Vol.2021-ICS-204 No.13
2021/9/16

ADSAT：敵対者が存在するMaxSAT

菅原 知也[1,,a)]　越村 三幸[1,]　横尾 真[1,]

概要：本論文では，ADSAT (Attacker Defender SAT) と呼ばれる，防御者と敵対者が存在する場合のMaxSAT について論じる．MaxSAT は，充足可能性問題 (SAT) を最適化問題に拡張したものであり，その目的は充足する節の数の最大化である．ADSAT において，防御者は充足する節の数の最大化を目的とし，敵対者はその最小化を目的とする．本研究の目的は，敵対者のあらゆる攻撃を想定した防御者の最適解を求めることである．このような頑健な解を求めることは理論上のみならず実用上も重要である．先行研究において，ADSAT を解くアルゴリズムとして IBR (Iterated Best Response) が提案されている．しかしながら，ADSAT は Σ_2^P 完全問題であるため，小さいサイズの問題でも実用的な時間で最適解を求めることが非常に困難である．そこで本論文では，ADSAT の近似解を求めるアルゴリズムとして，IBR を簡略化した BP-BR (Best Preference Best Response) と AE (Attack Enumeration) の 2 種を提案する．特に敵対者の攻撃を限定した場合，AE は効率的に近似解を求めることができる．また，計算機実験により 3 つのアルゴリズムの性能を比較し，評価を行う．

1. 序論

充足可能性問題 (SAT, Boolean Satisfiability Problem) は，ある命題論理式が与えられたとき，その論理式を充足するブール変数の割当の有無を判定する問題である．SAT は NP 完全と呼ばれる，現実的な時間では一般的に解けない問題のクラスに属する [1]．問題のクラスとは，同等の複雑さを持つ問題の集合であり，クラス NP は，解の候補が与えられた場合にその解の正しさを多項式時間で判定できる問題のクラスである．NP 完全問題は，クラス NP の中で，NP に属する他の問題から多項式時間で還元可能な決定問題を指す．SAT のように計算量の観点から厳密解を求めることが困難な問題に関する研究はこれまで盛んに行われている [2], [3]．一方，充足最大化問題 (MaxSAT, Maximum Satisfiability Problem) は，SAT を最適化問題に拡張したものであり [4], [5]，SAT が全ての節を充足する割当を探索する問題であるのに対して，MaxSAT は充足する節の数が最大となる割当を探索する問題である．MaxSAT は NP 困難と呼ばれる，NP に属する任意の問題と同等かそれ以上に難しい問題のクラスに属する．

本論文では，敵対者と呼ばれる，変数を反転することが可能なプレイヤーが存在する場合の充足最大化問題である ADSAT (Attacker Defender SAT) について考察する．ADSAT では，攻撃側と防御側の 2 つの立場が存在している．攻撃側は m 個の変数を反転することで充足する節の数を小さくすることを試みる一方，防御側は充足する節の数の最大化を試みる．攻撃側がいかなる攻撃を行ったとしても充足する節の数が小さくなりにくい，頑健な変数の割当を防御側の立場で求めることが本研究の大きな目的である．ADSAT は，攻撃側と防御側双方の戦略の数が変数の数に対して指数関数的に増加していくため，Σ_2^P 完全問題に分類される．クラス Σ_2^P は，任意のクラス NP に属する問題を定数時間で解くことが可能な手法（オラクル）が存在するという仮定の下で，解の候補が与えられた場合にその解の正しさを多項式時間で判定できる問題のクラスである [6]．Σ_2^P 完全問題は NP 完全問題と比較して解くことが極めて困難とされている．

Σ_2^P 完全問題である ADSAT の厳密解を求める過程で，近似解を有限時間内に効率良く探索するアルゴリズムとして，IBR (Iterated Best Response) アルゴリズムが知られている [7]．IBR は防御側の割当を先に求め，攻撃側と防御側が最適な戦略をお互いに求めることで充足する節の数の上限と下限，下限を与える頑健な変数の割当を求めるアルゴリズムである．しかしながら，IBR ではアルゴリズム中で類似の問題を繰り返し解くため，値の更新が起こりにくいという問題点が存在した．そこで，本論文では ADSAT の近似解を求めるアルゴリズムとして BP-BR (Best Preference Best Response) アルゴリズムと，AE (Attack Enumeration) アルゴリズムを提案する．BP-BR は，アル

