SAT ソルバーの最新動向と利用技術

宋 剛秀 web, 田村 直之 web

神戸大学情報基盤センター

第 19 回プログラミングおよびプログラミング言語ワークショップ
PPL2017
2017 年 3 月 9 日 @ 石和温泉

目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - SAT 型システムの成功事例と近年の話題
- SAT ソルバーの求解性能の進化
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- SAT ソルバーの機能の進化
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- SAT ソルバーの利用技術
  - 符号化の重要性，ハイブリッド符号化
- おわりに

目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - SAT 型システムの成功事例と近年の話題
- SAT ソルバーの求解性能の進化
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- SAT ソルバーの機能の進化
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- SAT ソルバーの利用技術
  - 符号化の重要性，ハイブリッド符号化
- おわりに

はじめに

SAT, SAT 問題, SAT ソルバー,
SAT 符号化, SAT 型システム

SAT

SAT (Boolean satisfiability testing) は，与えられた命題論理式を真にする値割当てが存在するか否かを判定する問題である．

- SAT は NP-完全であることが最初に証明された問題 [Cook, 1971].
- SAT は，理論上も実際上も計算機科学にとって中心的 [Garey+, 1979].

SAT 問題

具体的な SAT 問題 (SAT Instances) は，連言標準形 (CNF) で与えられる．

CNF 式

- **CNF 式** は，複数の節の論理積 (連言) である．
- **節** (clause) は複数のリテラルの論理和 (選言) である．
- **リテラル** (literal) は，命題変数かあるいはその否定である．

標準的フォーマットとしては DIMACS CNF web が用いられる．

p cnf 3 4	; 命題変数の数␣節の数
1 2 3 0	; $p_1 \vee p_2 \vee p_3$
-1 -2 0	; $\neg p_1 \vee \neg p_2$
-1 -3 0	; $\neg p_1 \vee \neg p_3$
-2 -3 0	; $\neg p_2 \vee \neg p_3$

SAT ソルバー

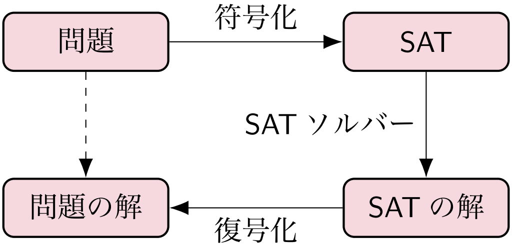
- **SAT ソルバー**は，与えられた SAT 問題が充足可能 (`SAT`) か充足不能 (`UNSAT`) かを判定するプログラムである．
- 通常，充足可能であればその値割当てを解として出力する．

- 系統的 SAT ソルバーは，`SAT` あるいは `UNSAT` を判定する．
  - ほとんどは **DPLL** を基にしたアルゴリズムを用いている．

SAT 型システム

SAT 型システムとは与えられた問題を **SAT 符号化** により SAT へと変換し，SAT ソルバーを用いて解くシステムである[†].

[†] 符号化と SAT ソルバーによる求解を繰り返すような場合も含む.



目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - **SAT 型システムの成功事例と近年の話題**
- SAT ソルバーの求解性能の進化
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- SAT ソルバーの機能の進化
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- SAT ソルバーの利用技術
  - 符号化の重要性，ハイブリッド符号化
- おわりに

SAT 型システムの成功事例と近年の話題

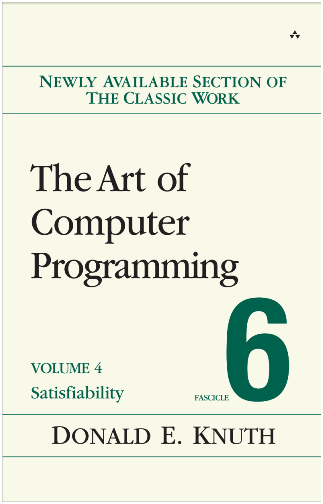
SAT 型システムの成功事例

- プランニング (SATPLAN, Blackbox) [Kautz+, 1992] ▸ web
- 自動テストパターン生成 [Larrabee, 1992]
- ジョブショップスケジューリング [Crawford+, 1994]
- 有界モデル検査 [Biere, 2009]
- ソフトウェア検証 (Alloy) [Jackson, 2006] ▸ web
- 書換えシステム (AProVE) [Jürgen+, 2004] ▸ web
- インテル社の i7 プロセッサの検証 [Kaivola+, 2009]
- Eclipse のコンポーネント間の依存解析 [Le Berre+, 2009] ▸ web
- 解集合プログラミング (clasp) [Gebser+, 2012] ▸ web
- Linux のパッケージマネージャである DNF の依存性解決 ▸ web
- 制約充足問題 (Sugar) [Tamura+, 2009] ▸ web
  - オープンショップスケジューリング問題の未解決問題の求解 [Tamura+, 2009]
  - パッキング配列問題の未解決問題の求解 [則武+, 2013]
- この他ペトリネットの検証，システム生物学，グラフ理論の問題などにも応用されている [Ogata+, 2004, Soh+, 2010, Soh+, 2014].

SAT に関連する特に最近の話題

- **2014 年 2 月**
  - Erdös Discrepancy Conjecture の $C = 2$ の場合の解決 [Konev+, 2014]
- **2015 年 12 月**
  - The Art of Computer Programming 最新分冊で SAT が取り上げられる [Knuth, 2015]
- **2016 年 5 月**
  - Boolean Pythagorean Triple 問題の解決 [Heule+, 2016]
    - ★ 発表当時 Nature 誌へこの話題が掲載される ▸ web
- **2017 年 2 月**
  - SHA-1 の衝突メッセージ作成の過程で SAT ソルバーが使われる ▸ web

The Art of Computer Programming (TAOCP)



→ Knuth 先生と食事した写真 (@Trento, SAT2012)

- スタンフォード大学 Knuth 教授著 "**The Art of Computer Programming**" の最新分冊である 4B 巻分冊 6 [Knuth, 2015] では SAT が 300 ページ以上にわたって取り上げられている.
- 序文には「SAT 問題は，非常に多くの問題を解くためのキーであることから，明らかに killer app である」と述べられている.

SHA-1 ハッシュ値の衝突

- 2017 年 2 月に同一の SHA-1 (Security Hash Algorithm 1) ハッシュ値をもつ 2 つのファイルが実際に作成された．

CV_0	4e	a9	62	69	7c	87	6e	26	74	d1	07	f0	fe	c6	79	84	14	f5	bf	45
$M_1^{(1)}$			7f	46	dc	93	a6	b6	7e	01	3b	02	9a	aa	1d	b2	56	0b		
			45	ca	67	d6	88	c7	f8	4b	8c	4c	79	1f	e0	2b	3d	f6		
			14	f8	6d	b1	69	09	01	c5	6b	45	c1	53	0a	fe	df	b7		
			60	38	e9	72	72	2f	e7	ad	72	8f	0e	49	04	e0	46	c2		
$CV_1^{(1)}$	8d	64	d6	17	ff	ed	53	52	eb	c8	59	15	5e	c7	eb	34	f3	8a	5a	7b
$M_2^{(1)}$			30	57	0f	e9	d4	13	98	ab	e1	2e	f5	bc	94	2b	e3	35		
			42	a4	80	2d	98	b5	d7	0f	2a	33	2e	c3	7f	ac	35	14		
			e7	4d	dc	0f	2c	c1	a8	74	cd	0c	78	30	5a	21	56	64		
			61	30	97	89	60	6b	d0	bf	3f	98	cd	a8	04	46	29	a1		
CV_2	1e	ac	b2	5e	d5	97	0d	10	f1	73	69	63	57	71	bc	3a	17	b4	8a	c5

