3J1-OS7-7

Nelson-Oppen法を組み込んだSMTソルバの設計

A Design of a SMT solver incorporated with Nelson-Oppen method

福田　寿志[*1]　岩沼　宏治[*2]　山本　泰生[*2]
Hisashi Fukuda　Koji Iwanuma　Yoshitaka Yamamoto

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

[*2]山梨大学大学院コンピュータ・メディア工学専攻
Department of Computer Science and Media Engineering, University of Yamanashi

SMT is an extended SAT technology to process the background theory described by the first-order logic. Practical background theory usually consists of two or more first-order theories. The Nelson-Oppen method combines two or more individual decision procedures each of which solvers the satisfiable problem for the corresponding background theory. In this paper, we show a design of a SMTsolver which efficiently deals with the Nelson-Oppen method.

1. はじめに

近年の命題論理の**充足可能性判定問題**を解く SAT 技術の進展に伴い，SAT を用いたソフトウェアやハードウェアの形式的検証や人工知能におけるプランニングなどが盛んに研究されている．一般にこれらの検証・推論には，等号や算術などに関する背景知識が必要となる．そのため，命題論理より表現能力の高い一階論理などの論理体系で背景理論を記述できれば，記述がコンパクトで簡潔になり，都合がよいことが多い．

背景理論付き SAT(Satisfiability Modulo Theories: SMT)[3][4] とは，このような命題論理よりも表現能力の高い論理体系で記述された背景理論を SAT 技法で効果的に取り扱うことを目的とした技術である．SMT の実用問題で必要とされる背景理論は複数の一階理論から構成されいるが, **Nelson-Oppen 法** [1] はそれら複数理論の個々の決定手続きを組み合わせて用いるための理論である．しかし, 現在ソースコードが公開されている Nelson-Oppen 法を組み込んだ SMT ソルバは存在しない. 今後, C-Tableau[2] の実装などの各種改良を考えたときに複数の背景理論からなる式を扱える SMT ソルバは重要である. そのため, 本研究ではこの Nelson-Oppen 法を実装した SMT ソルバの設計を行ったので, その報告を行う.

2. SAT 問題

SAT 問題を定義する．SAT 問題は通常**連言標準形 (conjunctive normal form: CNF)** で与えられる．CNF は，命題または命題の否定を表すリテラル (literal) の選言の連言である．SAT 問題を解くとは，全ての節を充足する命題の真偽値割り当てを求めることである．もしそのような真偽値割り当てが 1 つ以上存在するならば充足可能 (SAT) であり，そうでない場合，充足不可能 (UNSAT) と言う．

例えば，次のような CNF 式 $(\neg A \vee \neg B) \wedge (B \vee C) \wedge A$ が与えられたとき，$A : True$, $B : False$, $C : True$ と割り当てることで式を充足できる．

3. SMT 技術

SMT 技術は**事前処理型 SMT 技術**と**遅延処理型 SMT 技術**の 2 つに大別される [4]．事前処理型 SMT 技術とは，背景知識や質問を事前に命題論理式へコンパイル・符号化し，既存の高速 SAT ソルバで最終的に解かせるものである．常に最新の高性能 SAT ソルバを利用できるメリットがある反面，多くの背景理論に対して符号化方式を個別に作成する必要がある．また符号化出力した命題式は巨大なものになることも多く，メモリオーバなどの問題に直面することも多い．

これに対して遅延処理型 SMT 技術は，個別の理論に特化した既存ソルバと最新 SAT ソルバ技術を効果的に組み合わせる目的で発展してきた技術である．

この手法では，充足可能性を判定する一階論理式は，まず命題式として取り扱われ，SAT ソルバで処理される．もし SAT ソルバで充足不可能と判定されれば，処理はそこで終了する．充足可能と判定された場合，SAT ソルバから命題論理モデルが出力されるので，一階論理に復元され, 背景理論の専用ソルバが起動し，背景理論との無矛盾性を調べる．無矛盾であると判定されれば，そこで処理が終了する．矛盾していれば，SAT ソルバが別の命題論理モデルの探索を再開し，再度同じことを繰り返す．柔軟性に富む枠組みであり，全体として高速といわれている．

本論文では，遅延処理型 SMT 技術に関して議論していく．

3.1 単一背景理論に対する素朴な SMT ソルバ

ここでは, まず, 単一背景理論に対する素朴な SMT ソルバについて簡単に説明する．この SMT ソルバには Nelson-Oppen 法は組み込まれていないため, 単一の背景理論からなる式の充足可能性しか判定することができない. SMT ソルバのアルゴリズムを図 1 に, その流れを図 2 に示す.

