

SAT ソルバーの安定性向上のための粗な初期探索手法の検討と提案

A coarse initial search method for improving the robustness of a SAT solver

三神 直彬 *1

Naoaki Mikami

鍋島 英知 *2

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 Engineering, Department of Education Interdisciplinary

Graduate School of Medicine and Engineering, University of Yamanashi

*2 山梨大学大学院医学工学総合研究部

Department of Research Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi

In the paper, we propose a coarse initial search method for the stability improvement of a SAT solver. Our approach initially searches a SAT instance coarsely in order to guide the search space in the following main search process to promising area. In this study, we introduce two different coarse search methods. The first method ignores conflicts produced by learned clauses in the initial search to get promising search space. The second one is a variable activity sampling method. This method determines a good initial distribution of the variable activity. The experimental results show that our proposed method has possibility to improve stability of SAT solver.

1. はじめに

SAT 問題 (satisfiability testing) とは、与えられた命題論理式の充足可能性を判定する問題であり、Cook[1] により初めて NP 完全性が示された問題である。SAT 問題を解くソフトウェアを SAT ソルバーと呼ぶ。近年、SAT ソルバーは飛躍的に性能が向上しており、高速な SAT ソルバーはハードウェア・ソフトウェア検証、スケジューリング、プランニングといった様々な分野で利用されている。しかしこのような高速 SAT ソルバーは様々な高速化技術を慎重に組み合わせることで実現しており、環境やパラメータの些細な変化で求解可能な問題数が大きく変化してしまうことがしばしば発生する。このような SAT ソルバーの不安定さを解消し頑健性を高めることは、SAT ソルバーのさらなる性能向上のために重要である。そこで本稿では、SAT ソルバーの安定性向上を目的とした、粗な初期探索を行う SAT 問題の求解手法を提案する。提案手法では、SAT 問題の求解序盤で積極的に探索空間を広く荒く探索し、中盤以降で深く詳細な探索を行う。このような探索を行うことでその問題の解空間を推定し、従来の探索よりも安定した探索と求解性能の向上を狙う。また、評価実験を行い提案手法の有効性を検討していく。

本稿の構成は以下の通りである。まず 2 章で SAT 問題と CDCL ソルバーについて説明する。3 章では SAT ソルバーの安定性向上のための粗な初期探索について提案する。4 章で提案手法の性能評価を行い、5 章で考察を述べる。6 章でまとめと今後の課題について述べる。

2. SAT 問題と CDCL ソルバー

SAT 問題は与えられた命題論理式が充足可能であるかどうかを決定する問題であり、通常連言標準形 (Conjunctive Normal Form; CNF) で与えられる。命題変数及びその否定をリテラルといい、前者を正リテラル、後者を負リテラルと呼ぶ。これ

らのリテラルの選言を節と呼び、節を構成するリテラルの個数を、その節の節長という。偽を割り当てられたリテラルを除き、節長が 1 の節を単位節と呼ぶ。以下に CNF 式の例を示す。

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

ここで a, b, c, d は命題変数であり、 a や $\neg a$ がリテラル、 $a \vee b$ や $\neg a \vee c \vee \neg d$ が節である。

近年の高速 SAT ソルバーの多くは、DPLL アルゴリズム [2] に矛盾からの節学習 (conflict-driven clause learning; CDCL)[3][4] を導入した、CDCL アルゴリズムに基づいている。以下で CDCL アルゴリズムの概要を説明する。

CDCL アルゴリズムは与えられた命題論理式に対して充足可能 (SAT)、または充足不能 (UNSAT) を判定するまで以下の手順を繰り返す。

1. 単位伝播: 単位節を探し、これを充足させる変数割り当てを行う。
2. 変数選択: 単位節が見つからない場合、ある変数決定ヒューリスティクスに基づいて変数を選択し、真または偽を割り当てる。決定変数の数を決定レベルという。
3. 矛盾解析: ある真偽値割り当てのもとで矛盾が発生した場合、その矛盾の原因を解析し、同様の矛盾を防ぐための節を学習する。これにより獲得される節を学習節と呼ぶ。また、原因の解析時にバックトラックを行う適切な決定レベルを求めておき、その時点までバックトラックを行う (バックジャンプ)。

以上の手順を繰り返す過程で、与えられた問題のすべての節が充足するような変数割り当てを発見した場合、充足可能となり探索を終了する。決定レベル 0 において矛盾が発生した場合、与えられた問題は充足不能であると判定される。