CV_0	4e	a9	62	69	7c	87	6e	26	74	d1	07	f0	fe	c6	79	84	14	f5	bf	45
$M_1^{(2)}$			73	46	dc	91	66	b6	7e	11	8f	02	9a	b6	21	b2	56	0f		
			f9	ca	67	cc	a8	c7	f8	5b	a8	4c	79	03	0c	2b	3d	e2		
			18	f8	6d	b3	a9	09	01	d5	df	45	c1	4f	26	fe	df	b3		
			dc	38	e9	6a	c2	2f	e7	bd	72	8f	0e	45	bc	e0	46	d2		
$CV_1^{(2)}$	8d	64	c8	21	ff	ed	52	e2	eb	c8	59	15	5e	c7	eb	36	73	8a	5a	7b
$M_2^{(2)}$			3c	57	0f	eb	14	13	98	bb	55	2e	f5	a0	a8	2b	e3	31		
			fe	a4	80	37	b8	b5	d7	1f	0e	33	2e	df	93	ac	35	00		
			eb	4d	dc	0d	ec	c1	a8	64	79	0c	78	2c	76	21	56	60		
			dd	30	97	91	d0	6b	d0	af	3f	98	cd	a4	bc	46	29	b1		
CV_2	1e	ac	b2	5e	d5	97	0d	10	f1	73	69	63	57	71	bc	3a	17	b4	8a	c5

実際の衝突メッセージ．web より引用

- 報告したのは Google とオランダ国立数学・計算機科学研究センター（CWI）のチームで，2 番目の near-collision ブロックペアを見つける計算の一部で SAT ソルバーが使われた．

目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - SAT 型システムの成功事例と近年の話題
- **SAT ソルバーの求解性能の進化**
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- SAT ソルバーの機能の進化 (10 分)
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- SAT ソルバーの利用技術 (10 分)
  - 符号化の重要性，ハイブリッド符号化
- おわりに

SAT ソルバーの求解性能の進化

SAT ソルバーの求解性能の進化

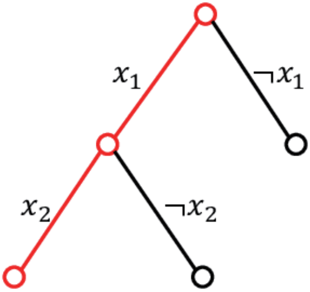
- 1960 年代
  - **DPLL** (Davis-Putnam-Logemann-Loveland) [Davis+, 1962]
- 1990 年代
  - **CDCL** (Conflict Driven Clause Learning) [Bayardo Jr.+, 1997, Marques-Silva+, 1999]
- 2000 年以降
  - 変数選択ヒューリスティック VSIDS [Moskewicz+, 2001]
  - 2 リテラルウォッチ [Moskewicz+, 2001]
  - リスタート [Luby+, 1993, Selman+, 1996, Eén+, 2003]
  - Phase Saving [Pipatsrisawat+, 2007]
  - 学習節の評価尺度 [Audemard+, 2009, 鍋島+, 2012]

SAT ソルバーの求解性能の進化

- 1960 年代
  - **DPLL** (Davis-Putnam-Logemann-Loveland) [Davis+, 1962]
- 1990 年代
  - **CDCL** (Conflict Driven Clause Learning) [Bayardo Jr.+, 1997, Marques-Silva+, 1999]
- 2000 年以降
  - 変数選択ヒューリスティック VSIDS [Moskewicz+, 2001]
  - 2 リテラルウォッチ [Moskewicz+, 2001]
  - リスタート [Luby+, 1993, Selman+, 1996, Eén+, 2003]
  - Phase Saving [Pipatsrisawat+, 2007]
  - 学習節の評価尺度 [Audemard+, 2009, 鍋島+, 2012]

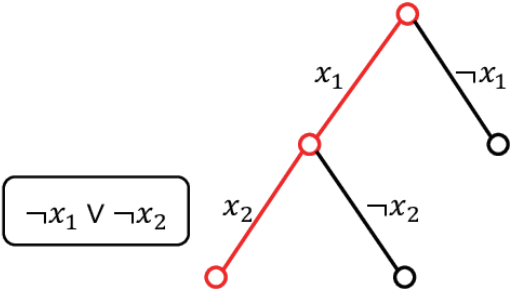
DPLL と CDCL の動作を簡単な例を用いて説明する

SAT ソルバーの基本 二分木の深さ優先探索



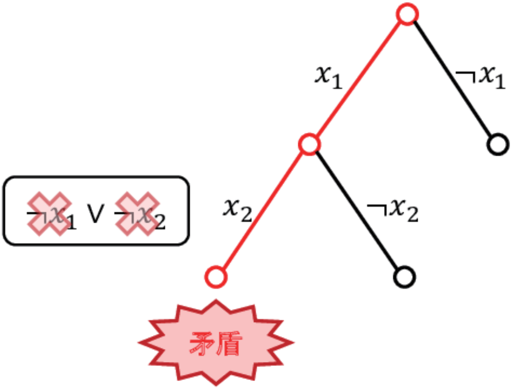
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論, 実装, 応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 二分木の深さ優先探索



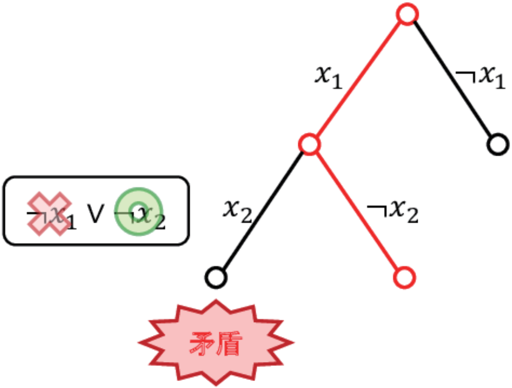
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 二分木の深さ優先探索



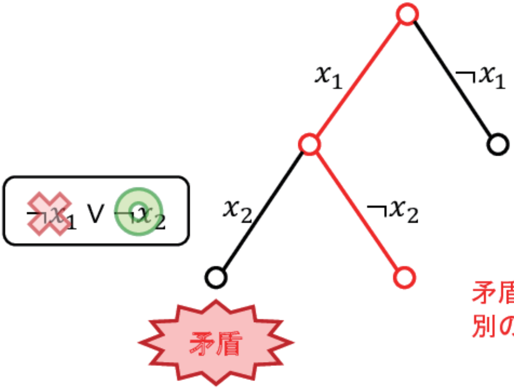
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 二分木の深さ優先探索



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 二分木の深さ優先探索

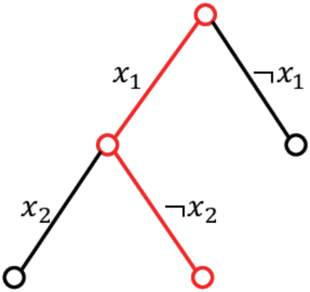


矛盾するとバックトラックし，
別の選択肢を探索

(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　単位伝搬

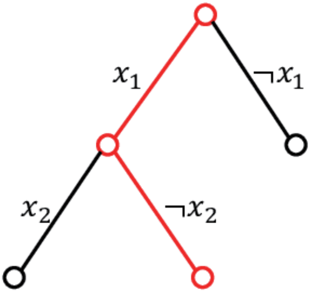
- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　単位伝搬

- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



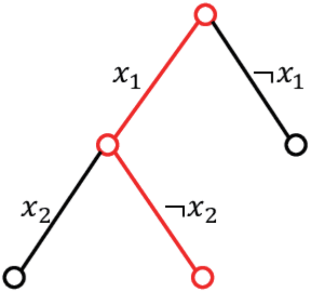
$x_2 \vee x_3$

$\neg x_3 \vee x_4$

(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　単位伝搬

- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



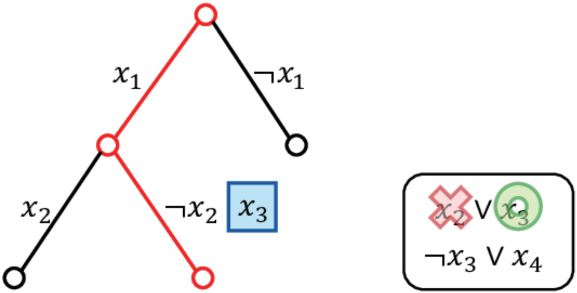
$\cancel{x_2} \lor x_3$

$\neg x_3 \lor x_4$

(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　単位伝搬

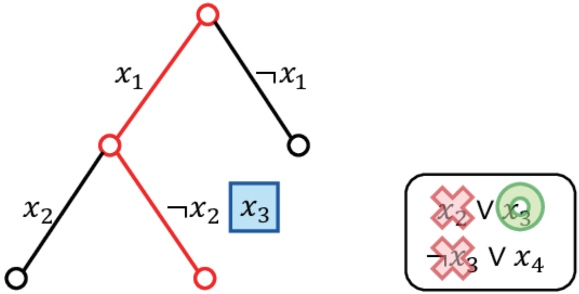
- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 単位伝搬

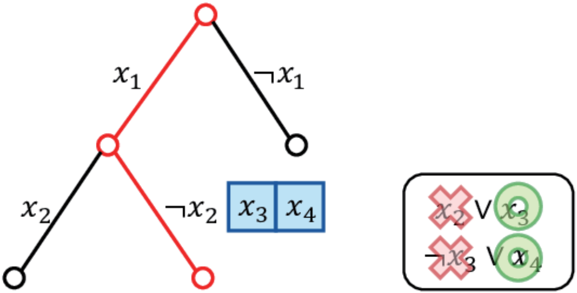
- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　単位伝搬

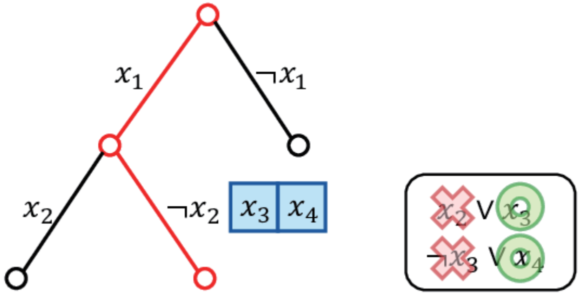
- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 単位伝搬

- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減

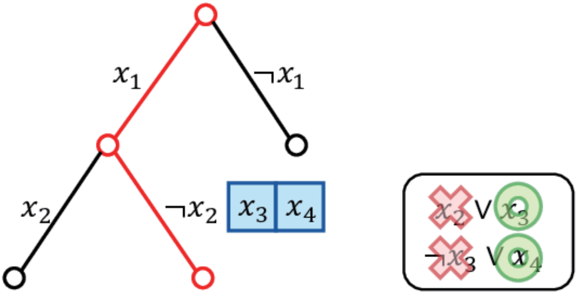


$x_1 \wedge \neg x_2$ の下では $x_3 = x_4 =$ 真が確定

(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 単位伝搬

- 単位伝搬により必然的な真偽値割当を検出
- 探索木の深さを低減

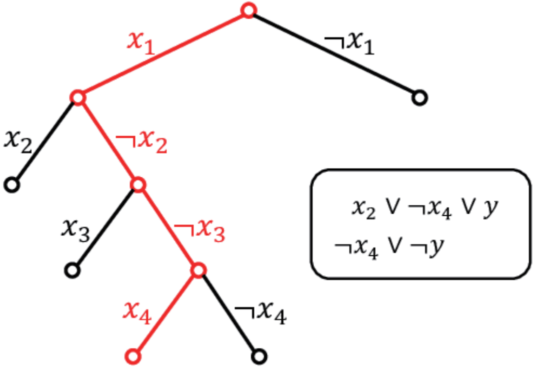


$x_1 \wedge \neg x_2$ の下では $x_3 = x_4 =$ 真が確定

二分木の深さ優先探索 ＋ 単位伝搬 ≒ DPLL

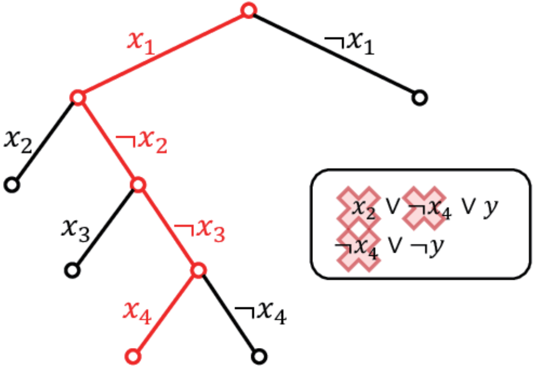
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　節学習 (CDCL)



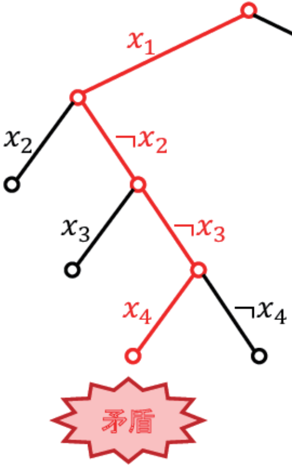
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 節学習 (CDCL)



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

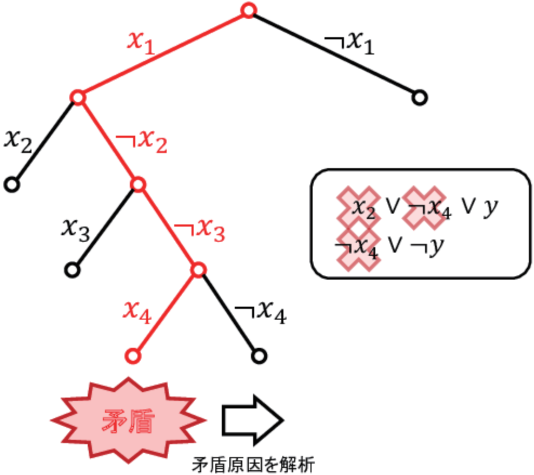
SAT ソルバーの基本　節学習 (CDCL)