このような SMT ソルバで, ソースコードの公開されているものとして KS-SMT ソルバ [3] があり, 現在この SMT ソルバの改造を計画している.

図 1 のアルゴリズムの動作例を示す. 以下の一階等号理論からなる式の充足可能性を判定する場合を考える.

$x = y \wedge y = z \rightarrow x = z$

まず，SMT ソルバでは，入力式を命題式に変換する．その結果が以下のようになる (1 行目).

$A \wedge B \rightarrow C$

上の式の A は $x = y$ を命題変数に抽象化したもの，同様に B は $y = z$，C は $x = z$ を命題変数に抽象化したものである

```
function Bool+T (φ: quantifier-free formula)
1    φ^p ⟵ fol2prop(φ);
2    A^p ⟵ fol2prop(Atoms(φ));
3    while Bool-satisfiable(φ^p) do
4        β^p ⟵ pick_total_assign(A^p, φ^p);
5        (ρ, π)⟵ T-satisfiable(prop2fol(β^p));
6        if (ρ == sat) then return sat;
7        φ^p ⟵ φ^p ∧ ¬fol2prop(π);
8    end while;
9    return unsat;
end function
```

図 1: 単一の背景理論に対する素朴な SMT ソルバのアルゴリズム [3]

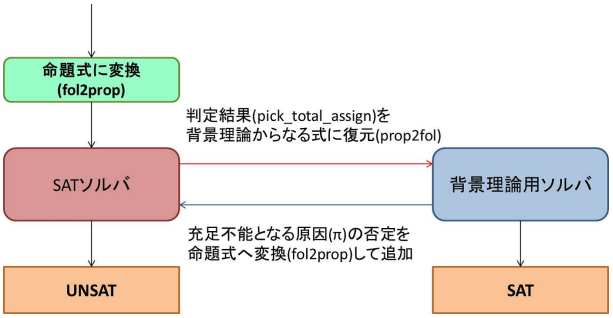

図 2: KS-SMT ソルバの概要

(2 行目). 次に命題式を SAT ソルバで判定するために，連言標準形に変換する．変換した式は以下のようになる．

$\neg A \vee \neg B \vee C$

変換した式を SAT ソルバに渡して充足可能性を判定させる (3 行目). その結果 $A : True, B : False, C : True$ の割り当てで充足可能であると判定された．次に今回の割り当てをもとに，背景理論の式を復元する．その結果が以下のようになる (4 行目).

$x = y \wedge y \neq z \wedge x = z$

A，C には $True$ が割り当てられたので，そのまま $x = y$ と $x = z$ に，B には $False$ が割り当てられたので，$y \neq z$ に復元している．

次に復元した式に専用ソルバを適用し，背景理論との無矛盾性を調べる (5 行目). その結果，上の式は左右の節と真ん中の節において矛盾していることがわかる．そのため，充足不能となる原因の否定である $\neg(A \wedge \neg B \wedge C)$, すなわち $\neg A \vee B \vee C$ を式に追加し (7 行目), 再度 SAT ソルバに探索させる (3 行目). その結果 $A : True, B : True, C : True$ の割り当てで充足可能であると判定されたため，先ほどと同様に今回の割り当てをもとに背景理論の式を復元する (4 行目).

$x = y \wedge y = z \wedge x = z$

この式に，専用ソルバを適用し，背景理論との無矛盾性を調べる (5 行目). その結果，今回の式では充足可能であると判定され，今回入力された式は，充足可能であると判定される (6 行目).