[1] 九州大学 大学院システム情報科学府
819- 0395 福岡県福岡市西区元岡 744 番地, (092)802-3576
[a)] sugaharat@agent.inf.kyushu-u.ac.jp

ゴリズム中の繰り返し部分を除くことで IBR における問題点を愚直に解消したアルゴリズムであり，IBR と比較して性能が向上することが確認されている．AE は，攻撃の組合せを次々と変化させることで値の更新が起こりやすくなるという考えをもとに，初めに攻撃側の立場で攻撃の組合せを決定し，防御側の立場で頑健な変数の割当を求め，さらにその割当に対し攻撃の組合せを求めることで上限と下限，頑健な変数の割当を求める．AE は攻撃側が高々定数個の変数しか攻撃できない状況において，IBR と比較して大幅に計算量を削減しながら値の更新回数を増やすことに成功している．本論文の最後に 3 つのアルゴリズムの性能を計算機実験によって評価し，エラー率の観点で AE が優れていることを示す．

2. 準備

本章では，ADSAT (Attacker Defender SAT) に関する基本的な用語の説明のため，SAT (Boolean Satisfiability Problem) と MaxSAT (Maximum Satisfiability Problem) について例を挙げて説明した後，ADSAT の定義を示す．

2.1 SAT

本節では SAT で用いられる用語を説明する．

- ブール変数：真偽値が割当てられる変数.
- リテラル：論理式に含まれるブール変数. またはその否定.
- 節：1 つ以上のリテラルの選言（論理和による結合）.
- 連言標準形：1 つ以上の節の連言（論理積による結合）.

充足可能性問題 (SAT) は，連言標準形 (CNF) の命題論理式が与えられたとき，その論理式を充足するブール変数の割当の有無を判定する問題である．与えられた論理式が充足することを充足可能，充足しないことを充足不能と呼ぶ．例として，以下の CNF が与えられた場合を考える．

(1) $(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_4 \vee \overline{x}_1 \vee x_5) \wedge (\overline{x}_6 \vee \overline{x}_1 \vee \overline{x}_3)$
(2) $(x_1 \vee x_2) \wedge (\overline{x}_1 \vee x_2) \wedge (x_1 \vee \overline{x}_2) \wedge (\overline{x}_1 \vee \overline{x}_2)$

(1) を入力とすると，全ての節を充足する割当の 1 つとして $(x_1, x_2, x_3, x_4, x_5, x_6) = (T, T, T, T, T, F)$ が挙げられる．このとき，(1) の入力は充足可能である．

(2) を入力として考えると, 考え得る全ての割当 $(x_1, x_2) = \{(T, T), (T, F), (F, T), (F, F)\}$ において 4 つ全ての節を充足する組合せは存在しないため充足不能である．

SAT の定義を以下に示す．

定義 1 (SAT) 連言標準形 (CNF) の命題論理式が与えられたとき，その命題論理式が充足可能か充足不能か判定する問題.

次に，SAT の拡張である充足最大化問題 (MaxSAT) を説明する．MaxSAT は，CNF の入力に対して充足する節の数を最大化する割当を求める問題である．SAT とは異なり，全ての節が充足しない場合も解は存在する点が大きな違いである．先ほどの例を用いて MaxSAT の実際の解を示す．

(1) を入力として考えると，SAT 解の 1 つとして $(x_1, x_2, x_3, x_4, x_5, x_6) = (T, T, T, T, T, F)$ を挙げた．全ての節を充足するとき，充足する節の数は最大となるためこの解は MaxSAT 解でもある．SAT 解が存在する場合，その解は MaxSAT 解でもある．

(2) を入力として考えると，(2) の論理式は充足不能なため，4 つの節を充足することはできない．しかしながら，$(x_1, x_2) = (T, T)$ などの割当において最大で 3 つの節を充足することが可能である．したがって，解の 1 つは $(x_1, x_2) = (T, T)$ となる．(2) のように SAT 解が存在しない場合でも，MaxSAT 解は存在する．

MaxSAT の定義を以下に示す．

定義 2 (MaxSAT) 連言標準形 (CNF) の命題論理式が与えられたとき，その命題論理式に含まれる節の数を最大化する割当を求める問題.