$x_2 \vee \neg x_4 \vee y$
$\neg x_4 \vee \neg y$

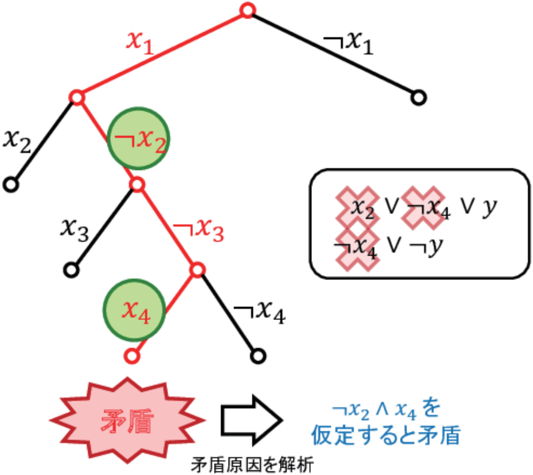
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 節学習 (CDCL)



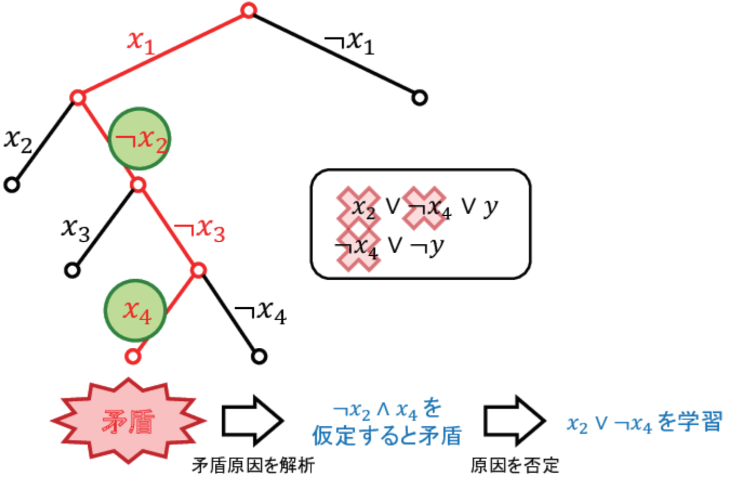
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　節学習 (CDCL)



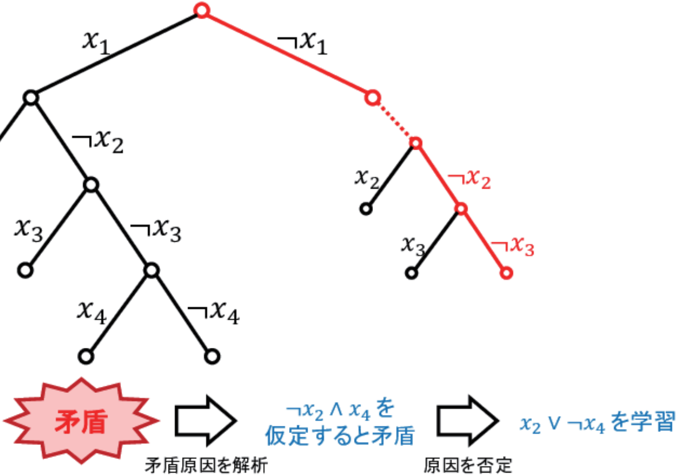
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論, 実装, 応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本　節学習 (CDCL)



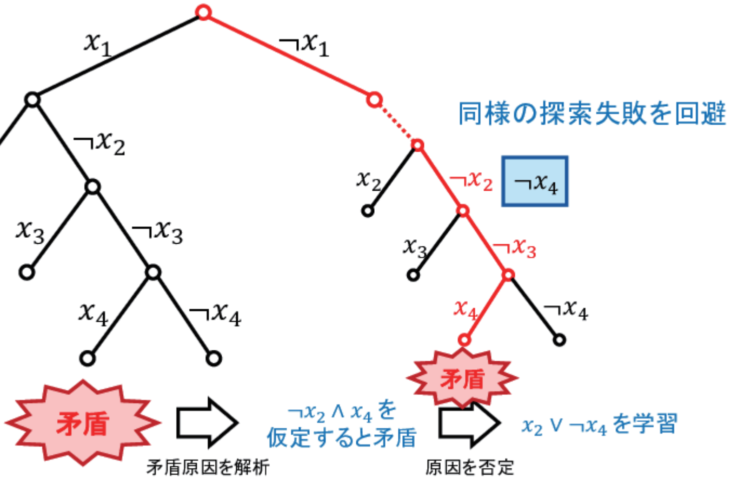
(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 節学習 (CDCL)



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの基本 節学習 (CDCL)



(次より引用) 鍋島英知, SAT ソルバーの最近の技術動向, 2016 年度人工知能学会全国大会オーガナイズドセッション「OS-2 SAT 技術の理論，実装，応用」, 1D4-OS-02a-1, 2016.

SAT ソルバーの求解性能の進化

- 1960 年代
  - **DPLL** (Davis-Putnam-Logemann-Loveland) [Davis+, 1962]
- 1990 年代
  - **CDCL** (Conflict Driven Clause Learcning) [Bayardo Jr.+, 1997, Marques-Silva+, 1999]
- 2000 年以降
  - 変数選択ヒューリスティック VSIDS [Moskewicz+, 2001]
  - 2 リテラルウォッチ [Moskewicz+, 2001]
  - リスタート [Luby+, 1993, Selman+, 1996, Eén+, 2003]
  - Phase Saving [Pipatsrisawat+, 2007]
  - 学習節の評価尺度 [Audemard+, 2009, 鍋島+, 2012]

SAT ソルバーの求解性能の進化

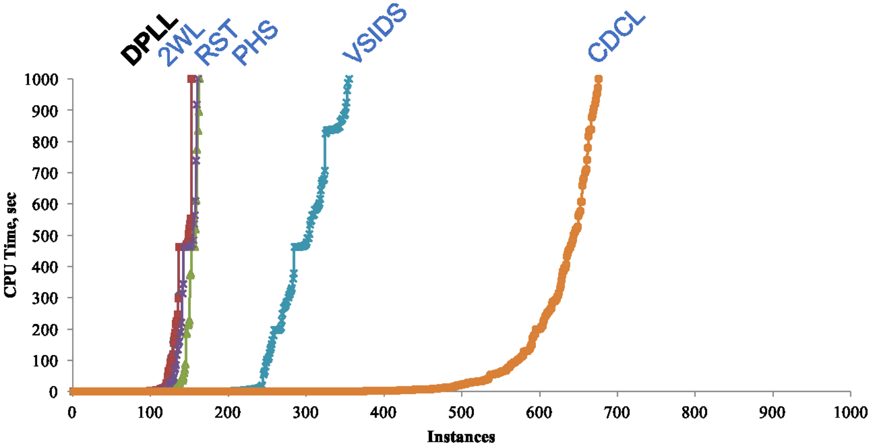
- 1960 年代
  - **DPLL** (Davis-Putnam-Logemann-Loveland) [Davis+, 1962]
- 1990 年代
  - **CDCL** (Conflict Driven Clause Learcning) [Bayardo Jr.+, 1997, Marques-Silva+, 1999]
- 2000 年以降
  - 変数選択ヒューリスティック VSIDS [Moskewicz+, 2001]
  - 2 リテラルウォッチ [Moskewicz+, 2001]
  - リスタート [Luby+, 1993, Selman+, 1996, Eén+, 2003]
  - Phase Saving [Pipatsrisawat+, 2007]
  - 学習節の評価尺度 [Audemard+, 2009, 鍋島+, 2012]

どの技術がどのくらい効いているのか?