このようにして SMT ソルバは単一の背景理論からなる式の充足可能性を判定する．

しかし, 実用問題では複数の背景理論が組み合わされている．そのような複合型の背景理論に対しての技術として Nelson-Oppen 法が開発された．

3.2 SMT 背景理論の決定手続きの結合法:Nelson-Oppen 法

SMT の実用問題で必要となる大規模で複雑な背景理論は，複数の背景理論が組み合わされて構成されている．ここの背景理論に対しては, 長年の研究に基づく非常に精密で高度な決定手続きとそれを実装した複雑なプログラム (専用ソルバ) が用意されている. そのような複合型理論に対して新たに専用ソルバを開発するのは大変な苦労が伴う．そのため，個々の背景理論に対して開発された既存の専用ソルバをそのまま組み合わせて利用する数学的枠組みとして Nelson-Oppen 法が開発されている．この手法は，30 年以上前に提案されたものであるが，これを本質的に超えるものはまだ存在していない．

Nelson-Oppen 法は, 等号以外は互いに共通記号を持たない背景理論 T_1, T_2 に対して, その合併理論 $T_1 \wedge T_2$ を考え, 基礎式 F が与えられたときに, F が $(T_1 \wedge T_2)$-充足可能か否かを新たに $(T_1 \wedge T_2)$-ソルバを作らずに, 既存の T_1-ソルバと T_2-ソルバだけを使って判定を可能にするものである [4]. この手法は次の 2 つのステップからなる手続きである．

1. **純化ステップ**

   複数の背景理論からなる式をそれぞれ単一の背景理論からなる複数の式に分割する．

2. **判定ステップ**

   分割した各式に既存の専用ソルバを用いて充足可能性を判定する．すべての式が充足可能と判定されれば，元の式は充足可能と判定する．

次に素朴な Nelson-Oppen 法の動作例を示す．以下の未定関数記号をもつ等式理論 (T_E) と算術理論 (T_Z) からなる式 μ の充足可能性を判定する．

$$\mu : 1 \leq a \wedge a \leq 2 \wedge f(a) \neq f(1) \wedge f(a) \neq f(2)$$

まず，純化ステップを行って単一の背景理論からなる式に分割する．その際，未定関数記号は T_E で扱われ，定数は T_Z で扱われるため，それらが同じ項に存在する $f(1)$ と $f(2)$ は式を分割する際に都合が悪い．そこで，定数を変数で置き換えて式を抽象化する．抽象化を行った式は以下のようになる．

$$\mu : 1 \leq a \wedge a \leq 2 \wedge f(a) \neq f(w_1) \wedge f(a) \neq f(w_2)$$
$$\wedge w_1 = 1 \wedge w_2 = 2$$

抽象化を行うことで純化を行いやすくなる．次に，式を単一の背景理論からなる式に分割する．分割した式を以下に示す．T_E 上の式を μ_E，T_Z 上の式を μ_Z と表す．

$$\mu_E : f(a) \neq f(w_1) \wedge f(a) \neq f(w_2)$$
$$\mu_Z : 1 \leq a \wedge a \leq 2 \wedge w_1 = 1 \wedge w_2 = 2$$

次に，判定ステップとして分割した式にそれぞれの背景理論を扱う専用ソルバを用いて充足可能性の判定を行う．その際，分割した式の間で矛盾が起こらないようにするために，それらの式に共通して出現する変数 (a, w_1, w_2) について制約条件を作成する．

制約条件は, 分割式のいずれかの論理的帰結から構築し, その条件のもとですべての式が充足可能か否かを判定する．いずれかの制約条件でもとの式が充足可能であれば, もとの式は充足可能である．

