなコストで調べる方法を説明した．P_σ^3 以降の任意の節 ($P^n(n \geq 3)$) との包摂関係については節の大きさによる証明が困難になるため P_σ^2 との包摂検査を拡張して対応する．パターン 1 を拡張すれば，融合節が小さくなり続ける限り，特定の実節 (P_σ^n) に含まれるリテラルを消すことが可能である．また，パターン 2 の拡張は P_σ^n と R_σ^{n-1} の間に存在する等しいリテラル数に注目する．融合節 R_σ^{n-1} の大きさが n と等しい間待ち続け，融合節が $n-1$ になっときに包摂するかを調べる．ただし，この間の包摂の可能性が維持されているか確認するために，*latest source* で削除されたリテラルが共通するリテラルではないことを調べ続ける．しかし，これでも不十分であるため，融合節が小さくなるときに，$|P_\sigma^n|$ を呼び出し，融合節との間で直接包摂関係を調べる．以下に例を示す．

$$\begin{aligned}
C_1 &= (a \vee b \vee c \vee d \vee g) \\
C_2 &= (\neg e \vee c \vee d \vee b \vee f \vee g) \\
C_3 &= (\neg f \vee b \vee d \vee g) \\
C_4 &= (\neg c \vee d \vee g) \\
C_5 &= (\neg b \vee d \vee g) \\
C_6 &= (\neg d \vee g) \\
\sigma_1 &= (\neg a \vee b)
\end{aligned}$$

C_i は，学習節の導出に使われた節であり，σ_1 は上記の節以外から導かれた融合節とする．σ_1 に対し，学習節の導出過程に従い $C_{1\sim6}$ を融合していく．その過程と結果を図 1 に示す．既存手法では，P_σ^1 のみと包摂関係を調べるため，C_2 と σ_3 と，C_3 と σ_4, C_4 と σ_5, C_5 と σ_6 でそれぞれ包摂関係が発見される．結果として得られる図 2 ではかなり簡単化されているが，類似した節が多く残る．

拡張動的包摂検査を適応した場合を考える．まず C_1 と σ_2 のリテラル数が等しいため，C_1 と σ_3 の包摂検査を行う．しかし，これらの節同士もリテラル数が等しいためパターン 2 を拡張した検査を行う．削除されたリテラル e の $latestsource$ の値は σ_1 を示す 1 である．e は C_1 に含まれないリテラルため包摂の可能性は残る．次は C_3 と σ_4 の節の大きさ変化を確認する．この時に $|C_3| > |\sigma_4|$ ならば，C_1 は σ_4 に包摂される可能性がある．この場合は節同士を直接比べて包摂関係を確認する．ここでは包摂関係が存在するため，C_1 からリテラル a を削除できる．また C_3 は融合節が小さくなり続ける間のみ，連続でリテラルを削除可能．同様の手法を繰り返し行うことで，図 3 のような節集合を得られる．結果として得られる節郡は，既存手法よりも簡単化されている．

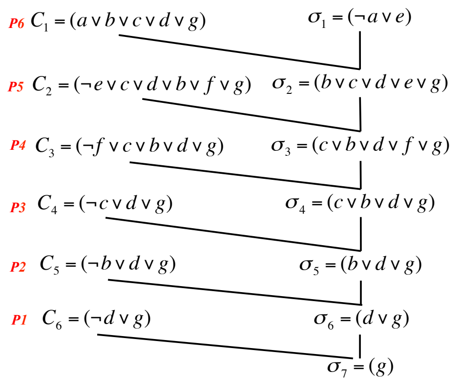

図 1: 節同士の線形融合

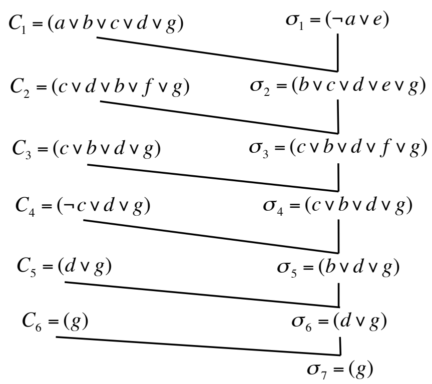

図 2: 動的包摂検査のみ適用後

5. 評価実験

拡張した動的包摂検査を CDCL ベースの SAT ソルバーである GlueMiniSat[7] へ実装した．GlueMiniSat は 2011 年の SAT Competition アプリケーション部門の UNSAT 部門と SAT+UNSAT 部門で優秀な成績を残したソルバーである．任意の節との包摂検査については今回はアルゴリズムの提案のみにとどめ，P^1，P^2 と R^0 に多雨する包摂検査のみ実装する．実験環境を以下に示す．使用した SAT ソルバーは GlueMiniSat2.2.7，問題は SAT Competetition 2013 の Application 部門より 300 問．CPU は Core2 Duo 1.83GHz，メモリは 2GB．制限時間は 1 問あたり 1000 秒とした．SAT は充足可能な問題，UNSAT は充足不可能な問題を表す．SAT ソルバーが解にたどり着けば求解とする．GlueMiniSat に動的包摂検査を加えた物を +DS，さらに拡張動的包摂検査を加えてものを +DS+DS$_{anc}$ とする．図 4 で示される結果では，提案手法である+DS$_{anc}$ は途中までは既存手法である+DS に抜かされてしまっているが後半を見れば追い越している．表 1 より求解数も UNSAT では負けているものの全体の求解数では GlueMiniSat や GlueMiniSat+DS よりも多くなっている．