高速化技術の貢献度 (Katebi 他による) [Katebi+, 2011]

- 2WL: 2 リテラルウォッチ, RST: リスタート, PHS: Phase Saving

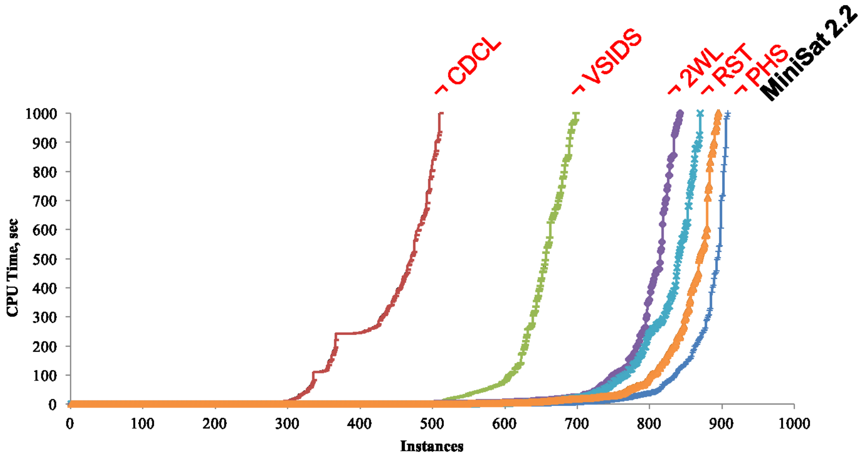
青字: DPLL にその技術を追加したソルバー



高速化技術の貢献度 (Katebi 他による) [Katebi+, 2011]

- 2WL: 2 リテラルウォッチ, RST: リスタート, PHS: Phase Saving

赤字: MiniSat 2.2 からその技術を削除したソルバー



代表的 SAT ソルバー (番原と鍋島による) [番原+, 2016]

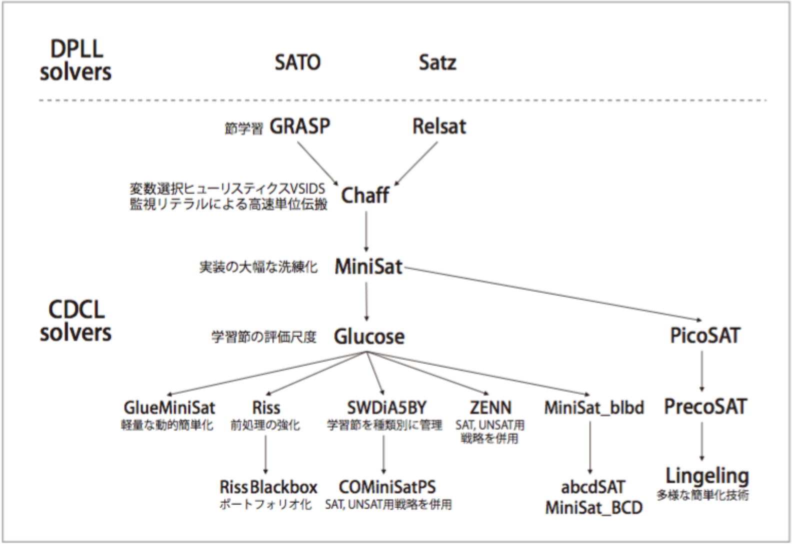


図 -3　歴史的 SAT ソルバーと 2013 ～2015 年の産業応用部門上位 SAT ソルバーの系統樹

SAT ソルバーの国際競技会

2002 年から継続して **SAT ソルバーの国際競技会** (SAT Competition, SATRace, SAT Challenge) が開催されている．

- 参加者は自分のソルバーを指定されたサーバにアップロードする．
- 参加者によるテスト期間後，最終版がオーガナイザ側で実行されて，SAT 国際会議の時に結果が公表される．

- 参加ソルバーのソースコードが原則公開.
  - 最新の技術・実装を誰でも入手でき，自分のアイデアを上乗せできる
  - 新規参入の敷居が低くなる
  - 競争が激しくなる
  - ほぼ毎年優勝ソルバーが変わり，優勝ソルバーには何らかの新しいアイデアが入っている．

この活発な競技会が SAT ソルバーの性能向上に大きく貢献している！

SAT ソルバーの国際競技会の 2016 年の結果 (逐次)

SAT+UNSAT	Gold	Silver	Bronze
Main Track (500)	MapleCOMSPS (203)	Riss (202)	Lingeling (201)

- **Main カテゴリ** は，ソフトウェア・ハードウェア検証，プランニング・スケジューリング等の AI 関連の問題，定理証明などの問題で構成されており，実問題における各 SAT ソルバーの性能が分かる最も重要な部門．29 ソルバーが参加．
- **MapleCOMSPS** は，機械学習を取り入れた変数選択ヒューリスティックと既存の VSIDS を両方使うソルバー．
- 結果サマリ ▸ web．各優勝ソルバの技術的なサマリ ▸ web．

SAT ソルバーの国際競技会の 2016 年の結果 (並列)

SAT+UNSAT	Gold	Silver	Bronze
Parellel Track (500)	Treengeling (315)	Plingeling (302)	CryptoMiniSat (297)
Main Track (500)	MapleCOMSPS (203)	Riss (202)	Lingeling (201)

- **Parallel カテゴリ** では 24 コア (48 スレッド) とメモリ 64GB (逐次は 24GB) の共有メモリ環境上で，Main の逐次ソルバーと同じベンチマークを使って並列 SAT ソルバーの性能が評価された．
- 逐次ソルバーの結果と比較すると約 1.5 倍の数の問題が解けるようになっており，大幅に性能が向上していることが分かる．

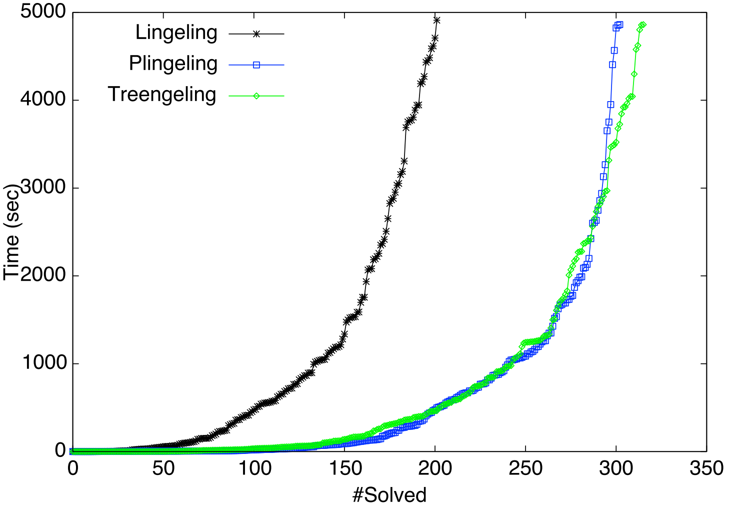
SAT ソルバーの国際競技会の 2016 年の結果 (並列)

SAT+UNSAT	Gold	Silver	Bronze
Parellel Track (500)	Treengeling (315)	Plingeling (302)	CryptoMiniSat (297)
Main Track (500)	MapleCOMSPS (203)	Riss (202)	Lingeling (201)