2.2 ADSAT

本節では，敵対者が存在する場合の MaxSAT として，ADSAT (Attacker Defendere SAT) を説明する．ADSAT は，通常の MaxSAT に敵対者が存在する場合を考えたもので，敵対者は最大 m 個のブール変数に対する割当を反転することで充足する節の数を減少させることができる．敵対者の攻撃を受けても充足する節の数が大きく減少しない，頑健な割当を発見することが本問題の目的である．ADSAT では，充足する節の数を減らそうとする敵対者を攻撃側，充足する節の数を増やそうとする側を防御側と呼ぶ．

ADSAT$\langle X, C, m, \theta \rangle$ は，

- $X = \{x_1, \ldots, x_n\}$ をブール変数の集合とする．
- $C = \{c_1, \ldots, c_\ell\}$ を X の選言の集合とする．
- $m \in N$ を攻撃側が反転可能な変数の個数の最大値とする．
- $\theta \in N$ を充足する節の数に関する閾値とする．

で構成される．

$\tau : X \to \{0, 1\}$ により，X に含まれる変数 $x_1, \ldots x_n$ の真偽の割当を表す．$s(\tau)$ は 割当 τ により充足する C の節の数を表す．攻撃側が反転する高々 m 個の変数の集合を $X'(|X'| \leq m, X' \subseteq X)$ とする．$\tau_{X'}$ を，ある割当 τ に対する X' による攻撃後の変数の割当とする．攻撃側は変数の反転（攻撃）により充足する節の数を可能な限り小さくすることを目的としており，攻撃側が最良の攻撃を行った後の攻撃後の充足する節の数を $s_m(\tau)$ で表す．すなわち，

$$s_m(\tau) = \min_{X' \subseteq X, |X'| \leq m} s(\tau_{X'})$$

となる．また，反転後に充足する節の数を最も大きくする割当，すなわち防御側にとって最適な割当を $\hat{\tau}$ で表し，

$$\hat{\tau} = \arg \max_{\tau' \in a(X \setminus X')} s_m(\tau')$$

となる．ここで，$a(X)$ は変数の集合 X に対して取り得る割当の集合である．

ADSAT を定義する前に，ADSAT-verif を定義する．

定義 3 (ADSAT-verif) インスタンス $\langle X, C, m \rangle$ とある割当 τ が与えられたとき，$\tau_{X'}$ が高々 θ 以下の節を充足する攻撃の組合せ $X'(X' \subseteq X, |X'| \leq m)$ は存在するか判定する問題．

ADSAT-verif は，ある 1 つの割当 τ に対する判定問題である．Watanabe らにより以下の定理が証明されている [7]．

定理 1 ADSAT-verif は NP 完全問題である．

次に，ADSAT の定義を以下に示す．

定義 4 (ADSAT (最適化問題)) インスタンス $\langle X, C, m \rangle$ が与えられたとき，任意の攻撃の組合せ $X'(X' \subseteq X, |X'| \leq m)$ に対して

$$\tau^* = \arg \max_{\tau' \in a(X \setminus X')} s_m(\tau')$$

を満たす割当 τ^* を求める問題．

ADSAT を判定問題として捉えると，以下のようになる．

定義 5 (ADSAT (判定問題)) インスタンス $\langle X, C, m, \theta \rangle$ が与えられたとき，任意の攻撃の組合せ $X'(X' \subseteq X, |X'| \leq m)$ に対して $\tau_{X'}$ が少なくとも θ 以上の節を充足する割当 τ は存在するか判定する問題．

ADSAT を解くには全ての割当に対する ADSAT-verif を解く必要がある．割当の総数は 2^n であるため，直感的には，指数関数的に増加する NP 完全問題を解かなければならないことを意味する．Watanabe らにより，以下の定理が証明されている [7]．

定理 2 ADSAT は Σ_2^P 完全問題である．

3. 提案アルゴリズム

本章では，ADSAT の解を求める方法として従来の IBR (Iterated Best Response) アルゴリズムを説明した後，新たに BP-BR (Best Preference Best Response), AE (Attack Enumeration) アルゴリズムを提案する．

3.1 IBR アルゴリズム

本節では，従来のアルゴリズムである IBR を説明する．初めに，IBR で用いる上限と下限の定義を示す．

定義 6 (下限 (lb)) ある割当 τ に対して，いかなる攻撃を受けても保証される充足する節の数．

定義 7 (上限 (IBR, AE) (ub)) ある攻撃の組合せ X' において，あらゆる割当に対して保証される充足する節の数．

ADSAT における最適解は，すべての割当に対する下限の最大値をとるときの割当である．しかしながら，Σ_2^P 完全問題である ADSAT における最適解を求めることは現実時間内では不可能である．最適解の存在し得る範囲を求めるため，なるべく大きな下限の値と小さな上限の値を探索することが現実的に可能なアプローチである．