この流れを判定木を用いて表すと図 3 に示すようになる．

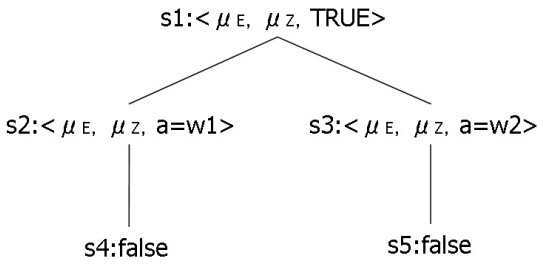

図 3: 判定木

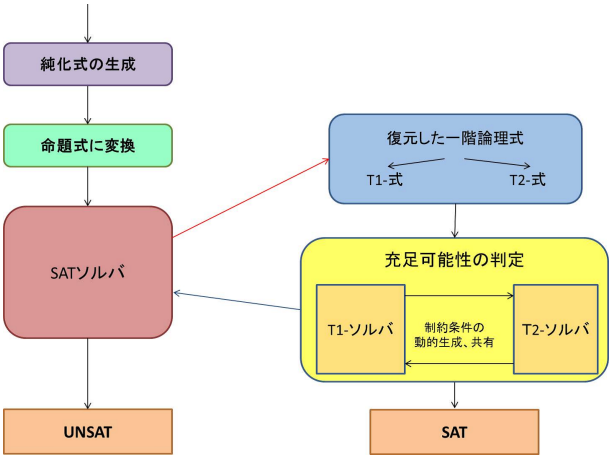

図 4: Nelson-Oppen 法の概要

s_2, s_3 は $\mu_Z \models a = w_1 \vee a = w_2$ より導出され, 制約条件は $a = w_1$ と $a = w_2$ の 2 つとなる. いずれかの条件で μ_E と μ_Z が充足可能であれば良い. しかし, s_4 は $\mu_E \wedge a = w_1$ が T_E-充足不能より, s_5 は $\mu_E \wedge a = w_2$ が T_E-充足不能より導出される. すべての葉が充足不能となったため, 元の式 μ は充足不能と判定される.

以上のように Nelson-Oppen 法では, 問題式の充足可能性を判定する.

Nelson-Oppen 法は, 図 2 の背景理論ソルバを図 4 のように各背景理論用ソルバに置き換え, それらの間の制約条件を動的に作成, 共有するように変更するメカニズムを付加することで実装できる. しかし, このような素朴な Nelson-Oppen 法の実装には問題点がいくつか存在する. その問題点と改善案について次章で紹介する.

4. Delayed Theory Combination とその実装

素朴な Nelson-Oppen 法では, 図 4 に示したように, 問題式の命題充足可能性を判定する. その後, 共通変数の制約条件を背景理論から推論し, 他の背景理論とその情報を共有しながら充足可能性を判定木を作成しながら調べていく. そのために背景理論用ソルバが制約条件の情報を共有する必要がある. しかし, 既存の背景理論用ソルバは極めて精巧に作られており, 制約条件を共有させるための改造は非常に難しいため, 各背景理論用ソルバは改造せずにそのまま利用したい. また, 制約条件を, 実行中に動的に生成することは手間がかかるので, 可能であれば, 回避したい. 更に命題充足可能性を探索するための木のほかに Nelson-Oppen 法を実行するための判定木を別に作ることもオーバーヘッドが大きいので, これも可能であるならば避けたい. そのための改良案として **Delayed Theory Combination(DTC)**[5] が開発された. DTC では, あらかじめ共通変数の制約条件すべてを用意し, 命題充足可能性を判定する際に変数間の関係 (制約条件) も決めてしまう. 問題式を各背景理論の式に分割する際に, 予め決めてあった制約条件も一緒にソルバに渡し, 複数の背景理論 (ソルバ) 間の矛盾を防止する.

以下の図 5 に DTC のアルゴリズムを示す.

function Bool+T_1+T_2 (ϕ: *quantifier-free formula*)
1 $\phi \longleftarrow purify(\phi)$;
2 $\phi^p \longleftarrow fol2prop(\phi)$; $\mathcal{A}^p \longleftarrow fol2prop(Atoms(\phi) \cup IE(\phi))$;
3 **while** Bool-*satisfiable* (ϕ^p) **do**
4 $\beta_1^p \wedge \beta_2^p \wedge \beta_e^p = \beta^p \longleftarrow pick\_total\_assign(\mathcal{A}^p, \phi^p)$;
5 $(\rho_1, \pi_1) \longleftarrow T_1$-*satisfiable* ($prop2fol(\beta_1^p \wedge \beta_e^p)$);
6 $(\rho_2, \pi_2) \longleftarrow T_2$-*satisfiable* ($prop2fol(\beta_2^p \wedge \beta_e^p)$);
7 **if** ($\rho_1 ==$ sat **and** $\rho_2 ==$ sat) **then return** sat;
8 **if** ($\rho_1 ==$ unsat) **then** $\phi^p \longleftarrow \phi^p \wedge \neg fol2prop(\pi_1)$;
9 **if** ($\rho_2 ==$ unsat) **then** $\phi^p \longleftarrow \phi^p \wedge \neg fol2prop(\pi_2)$;
10 **end while**;
11 **return** unsat;
end function