- **Parallel カテゴリ** では 24 コア (48 スレッド) とメモリ 64GB (逐次は 24GB) の共有メモリ環境上で，Main の逐次ソルバーと同じベンチマークを使って並列 SAT ソルバーの性能が評価された．
- 逐次ソルバーの結果と比較すると約 1.5 倍の数の問題が解けるようになっており，大幅に性能が向上していることが分かる．

逐次ソルバー Lingeling の**ポートフォリオ型**並列版 Plingeling と**探索空間分割型**並列版 Treengeling の性能向上をカクタスプロットで見てみよう．

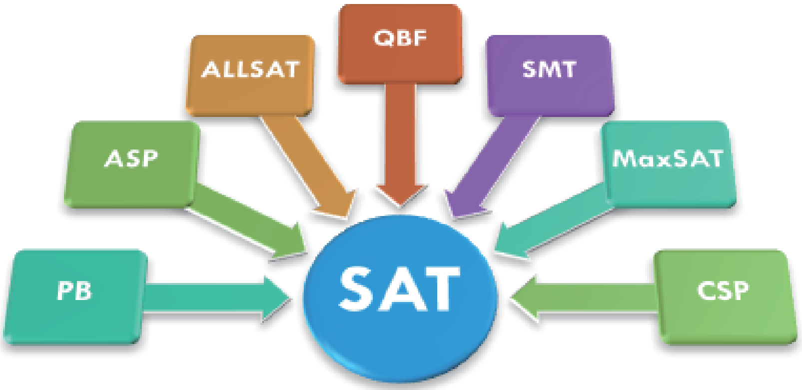
Lingeling における並列化の効果



- ポートフォリオ型でも大幅に性能が向上している．探索空間分割型ではさらに求解性能の伸びが見られる．

SAT の拡張問題への適用

ここまで説明した求解性能の向上を背景に，SAT ソルバーをコアエンジンとした SAT の拡張問題のソルバーも数多く開発されている．



目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - SAT 型システムの成功事例と近年の話題
- SAT ソルバーの求解性能の進化
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- **SAT ソルバーの機能の進化**
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- SAT ソルバーの利用技術
  - 符号化の重要性，ハイブリッド符号化
- おわりに

SAT ソルバーの機能の進化

SAT ソルバーの機能の進化

SAT ソルバーの国際競技会は単純な求解性能の進化だけでなく，様々な競技カテゴリを設けることで **SAT ソルバーの機能の拡張と標準化**にも貢献している．

- **Certified UNSAT** トラック
  - 問題が `UNSAT` の時に判定結果だけでなく，その証明を DRAT 形式 [Wetzler+, 2014] で出力・検証する必要があるトラック．
- **Minimal Unsatisfiable Subset** (MUS) トラック
  - 問題が `UNSAT` の時に判定結果だけでなく，与えられた問題が `UNSAT` となる極小の部分節集合を 1 つ出力する必要があるトラック．
- **インクリメンタル SAT** トラック
  - 少しずつ異なる複数の SAT 問題を連続的に解く性能を競うトラック．
  - この機能のための標準 API である IPASIR ▸ web の仕様が公開されている．

SAT ソルバーの機能の進化

SAT ソルバーの国際競技会は単純な求解性能の進化だけでなく，様々な競技カテゴリを設けることで **SAT ソルバーの機能の拡張と標準化**にも貢献している．

- **Certified UNSAT** トラック
  - 問題が `UNSAT` の時に判定結果だけでなく，その証明を DRAT 形式 [Wetzler+, 2014] で出力・検証する必要があるトラック．
- **Minimal Unsatisfiable Subset** (MUS) トラック
  - 問題が `UNSAT` の時に判定結果だけでなく，与えられた問題が `UNSAT` となる極小の部分節集合を 1 つ出力する必要があるトラック．
- **インクリメンタル SAT** トラック
  - 少しずつ異なる複数の SAT 問題を連続的に解く性能を競うトラック．
  - この機能のための標準 API である IPASIR ▸ web の仕様が公開されている．

これらの機能を使った応用例を 1 つずつ紹介する．

Certified UNSAT の応用例

- **Certified UNSAT 機能** は UNSAT が正しいことを保証する.
- その原理は反駁に至る導出木が正しいか検証することであるが，効率的な検証のために，ソルバーが探索課程で獲得する各学習節が正しいかを順に調べる手法が利用されている [Gelder, 2007, Heule+, 2013, Wetzler+, 2014].

実際に以下の SAT ソルバーを使った事例では，Certified UNSAT 機能を使うことで証明が正しいことが確認された.

- Erdös discrepancy 予想の $C = 2$ の場合の解決 [Konev+, 2014]
- Boolean Pythagorean Triple 問題の解決 [Heule+, 2016] ▸ web

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \lor x_2$	$\neg x_6 \lor x_2$	$\neg x_2 \lor x_1$	$\neg x_1$
$\neg x_6 \lor x_8$	$x_6 \lor \neg x_8$	$x_2 \lor x_4$	$\neg x_4 \lor x_5$
$x_7 \lor x_5$	$\neg x_7 \lor x_5$	$\neg x_5 \lor x_3$	$\neg x_3$

入力となる CNF 式.
MaxSAT 問題では同時に充足できる節の数を最大化する (充足しない節を最小化する).

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \lor x_2$	$\neg x_6 \lor x_2$	$\neg x_2 \lor x_1$	$\neg x_1$
$\neg x_6 \lor x_8$	$x_6 \lor \neg x_8$	$x_2 \lor x_4$	$\neg x_4 \lor x_5$
$x_7 \lor x_5$	$\neg x_7 \lor x_5$	$\neg x_5 \lor x_3$	$\neg x_3$

まず最初にこの CNF 式が UNSAT であることを SAT ソルバーで判定し，UNSAT コアを抽出する．

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$$
\begin{array}{cccc}
x_6 \lor x_2 & \neg x_6 \lor x_2 & \neg x_2 \lor x_1 \lor b_1 & \neg x_1 \lor b_2 \\
\neg x_6 \lor x_8 & x_6 \lor \neg x_8 & x_2 \lor x_4 \lor b_3 & \neg x_4 \lor x_5 \lor b_4 \\
x_7 \lor x_5 & \neg x_7 \lor x_5 & \neg x_5 \lor x_3 \lor b_5 & \neg x_3 \lor b_6
\end{array}
$$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$

(問題の緩和) ブロック変数を追加し，UNSAT コアに含まれていた部分節集合の中の 1 つの節については充足しなくても良いようにする． f は SAT 符号化.

UNSATコアを利用したMaxSAT問題の解法 [Fu+, 2006]

$x_6 \vee x_2$	$\neg x_6 \vee x_2$	$\neg x_2 \vee x_1 \vee b_1$	$\neg x_1 \vee b_2$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee b_3$	$\neg x_4 \vee x_5 \vee b_4$
$x_7 \vee x_5$	$\neg x_7 \vee x_5$	$\neg x_5 \vee x_3 \vee b_5$	$\neg x_3 \vee b_6$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$

もう一度このCNF式の充足性をSATソルバーを使って判定し，再びUNSATが得られる． UNSATコアを抽出する．

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \vee x_2 \vee b_7$	$\neg x_6 \vee x_2 \vee b_8$	$\neg x_2 \vee x_1 \vee b_1 \vee b_9$	$\neg x_1 \vee b_2 \vee b_{10}$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee b_3$	$\neg x_4 \vee x_5 \vee b_4$
$x_7 \vee x_5 \vee b_{11}$	$\neg x_7 \vee x_5 \vee b_{12}$	$\neg x_5 \vee x_3 \vee b_5 \vee b_{13}$	$\neg x_3 \vee b_6 \vee b_{14}$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$
$$f(b_7 + b_8 + b_9 + b_{10} + b_{11} + b_{12} + b_{13} + b_{14} \leq 1)$$

(問題の緩和) ブロック変数を追加し，UNSAT コアに含まれていた部分節集合の中の 1 つの節については充足しなくても良いようにする．f は SAT 符号化.