IBR では，初めにある割当 τ に対して保証される下限 lb と τ に対する最良の攻撃 X' を探索した後，X' に対して保証される上限 ub と最適な割当 $\hat{\tau}$ を求める．上限または下限が更新された場合は割当 $\hat{\tau}$ に対して再び同様の操作を繰り返す．上限または下限の更新が無い場合，別の割当に対して同様の操作を行うことで上限と下限の値を更新し続け，上限と下限が一致するか全ての割当に対する探索を終えると IBR は終了する．

IBR は以下で定義される．

定義 8 (IBR アルゴリズム) 入力 $\langle X, C, m \rangle$ に対して以下の操作を行う．

(1) $ub = 0, lb = \infty$ として初期化を行う．
(2) まだ探索していない割当の中で充足する節数が最大となる割当 τ に関して，攻撃側は充足する節数が最小となる攻撃の組合せ $X' \subseteq X$（反転するブール変数の組合せ）を探索する．$s(\tau_{X'}) > lb$ の場合，下限 lb を更新する．
(3) 2 で求めた攻撃 X' に対して，防御側は攻撃を受ける変数の値は固定し，それ以外の変数に関して充足する節数が最大となる割当 $\hat{\tau} \in a(X \setminus X')$ を探索する．$s(\hat{\tau}) < ub$ の場合， 上限 ub を更新する．
(4) 上限または下限が更新されていなければ他の新たな割当を τ とする．更新されていればその割当 $\hat{\tau}$ を新たに τ とする．
(5) 2〜4 を繰り返し行い，上限と下限が一致するか全ての割当に対して探索を終えた場合，アルゴリズムは終了し，下限を与える割当 τ, 上限 ub, 下限 lb を出力する．

また，IBR は anytime アルゴリズムであるため，実行中，

$$lb \leq s_m(\tau^*) \leq ub$$

を常に満たすことが証明されている [7]．すなわち，IBR を実行途中で中断しても，上限と下限，下限を与える割当を出力することが可能である．

3.2 BP-BR アルゴリズム

本節では，1 つ目の新規アルゴリズムで，IBR を簡略化した BP-BR を説明する．BP-BR は IBR における防御側の探索を省略し，次のループに進む際の割当を確実に変更する，IBR を簡略化したアルゴリズムである．BP-BR における上限の定義は IBR と異なり，以下となる．

定義 9 (上限 (BP-BR) (ub)) 攻撃側の行動を考慮せず探索した割当の中で，充足させることが可能な節の数の最大値．

BP-BR は以下で定義される．

定義 10 (BP-BR アルゴリズム) 入力 $\langle X, C, m\rangle$ に対して以下の操作を行う．

(1) $ub = 0, lb = \infty$ として初期化を行う．
(2) まだ探索していない割当の中から，充足する節数が最大となる割当 τ を見つけ，上限 ub を更新する．
(3) 割当 τ に対して，攻撃側は充足する節数が最小となる攻撃の組合せ $X' \subseteq X$ を探索する．$s(\tau_{X'}) > lb$ の場合，下限 lb を更新する．
(4) 2〜3 を繰り返し行い，上限と下限が一致するか全ての割当に対して探索を終えた場合，アルゴリズムは終了し，下限を与える割当 τ, 上限 ub, 下限 lb を出力する．

3.3 AE アルゴリズム

本節では，2 つ目の新規アルゴリズムである AE を説明する．AE では，初めにある攻撃の組合せ X' に対して保証される上限 ub と最適な割当 τ を探索した後，τ によって保証される下限 lb を求める．上限や下限の更新に関係なく，別の攻撃の組合せに対して同様の操作を繰り返すことで上限と下限の値を更新し続け，上限と下限が一致するか全ての攻撃の組合せに対する探索を終えると AE は終了する．また，全ての攻撃の組合せを探索するため，AE が終了したとき上限は確定する．