著名な SAT ソルバーである Minisat[5] では、各変数に活性度という指標を持たせ、矛盾が発生するたびに、その矛盾の導出に関わった変数の活性度を向上させる。また、定期的にすべての変数の活性度を定数で割る。よって、活性度の高い変数は

連絡先: 三神 直彬, 山梨大学大学院医学工学総合研究部 コンピュータ・メディア工学専攻, 〒400-8511 山梨県甲府市武田 4-3-11, E-mail: mikami@nabelab.org

表 1: 粗な初期探索の求解性能の比較

提案手法	求解数 (SAT+UNSAT)
GlueMiniSat	389 (167+222)
Beyond-confl	394 (173+221)

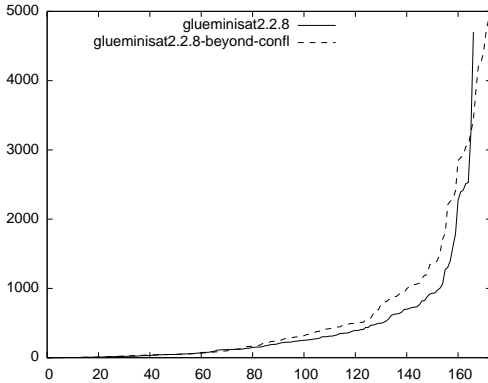


図 1: SAT な問題の求解速度

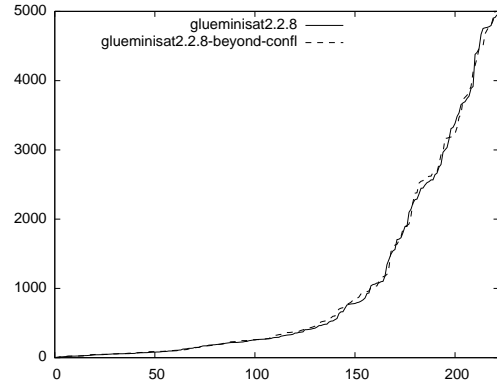


図 2: UNSAT な問題の求解速度

局所的な矛盾を引き起こしやすい変数であると考えることができ、Minisat ではこれを変数決定ヒューリスティクスとして利用し、優先して真偽値割り当てを行っていく。

3. 粗な初期探索

SAT ソルバーの安定性を向上させるための代表的な手法が、ポートフォリオ型並列 SAT ソルバーである。これは、種類の異なる複数のソルバーまたは探索パラメータを変化させた複数のソルバーを並列に実行し、複数の空間を一斉に探索する手法である。一つの問題に対して、複数のソルバー・パラメータを用いた求解を行うことで安定性を向上させることが可能となる。この他、逐次型ソルバーにおける安定化向上の手法として、Minisat では変数の活性度の初期値をランダムに決定する手法を実装している（本稿ではこれを random-activity と呼ぶ）。通常、すべての変数の活性度の初期値は 0 であるが、これをランダムにすることで探索開始時の変数選択の偏りを低減し、安定性を向上させる狙いがある。

提案手法ではまず、探索の序盤で探索空間を広く浅く探索する粗な探索を行う。中盤以降では、序盤の粗な探索から得た情報を元にして通常の探索を行い、探索空間を深く調べていく。このような探索により、序盤では局所的な探索に注力することを避け、解の存在する空間の絞り込みを行うことで、安定性の向上を図る。本稿では粗な探索を行う探索序盤のことを初期探索と呼ぶ。

3.1 初期探索

提案手法では初期探索の定義として、全変数中の活性度が変化した変数の割合を利用した。活性度が変化した変数の割合が一定値を超えた場合、初期探索が終了したと判断する。初期探索終了時には一定割合の変数の活性度が変化していることになる。

3.2 矛盾無視による粗な初期探索

初期探索中では、学習節から発生する矛盾をすべて無視し探索を継続する手法である。学習節は探索空間の枝刈りを行う効果がある。矛盾を無視することは、学習節が存在しなかった場合の探索を行うことに等しい。学習節による探索空間の枝刈

りを一時的に無効化することで、より広い探索空間を調べていくことができると考える。

提案手法では、ある節から矛盾が発生した場合にその節が学習節かどうかを確認し、学習節だった場合はその矛盾から獲得した学習節を破棄し単位伝播を継続する。バックジャンプも行わない。例外として、明らかに有用であると考えられる学習節を獲得した場合は破棄せず、バックジャンプを行う。今回の実装では、節長が 2 以下の学習節を明らかに有用であると判断する。