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \lor x_2 \lor b_7$	$\neg x_6 \lor x_2 \lor b_8$	$\neg x_2 \lor x_1 \lor b_1 \lor b_9$	$\neg x_1 \lor b_2 \lor b_{10}$
$\neg x_6 \lor x_8$	$x_6 \lor \neg x_8$	$x_2 \lor x_4 \lor b_3$	$\neg x_4 \lor x_5 \lor b_4$
$x_7 \lor x_5 \lor b_{11}$	$\neg x_7 \lor x_5 \lor b_{12}$	$\neg x_5 \lor x_3 \lor b_5 \lor b_{13}$	$\neg x_3 \lor b_6 \lor b_{14}$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$
$$f(b_7 + b_8 + b_9 + b_{10} + b_{11} + b_{12} + b_{13} + b_{14} \leq 1)$$

もう一度この CNF 式の充足性を SAT ソルバーを使って判定し，SAT が得られる．

UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \vee x_2 \vee b_7$	$\neg x_6 \vee x_2 \vee b_8$	$\neg x_2 \vee x_1 \vee b_1 \vee b_9$	$\neg x_1 \vee b_2 \vee b_{10}$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee b_3$	$\neg x_4 \vee x_5 \vee b_4$
$x_7 \vee x_5 \vee b_{11}$	$\neg x_7 \vee x_5 \vee b_{12}$	$\neg x_5 \vee x_3 \vee b_5 \vee b_{13}$	$\neg x_3 \vee b_6 \vee b_{14}$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$
$$f(b_7 + b_8 + b_9 + b_{10} + b_{11} + b_{12} + b_{13} + b_{14} \leq 1)$$

12 節の中で，2 つの節については充足しないことを許すことで，初めて入力 CNF 式が SAT になり MaxSAT の解は 10 になる．

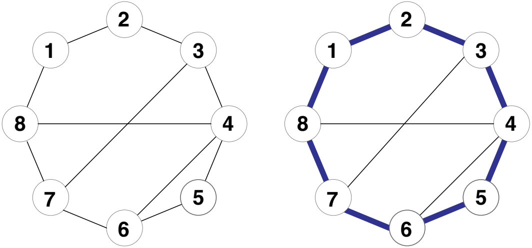
UNSAT コアを利用した MaxSAT 問題の解法 [Fu+, 2006]

$x_6 \vee x_2 \vee b_7$	$\neg x_6 \vee x_2 \vee b_8$	$\neg x_2 \vee x_1 \vee b_1 \vee b_9$	$\neg x_1 \vee b_2 \vee b_{10}$
$\neg x_6 \vee x_8$	$x_6 \vee \neg x_8$	$x_2 \vee x_4 \vee b_3$	$\neg x_4 \vee x_5 \vee b_4$
$x_7 \vee x_5 \vee b_{11}$	$\neg x_7 \vee x_5 \vee b_{12}$	$\neg x_5 \vee x_3 \vee b_5 \vee b_{13}$	$\neg x_3 \vee b_6 \vee b_{14}$

$$f(b_1 + b_2 + b_3 + b_4 + b_5 + b_6 \leq 1)$$
$$f(b_7 + b_8 + b_9 + b_{10} + b_{11} + b_{12} + b_{13} + b_{14} \leq 1)$$

- UNSAT コアを利用した最適化は継続して研究されている [Morgado+, 2013].
- 応用として ASP ソルバーのシステムが時間割問題の既存手法を大きく上回る性能を示すことも報告されている [Banbara+, 2013].

インクリメンタルSAT: ハミルトン閉路問題への応用



- ハミルトン閉路問題 (HCP) は与えられたグラフの全頂点を通る閉路を求める NP 完全問題である.
- グラフ理論における重要な問題であり，計算機科学においても巡回セールスマン問題 (TSP) との関係があり重要である.
- TSP の潜在的な困難さが HCP にある可能性の示唆 [Filar '06].

インクリメンタル SAT: ハミルトン閉路問題への応用

2003 年までの従来方法

- [Iwama+, 1994] log encoding
- [Hoos, 1999] absolute encocoding
- [Prestwich, 2003] relative encoding
  - 連結制約を任意の 3 頂点の遷移関係を用いて符号化． n^3 **の節**が必要．

[Velev '09]

- [Velev+, 2009] は [Prestwich, 2003] の方法を踏襲しながらグラフの三角化を用いることで符号化節の大幅な削減に成功．
- 改善された方法は 1000 倍以上の高速化を実現．

HCP/TSP 専用ソルバー LKH web は 10000 頂点規模の HCP を解けるが，SAT 解法は [Velev '09] らの方法を使っても 1000 頂点以上の問題は解くのが難しい．

インクリメンタルSAT: ハミルトン閉路問題への応用

- V: n 個の頂点集合, E: 辺の集合, $G=(V,E)$: 無向グラフ
- 辺集合 E に対応する弧の集合 $A=\{(i,j)\in V^2 \mid \{i,j\}\in E\}$
- ブール変数 x_{ij} $(i \neq j)$ を以下のように定義.

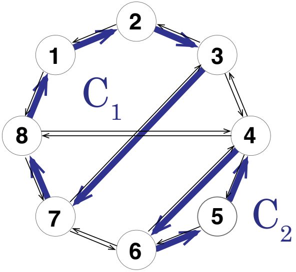
$x_{ij}=1 \Leftrightarrow$ **弧 (i,j) が解に含まれる**

(出次数制約)	$\displaystyle\sum_{(i,j)\in A} x_{ij}=1$	各頂点 $i=1,\ldots,n$ に対して
(入次数制約)	$\displaystyle\sum_{(i,j)\in A} x_{ij}=1$	各頂点 $j=1,\ldots,n$ に対して
(連結制約)	$\displaystyle\sum_{i,j\in S} x_{ij}\leq \lvert S\rvert-1,$	$S\subset V,\ 3\leq \lvert S\rvert\leq n-2$

- **次数制約**は解に含まれる弧が閉路を構成することを保証する
- **連結制約**は解に含まれる弧が全ての頂点を連結することを保証する

インクリメンタルSAT: CEGAR-HCP [Soh+, 2014]