AE は以下で定義される．

定義 11 (AE アルゴリズム) 入力 $\langle X, C, m\rangle$ に対して以下の操作を行う．

(1) $ub = 0, lb = \infty$ として初期化を行う．
(2) ある攻撃の組合せ $X' \subseteq X$ に関して，防御側は充足する節数が最大となる割当 $\tau \in a(X \setminus X')$ を探索する．$s(\tau) < ub$ の場合， 上限 ub を更新する．
(3) 2 で求めた割当 τ に対して，攻撃側は充足する節が最小になる攻撃の組合せ $X'' \subseteq X$（反転するブール変数の組合せ）を探索する．$s(\tau_{X''}) > lb$ の場合，下限 lb を更新する．
(4) 2 と 3 を繰り返し行い，上限と下限が一致するか全ての攻撃の組合せに対して探索が終了した場合，アルゴリズムは終了し，下限を与える割当 τ, 上限 ub, 下限 lb を出力する．

IBR と同様に，BP-BR と AE も anytime アルゴリズムであるため，実行中，

$$lb \leq s_m(\tau^*) \leq ub$$

を常に満たすことが証明できる．したがって，各アルゴリズムを実行途中に中断したとしても，上限と下限，下限を与える割当を出力することが可能である．

4. 計算機実験

本章では，従来の IBR アルゴリズムと新たに提案した 2 つのアルゴリズムを用いて ADSAT の問題に対して出力される上限と下限を比較し，性能を評価する．実験の問題設定は，変数の数 $n = 20, 30$，各節のリテラルの数 $k = 3$，節の数 $\ell = 25, 50, 75, 100, 125, 150$ とし，ランダムに生成された問題 10 インスタンスに対して Intel core i7-6700X CPU @4.00GHz プロセッサと 32GB メモリを搭載した Windows 10 Education 64bit Edition マシンを用いて実験を行った．また，アルゴリズム中に用いた SAT ソルバーは PySAT[8] である．変数の数が 20 個の場合，攻撃の組合せの数が ${}_{20}C_2 = 190$ となり，攻撃総数は ${}_{20}C_2 \cdot 2^2 = 720$ である．本実験の問題設定では 5 分以内に全ての攻撃を試行することが可能であるため，AE によって防御側の上限を与える割当は確実に求まる．また，IBR と BP-BR の実行時間は AE の全ての攻撃の組合せの実行時間に合わせて同じ時間で行った．

表 1 に $n = 20$ の場合において各アルゴリズムにより得られた上限と下限の平均値を示す．各アルゴリズムにより出力された ub (上限) と lb (下限) の値を比較すると，下限の値はほぼ同程度であるのに対し，上限の値は AE が最も小さな値を，次点で BP-BR が小さな値をとっており，IBR は最も大きな値をとっている．IBR は上限と下限の両方を更新しながら二重のループを繰り返すのに対し，BP-BR はループを行わずに別の割当に即座に移行することで不要なループが削減され，より小さな上限の値を出力したと考えられる．AE は攻撃の組合せ順に上限を先に求めた後にその割当に対する下限を求めるだけで次のループに入るため，上限を優先して確実に求めることが可能であり，最も小さな上限の値を出力したと考えられる．

図 1 は，節の数に対するエラー率を表す．エラー率とは，$(ub - lb)/lb$ で表される値であり，値が小さいほど望ましい．$n = 20$ の場合を見ると，IBR ではエラー率は 5〜10% 程度の値をとるのに対して AE ではエラー率は 3〜7%の間に収まっており，常に AE のエラー率の方が低い値を示している．BP-BR は IBR と AE の中間といえる値をとっており，節の数が小さなときは IBR と，大きなときは AE と同程度の値をとった．$n = 30$ の場合も同様の傾向を示しており，節の数がいずれの場合においても AE は低いエラー率を示していることから，今回の実験における AE の性能は最も優れているといえる．

5. 結論

本論文では Σ_2^P 完全問題である ADSAT を扱い，有限時間内に近似解を求めるアルゴリズムとして，IBR を簡略化した BP-BR アルゴリズムと，AE アルゴリズムを提案した．ADSAT は複数の変数を反転することが可能な敵対者が存在する場合の MaxSAT であり，攻撃を受けた後も充足する節の数が大きくなる頑健な割当を求めることが目的