4. 粗な初期探索の評価実験

GlueMiniSat2.2.8[6] に提案手法を実装し、提案手法の性能向上の可能性について評価実験を行った。実験に使用した問題は、SAT Competition[7] 2013 Application 部門の問題 300 問と SAT Competition 2014 Application 部門の問題 300 問のうち、重複する問題を除いた 512 問である。実験は 1 問当たり制限時間 5000 秒、CPU Intel(R) Xeon(R) E3-1230 V2 @ 3.30GHz、メモリ 8GB の環境で行った。なお、初期探索の終了条件は、活性度が変化した割合が全変数中の 30% を超えた場合とした。

実験結果を表 1 に示す。GlueMiniSat は通常の GlueMiniSat2.2.8 での実行結果、Beyond-confl は矛盾無視での実行結果を示している。表 1 より、Beyond-confl では SAT な問題の求解数が 6 問向上し、UNSAT な問題の求解数が 1 問低下していることがわかる。全体として 5 問の性能向上が見られた。図 1、5.1 は、各ソルバーでの SAT な問題と UNSAT な問題についての求解時間を表すグラフである。縦軸が求解時間で、横軸が求解数を表している。グラフが右下にある程、求解速度が速いソルバーとなる。図 1 より SAT な問題において、求解時間の少ない問題では提案手法の方が求解性能は低い。しかし求解時間が 3000 秒を超えた辺りから求解性能が逆転していることが分かり、提案手法は SAT な問題においては大规模な問題に効果が高いと考えられる。また、UNSAT な問題での求解性能を示す図 5.1 より提案手法は、求解数は減少しているが、GlueMiniSat とほぼ同等の求解速度を持っていると考えられる。

表 2: 活性度を用いた初期探索の求解性能の比較

提案手法	求解数 (SAT+UNSAT)
GlueMiniSat	228 (107+121)
Beyond-confl	231 (113+118)
Act-sampling	225 (109+116)

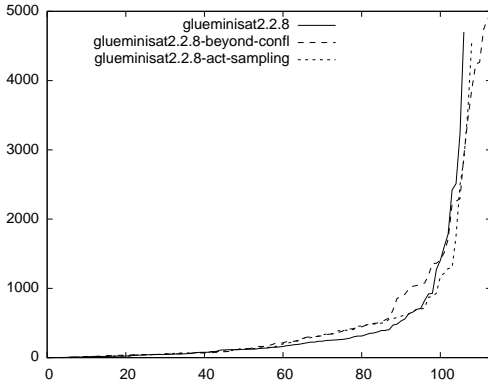


図 3: SAT な問題の求解速度

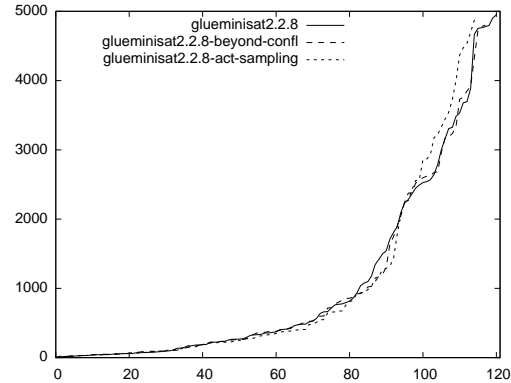


図 4: UNSAT な問題の求解速度

5. 考察

提案手法は本来の GlueMiniSat2.2.8 と比較して SAT な問題において性能向上が見られた。しかし SAT な問題は僅かな変化で求解に要する時間が大きく変動してしまうことがわかっている。そのため、6 問の求解数増加では十分な性能向上が見られたとは言い難い。そこで提案手法の性能向上の可能性について、調査と実験を行った。

5.1 初期探索から引き継ぐ活性度

初期探索は問題を大まかに探索することでその後に探索すべき空間を推定する狙いがある。初期探索終了後、通常の探索に移行する際に引き継がれる情報は、主として各変数の活性度と有用と判断された学習節である。粗な初期探索から引き継ぐ情報として、変数の活性度と学習節のどちらが価値があるのかを検証するために、活性度に重点を置いた活性度のサンプリング手法を提案し、矛盾無視による粗な初期探索との比較を行う。