図 5: DTC を用いた 2 つの背景理論に対する SMT ソルバのアルゴリズム [5]

DTC は, 図 6 のように予めすべて制約条件を作成しておくことで, 一階論理式の充足可能性を判定する際に, 各背景理論用ソルバを完全に独立して扱うことができる.

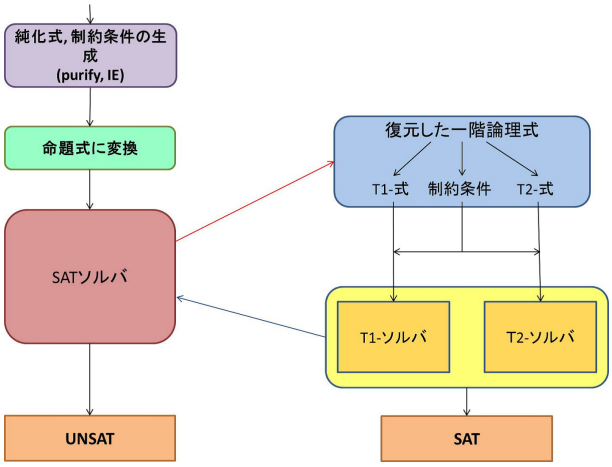

図 6: DTC の概要

次に, このアルゴリズムの動作例を示す. Nelson-Oppen 法の動作例と同じ以下の式の充足可能性を判定する.

$$\mu : 1 \leq a \wedge a \leq 2 \wedge f(a) \neq f(1) \wedge f(a) \neq f(2)$$

さきほどと同様に純化を行う (1 行目).

$$\mu : 1 \leq a \wedge a \leq 2 \wedge f(a) \neq f(w_1) \wedge f(a) \neq f(w_2) \wedge w_1 = 1 \wedge w_2 = 2$$

制約条件の集合を作成し, 原子論理式と併せて命題化する. ここでは

$$S = \{1 \leq a, a \leq 2, f(a) = f(w_1), f(a) = f(w_2), w_1 = 1, w_2 = 2, a = w_1, a = w_2, w_1 = w_2\}$$

を

$$S^p = \{A, B, C, D, E, F, G, H, I\}$$

のように命題化する.

次に, 式 μ の命題充足可能性を判定する. そのため, 式の各項を先ほど対応付けた命題変数で置き換える (2 行目).

$$\mu^p : A \wedge B \wedge \neg C \wedge \neg D \wedge E \wedge F$$

この式を SAT ソルバを用いて判定する (3 行目). 充足不能と判定されれば元の式は充足不可能であり, 充足可能と判定されれば, 以下の処理を行う.

今回, C, D に $False$ が割り当てられ, それ以外に $True$ が割り当てられたとする. この割り当てをもとに, 一階論理式へ復元し, 各背景理論からなる式と共通変数への制約に分割する (4 行目).

$$\mu_E : f(a) \neq f(w_1) \wedge f(a) \neq f(w_2)$$
$$\mu_Z : 1 \leq a \wedge a \leq 2 \wedge w_1 = 1 \wedge w_2 = 2$$
$$e : a = w_1 \wedge a = w_2 \wedge w_1 = w_2$$

これを各背景理論用のソルバに制約条件と一緒に渡し, 充足可能性を判定する (5,6 行目). T_Eソルバ には, $\mu_E \wedge e$ を, T_Z ソルバ には, $\mu_z \wedge e$ を渡して判定する.