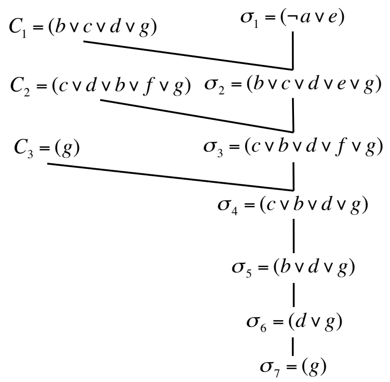

図 3: 拡張動的包摂検査の適用した場合

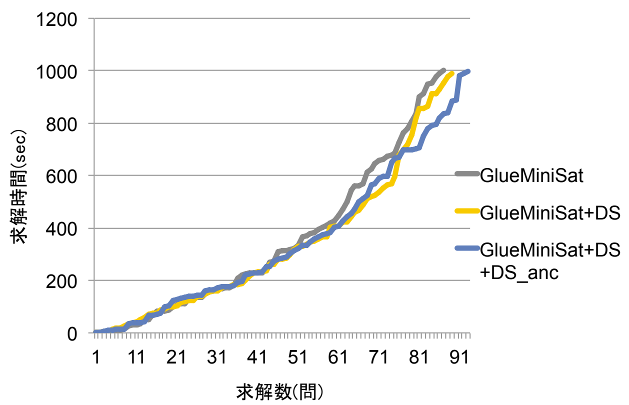

図 4: 求解数と求解時間の変化

リテラルを削除すると単位伝搬や，変数選択ヒューリスティクスに影響を及ぼすため，その結果として無駄な探索が生じてしまったのだと考える．

求解時間が 700 秒以下の時はそのような場合が多く含まれているのではと予想する．一方で求解時間が多くかかっていれば簡単化はそれだけ多く行われており，表 2 より包摂によるリテラルの削除数が Hamadi らの手法の約 2 倍になっているため，節内の多くの節を簡単化することで求解に辿り着いたのではと予想する．

6. まとめと今後の課題

本研究では，Hamadi らの動的包摂検査手法を拡張し，任意の祖先との包摂検査を行う軽量な手法を提案した．この手法のうち，融合節とその親および祖父節との包摂関係を検査する手法を実装し，評価実験の結果，特に SAT に有効であることを示した．今後の課題として，任意の祖先との包摂検査手法の実装と評価と，充足不能な問題に対する性能低下の原因解析と改善が挙げられる．評価実験により，CDCL ベースのソルバーに手法を実装し，SAT と UNSAT の共にその性能の向上を確かめた．任意の祖先との包摂検査については，簡単なアルゴリズムのみを提案した．

今後の課題として，任意の祖先との包摂検査の実装と，充足不能な問題に対するアプローチが課題である．

表 1: 包摂回数

	包摂回数
GlueMiniSAT+DS	3027684
GlueMiniSAT+DS+DS$_{anc}$	5886558

表 2: 求解数

	SAT	UNSAT	Total
GlueMiniSAT	62	23	85
GlueMiniSAT+DS	60	29	89
GlueMiniSAT+DS+DS$_{anc}$	66	26	92

参考文献

[1] Gilles Audemard and Laurent Simon. Predicting learnt clauses quality in modern sat solvers. In *IJCAI*, Vol. 9, pp. 399–404, 2009.

[2] Roberto J Bayardo Jr and Robert Schrag. Using csp look-back techniques to solve real-world sat instances. In *AAAI/IAAI*, pp. 203–208, 1997.

[3] Niklas Eén and Armin Biere. Effective preprocessing in sat through variable and clause elimination. In *Theory and Applications of Satisfiability Testing*, pp. 61–75. Springer, 2005.

[4] Youssef Hamadi, Saïd Jabbour, and Lakhdar Saïs. Learning for dynamic subsumption. *International Journal on Artificial Intelligence Tools*, Vol. 19, No. 04, pp. 511–529, 2010.

[5] João P Marques-Silva and Karem A Sakallah. Grasp: A search algorithm for propositional satisfiability. *Computers, IEEE Transactions on*, Vol. 48, No. 5, pp. 506–521, 1999.

[6] Niklas Sörensson and Armin Biere. Minimizing learned clauses. In *Theory and Applications of Satisfiability Testing-SAT 2009*, pp. 237–243. Springer, 2009.

[7] 鍋島英知, 宋剛秀. 高速 sat ソルバーの原理. 人工知能学会誌, Vol. 25, No. 1, pp. 68–76, 2010.