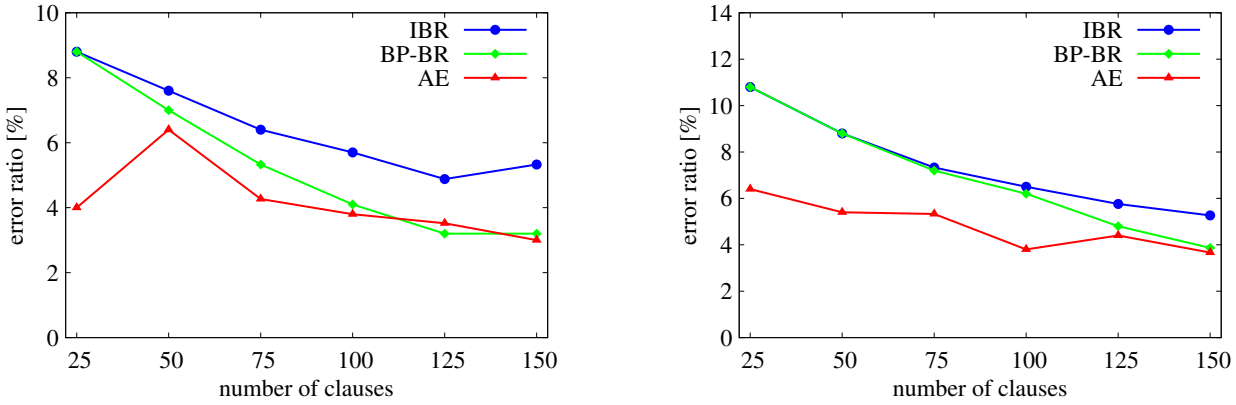
表 1: 3 つのアルゴリズムの比較 ($n = 20$)

(a) $\ell = 25, 50, 75$

	$\ell = 25$		$\ell = 50$		$\ell = 75$	
	lb	ub	lb	ub	lb	ub
IBR	22.8	25.0	45.9	49.7	69.2	74.0
BP-BR	22.8	25.0	45.9	49.4	69.2	73.2
AE	23.0	24.0	45.8	49.0	69.2	72.4

(b) $\ell = 100, 125, 150$

	$\ell = 100$		$\ell = 125$		$\ell = 150$	
	lb	ub	lb	ub	lb	ub
IBR	91.7	97.4	114.4	120.5	136.9	144.9
BP-BR	91.7	95.8	114.4	118.4	137.0	141.8
AE	91.7	95.5	114.3	118.7	136.9	141.4



(a) $n = 20$　　図 1: 節の数とエラー率　　(b) $n = 30$

である．AE の大きな特長として，全ての攻撃の組合せの数が定数とみなせる問題であれば効率的に上限と下限を求めることが可能である．

また，計算機実験により 3 つのアルゴリズムを比較し，同じ問題に対するエラー率の減少を確かめることで，AE アルゴリズムの有用性を示した．今後の課題として，変数の数や節に含まれるリテラルの数を変更する場合や，敵対者の攻撃を変更可能な場合の考察などが挙げられる．

謝辞　本研究は，日本学術振興会 科学研究費補助金 JP19H04175 および JP20H00609 の助成を受けたものです．深く感謝致します．

参考文献

[1] Karp, Richard M. Reducibility among combinatorial problems. Complexity of computer computations, Springer, pp. 85-103, 1972.
[2] Armin Biere, Marijn Heule, Hans, van Maaren, and Toby Walsh. Handbook of Satisfiability : Volume 185 Frontiers in Artificial Inteligence and Applications, IOS Press, 2009.
[3] 番原睦則, 鍋島英知. SAT 技術の進化と応用〜パズルからプログラム検証まで〜：1. SAT 技術の進化. 情報処理, Vol.57, No.8, pp. 704-709, 2016.
[4] Antonio Morgado, Federico Heras, Mark Liffiton, Jordi Planes and Joao Marques-Silva. Iterative and core-guided MaxSAT solving: A survey and assessment. Constraints, Vol.18, No.4 pp. 478-534, 2013.
[5] 越村三幸，藤田博．MaxSAT:SAT の最適化問題への拡張 -MaxSAT ソルバーの活用法- 情報処理学会誌 Vol.57, No.8, pp. 730-733, 2016.
[6] Larry J. Stockmeyer. The Polynomial-Time Hierarchy. Theoretical Computer Science. Vol.3, No.1, pp. 1-22, 1976.
[7] Emi Watanabe, Miyuki Koshimura, Nathanael Barrot, and Makoto Yokoo. Fighting against an adversary with NP oracle: Challenge to Σ_2^p-complete Problems. ゲーム理論ワークショップ 2020, 2020.
[8] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. PySAT: A Python toolkit for Prototyping with SAT Oracles. the 21st International Conference on Theory and Applications of Satisfiability Testing (SAT-2018), Springer, pp. 428-437, 2018.