その結果, 上の条件では充足不能であると判定される. その際, 充足不能の原因として T_Eソルバ から $f(a) \neq f(w_1) \wedge a = w_1$ と $f(a) \neq f(w_2) \wedge a = w_2$ が, T_Zソルバ から $w_1 = 1 \wedge w_2 = 2 \wedge w_1 = w_2$ が得られる. 次に, その否定を式に付加し, もう一度同じ動作を行う (8.9 行目).

$$\mu^p : A \wedge B \wedge \neg C \wedge \neg D \wedge E \wedge F \wedge \neg(\neg C \wedge G) \wedge \neg(\neg D \wedge H) \wedge \neg(E \wedge F \wedge I)$$

今度は SAT ソルバが C, D, G, H, I に $False$, それ以外に $True$ を割り当てたとする. この割り当てをもとに再度一階論理式へ復元し, 各背景理論からなる式と制約条件に分割する (4 行目).

$$\mu_E : f(a) \neq f(w_1) \wedge f(a) \neq f(w_2)$$
$$\mu_Z : 1 \leq a \wedge a \leq 2 \wedge w_1 = 1 \wedge w_2 = 2$$
$$e : a \neq w_1 \wedge a \neq w_2 \wedge w_1 \neq w_2$$

先ほどと同様に各背景理論用のソルバに渡し, 充足可能性を判定する (5,6 行目). その結果, 今回も充足不能であると判定される. その際, 充足不能の原因として T_Zソルバ から $1 \leq a \wedge a \leq 2 \wedge w_1 \leq 1 \wedge w_2 \leq \wedge a \neq w_1 \wedge a \neq w_2$ が得られる. そのため, その否定を式に付加し, もう一度同じ動作を繰り返す (8,9 行目).

$$\mu^p : A \wedge B \wedge \neg C \wedge \neg D \wedge E \wedge F \wedge \neg(\neg C \wedge G) \wedge \neg(\neg D \wedge H) \wedge \neg(E \wedge F \wedge I) \wedge \neg(A \wedge B \wedge E \wedge F \wedge \neg G \wedge \neg H)$$

今度は SAT ソルバが上の式は充足不可能であると返すので, もとの式 μ は充足不能であると判定される (11 行目).

このように DTC では, あらかじめ制約条件を作成することで, 各ソルバを完全に独立して実行させることができる.

DTC は, SMT ソルバの世界競技会である SMT-COMP で上位に入選したソルバにも使われている手法である. しかし DTC を組み込んだソースコードが公開されている SMT ソルバは存在しない. そのため, まずは公開されていた単一の背景理論用ソルバである KS-SMT ソルバ [3] に自前で DTC を組み込みを計画し, 準備作業している.

5. まとめと今後の課題

Nelson-Oppen 法などの背景理論の決定手続き結合法は, SMT ソルバの主要技術の 1 つである. そのため, 本論文では Nelson-Oppen 法と, その実装技術である Delay Theory Combination の概要を示した. しかし, ソースコードが公開されている Nelson-Oppen 法を組み込んだ SMT ソルバが存在しない. そこで, 現在 DTC をソースコードが公開されている KS-SMT ソルバに実装している. 性能評価などについては, 改めて発表をしたいと考えている.

謝辞

本研究は一部, 文科省科学研究費補助金（基盤 C：No.22500127）の援助を受けている.

参考文献

[1] Nelson, G. and Oppen, D, C.: Simplification by Cooperating Decision Procedures, ACM Transaction on Prograannubg Languages and systems, Volume 1, 2, pp.245-257(1979)

[2] 岩沼宏治.: 共通記号を持つ背景理論の決定手続きの結合法とその効率化について (修正版), 信学技報, vol.109, no.456, pp.115-120(2010)

[3] Daniel Kroening, Ofer Strichman.: Decision Procedures An Algorithmic Point of View, Springer-Verlag New York Inc(2010)

[4] 岩沼宏治. 鍋島英知.: SMT:個別理論を取り扱う SAT 技術, 人工知能学会誌, 25 巻 1 号, pp.86-95(2010)

[5] Marco Bozzano, Roberto Bruttomesso, Alessandro Cimatti, Tommi Junttila, Silvio Ranise, Peter van Rossum, Roberto Sebastiani.: Efficient theory combination via boolean search, Information and Computation, Volume 204, pp 1493-1525(2006)