活性度のサンプリング手法は、random-activity からさらにコストをかけることでより良い活性度の初期値獲得を目指す手法である。初期探索として、random-activity を用いて初期化された活性度での探索を行う。初期探索が終了する度に、その時点での変数の活性度を保存する。その後再び変数の活性度を random-activity に基づいて初期化し、初期探索を再び行う。この動作を一定回数繰り返す、繰り返した回数分の活性度の分布を獲得する。これらの分布はそれぞれ異なる活性度の初期値から探索を行っているため、いずれも異なる分布になっていることが考えられる。獲得した活性度分布を元に、本探索での活性度の初期値を決定する。今回の実装では、獲得した活性度分布それぞれに対して、活性度の高い順に変数の順位付けを行った。そして全試行を通して平均順位が高い変数ほど、活性度の初期値として高い値を与えた。

この手法について、4. 章で示した GlueMinisat2.2.8、矛盾無視による粗な初期探索との比較実験を行った。使用問題は SAT Competition 2013 Application 部門の問題 300 問で、実験環境・制限時間は前述の条件と同様である。提案手法のパラメータとして、サンプリング回数を 10 回、初期探索終了と

判定する活性度の変化割合を 10% とした。実験結果を表 2 に示す。Act-sampling が本節で提案した活性度のサンプリングによる初期探索を実装した結果である。SAT な問題では 2 問の向上が見られたが、UNSAT な問題で 5 問の低下が見られ、全体として求解数は 3 問減少したという結果となった。求解速度の比較を表すグラフを図 3, 4 に示す。図 3 より、SAT な問題においては、求解速度の傾向は矛盾無視による粗な初期探索とほぼ同じであると考えられるが、全体として性能は低下していることが分かる。図 4 より、UNSAT な問題においても、グラフ上ではほぼ同等の求解速度だった GlueMiniSat と Beyond-confl に対して、Act-sampling は性能が低下していることが分かる。結果として、本節で提案した手法を用いた活性度に重きを置いた初期探索では、性能向上は見られなかった。

5.2 探索過程での活性度の変動

図 5.1 は、SAT Competition 2013 Application 部門のとある 2 つの問題について、GlueMiniSat2.2.8 で求解したときの変数の活性度の変化を表したグラフである。横軸は求解の時間経過を表しており、縦軸は変数番号を求解時の活性度で降順ソートしたものとなる。グラフ上には、各ステップでの活性度が高い変数の上位 1% がプロットされている。プロットされている点がグラフ上で縦軸の低い位置に集まった場合、活性度が高くなる変数は求解中のどの時点でも概ね同じであることを表している。点が広範囲に分布している場合、活性度が高くなる変数は常に変動していることを表している。図 5.1 より、いずれのグラフでも点は非常に広範囲に分布しており、活性度が高くなる変数は求解中で大きく変化していることがわかった。またこれらの問題に対して、random-activity を用いた場合でも同様の傾向が見られた。提案手法では活性度の初期値を変化させているに過ぎないため、初期探索で獲得した活性度の分布は本来の探索が始まるとすぐに失われてしまっている可能性が高い。よって活性度の初期値のみで今後の探索を制御することは難しいと考えられる。しかし図 5.1 から、一部の变数には規則性が見られた。特に、求解時に活性度の高かった変数は求解過程でも活性度が高くなる傾向にあることがわかった。今後これらの変数に注目した手法を考案することで、活性度による探

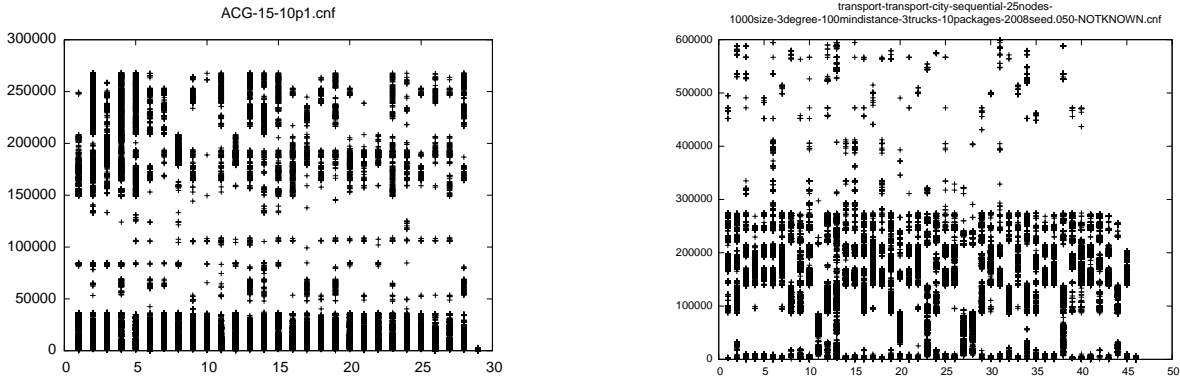


図 5: 活性度の変動

索の制御にも性能向上の可能性があると考える。

5.3 安定性の調査

5.1 節の実験結果を元に、提案手法を導入することで SAT ソルバーの安定性の向上が達成できたのかを検証する。SAT ソルバーの安定性を評価するために、5.1 節の実験と同様の環境・問題で行った、パラメータ調整の結果やこれまでに開発してきたソルバーの実験結果から、すべてのソルバーが解けた問題、解けなかった問題を除いた問題数を調査した。その結果、およそ 50 問程度の問題がこれに該当した。これらの問題はパラメータや探索の戦略を変化させることで解くことができ、SAT ソルバーの安定性を向上させることで求解可能であると考えられる。表 2 より、性能向上が見られた Beyond-conf1 においても求解数の向上は SAT と UNSAT を合わせて 3 問である。これは全体的問題数と比較しても、いずれかのソルバーで求解可能であった問題数と比較しても僅かな数値であり、安定性が向上しているとは言い難い。

5.4 学習節ベースの粗な初期探索の検討

5.3 節より、提案手法の求解性能の向上はごく僅かで、安定性が向上しているとも言い難い。また 5.2 節より、活性度の初期値を変化させることによる探索の制御は難しいことがわかった。そこで初期探索から引き継がれる情報として考えられるもう一つの情報である、学習節の有効性についての検証が今後必要になる。若干の性能向上が見られた矛盾無視による粗な初期探索は、学習節によってブロックされる空間を強引に探索するため、通常の探索では発生する可能性の低い矛盾を引き起こす。そのため、通常の探索とは異なる活性度の変化と学習節の獲得が行われると考えられ、活性度と学習節のどちらも利用している手法であるといえる。粗な初期探索から獲得する学習節が有効であった場合、矛盾無視による粗な初期探索終了時に、学習節のみを引き継がせた場合にも性能向上が見られるはずである。この場合、粗な初期探索を行うことでどのような学習節が新たに獲得できるのか詳細な調査を行うことで、さらなる性能向上につなげることができると考える。

6. まとめと今後の課題

SAT ソルバーの安定性向上を目的とした粗な初期探索手法を提案し、評価を行った。評価実験の結果から、矛盾無視による粗な初期探索では SAT な問題に対して僅かに効果が見られ、性能向上の可能性が見られた。しかし安定性が向上しているとは言い難く、より詳細な調査を行い安定性を高めるための要素を探す必要がある。

また、活性度の初期値に重点を置いた、活性度のサンプリング手法を提案し、活性度の初期値を変化させることで中盤以降の探索を制御することが可能かを検証した。結果として、活性度は求解過程で大きく変動することが確認でき、初期値のみを変化させることでは探索に大きな影響を与えることは難しい可能性が高い。

今後の課題として、より安定性を向上させるための手法を検討することが挙げられる。初期探索終了後に引き継がれる情報としては、活性度と学習節の 2 つが考えられる。活性度に関しては、それ単体では性能向上につなげることは難しいと考えられることがわかった。次の段階として、初期探索で獲得する学習節が探索に有効に働いているのかどうかを検証し、有効であることが確認できた場合にはより良い学習節を獲得するための手法を検討する必要がある。

参考文献

- [1] Cook, S.A.: The complexity of theorem-proving procedures, STOC '71 Proceedings of the third annual ACM symposium on Theory of computing, pp.151-158, (1971).
- [2] Davis, M., Logemann, G. and Loveland, D. W.: A Machine Program for Theorem-Proving, Commun. ACM, Vol. 5, No. 7, pp. 393-397 (1962).
- [3] Bayardo, R. J. and Schrag, R. C.: Using CSP Look-Back Techniques to Solve Real-World SAT Instances, Proceedings of AAAI-97, pp. 203-208 (1997).
- [4] Silva, J. P. M. and Sakallah, K. A.: GRASP: A Search Algorithm for Propositional Satisfiability, IEEE Transactions on Computers, Vol. 48, pp. 506-521 (1999).
- [5] Een, N. and Sorensson, N.: An Extensible SAT-solver, SAT, Proceedings of the 6th International Conference on Theory and Applications of Satisfiability Testing (SAT), pp. 502-518 (2003).
- [6] Nabeshima, H., Iwanuma, K., and Inoue, K.: GlueMiniSat 2.2.8, [http://glueminisat.nabelab.org], (2015-03-17).
- [7] The international SAT Competitions web page, [http://www.satcompetition.org], (2015-03-17).