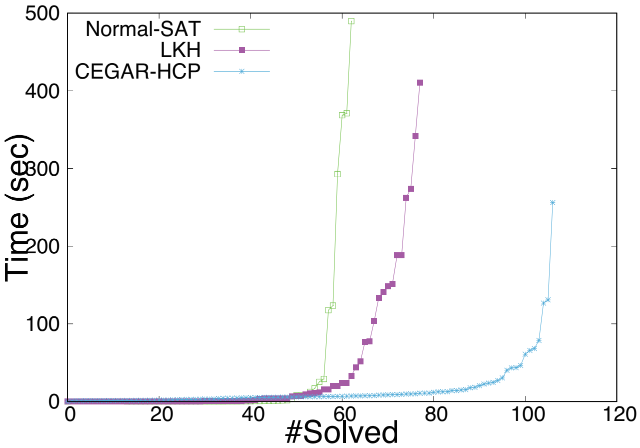
$\Psi := G$ の次数制約のみを SAT 符号化 (緩和問題を符号化) ;
while (Ψ が充足可能)　　# SAT ソルバー呼び出し
　if (解が閉路を 1 つしか含まない)
　　return G のハミルトン閉路;
　else
　　Ψ_{block} := 反例からブロック節を生成 ;
　$\Psi := \Psi \land \Psi_{block}$;
return ハミルトン閉路は存在しない;



ブロック節

C_1 $\neg x_{12} \lor \neg x_{23} \lor \neg x_{37} \lor \neg x_{78} \lor \neg x_{81}$
C'_1 $\neg x_{87} \lor \neg x_{73} \lor \neg x_{32} \lor \neg x_{21} \lor \neg x_{18}$
C_2 $\neg x_{46} \lor \neg x_{65} \lor \neg x_{54}$
C'_2 $\neg x_{45} \lor \neg x_{56} \lor \neg x_{64}$

インクリメンタルSAT: CEGAR-HCP [Soh+, 2014]



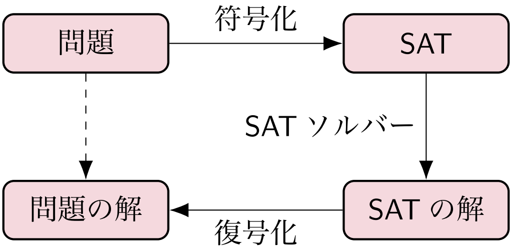
普通に SAT 問題に符号化する場合 (**Normal**), 巡回セールスマン問題の世界記録保持ソルバー (**LKH**) ▸ web, CEGAR を使ったインクリメンタル SAT 解法 (**CEGAR-HCP**) の比較.

目次

- はじめに
  - SAT, SAT 問題, SAT ソルバー, 符号化, SAT 型システム
  - SAT 型システムの成功事例と近年の話題
- SAT ソルバーの求解性能の進化
  - DPLL, CDCL, SAT 競技会, 並列 SAT
- SAT ソルバーの機能の進化
  - Certified UNSAT, UNSAT コア, インクリメンタル SAT, CEGAR の紹介とそれらを利用した応用例
- **SAT ソルバーの利用技術**
  - 符号化の重要性，ハイブリッド符号化
- おわりに

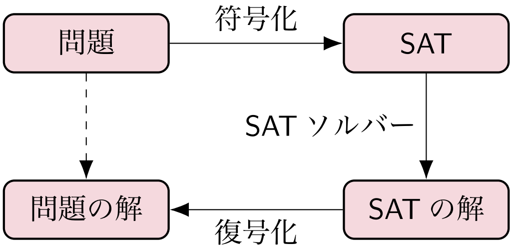
SAT ソルバーの利用技術
〜 SAT 符号化について 〜

SAT ソルバーを利用するには?



- ここまで SAT ソルバーの求解性能や機能の進化を説明した．
- しかし，解きたい問題が CNF 式で記述されていることは非常に少ないため，SAT ソルバーを利用するには，与えられた問題を SAT ソルバーの入力に変換する **SAT 符号化** が必須になる．

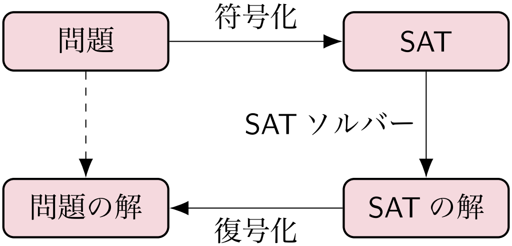
SAT ソルバーを利用するには?



- ここまで SAT ソルバーの求解性能や機能の進化を説明した．
- しかし，解きたい問題が CNF 式で記述されていることは非常に少ないため，SAT ソルバーを利用するには，与えられた問題を SAT ソルバーの入力に変換する **SAT 符号化** が必須になる．

CNF 式にさえすれば良い? SAT ソルバーが勝手に高速に解く?

SAT ソルバーを利用するには?



- ここまで SAT ソルバーの求解性能や機能の進化を説明した．
- しかし，解きたい問題が CNF 式で記述されていることは非常に少ないため，SAT ソルバーを利用するには，与えられた問題を SAT ソルバーの入力に変換する **SAT 符号化** が必須になる．

CNF 式にさえすれば良い? SAT ソルバーが勝手に高速に解く?

No!

例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \le k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化

例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \leq k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化
  - この単純な場合でも，数多くの SAT 符号化が研究されている!
    [Warners, 1998, Bailleux+, 2003, Sinz, 2005, Chen, 2010, Roberto+, 2011, Abío+, 2013, Nguyen+, 2015]

例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \leq k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化
  - この単純な場合でも，数多くの SAT 符号化が研究されている! [Warners, 1998, Bailleux+, 2003, Sinz, 2005, Chen, 2010, Roberto+, 2011, Abío+, 2013, Nguyen+, 2015]
- x_i が 0-1 変数の場合 (**擬似ブール制約**) の符号化 [Eén+, 2006, Bailleux+, 2006, Bailleux+, 2009, Roussel+, 2009, Abío+, 2012, Ogawa+, 2013, Tamura+, 2013, Philipp+, 2015, Sakai+, 2015]

例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \leq k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化
  - この単純な場合でも，数多くの SAT 符号化が研究されている! [Warners, 1998, Bailleux+, 2003, Sinz, 2005, Chen, 2010, Roberto+, 2011, Abío+, 2013, Nguyen+, 2015]
- x_i が 0-1 変数の場合 (**擬似ブール制約**) の符号化 [Eén+, 2006, Bailleux+, 2006, Bailleux+, 2009, Roussel+, 2009, Abío+, 2012, Ogawa+, 2013, Tamura+, 2013, Philipp+, 2015, Sakai+, 2015]
- x_i **が有限整数領域上のドメインを持つ** 場合の符号化 [de Kleer, 1989, Kasif, 1990, Iwama+, 1994, Walsh, 2000, Gent, 2002, Gavanelli, 2007, Prestwich, 2009, Tamura+, 2009, 田村+, 2010, 丹生+, 2013, Soh+, 2017]

例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \le k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化
  - この単純な場合でも，数多くの SAT 符号化が研究されている! [Warners, 1998, Bailleux+, 2003, Sinz, 2005, Chen, 2010, Roberto+, 2011, Abío+, 2013, Nguyen+, 2015]
- x_i が 0-1 変数の場合 (**擬似ブール制約**) の符号化 [Eén+, 2006, Bailleux+, 2006, Bailleux+, 2009, Roussel+, 2009, Abío+, 2012, Ogawa+, 2013, Tamura+, 2013, Philipp+, 2015, Sakai+, 2015]
- x_i **が有限整数領域上のドメインを持つ** 場合の符号化 [de Kleer, 1989, Kasif, 1990, Iwama+, 1994, Walsh, 2000, Gent, 2002, Gavanelli, 2007, Prestwich, 2009, Tamura+, 2009, 田村+, 2010, 丹生+, 2013, Soh+, 2017]

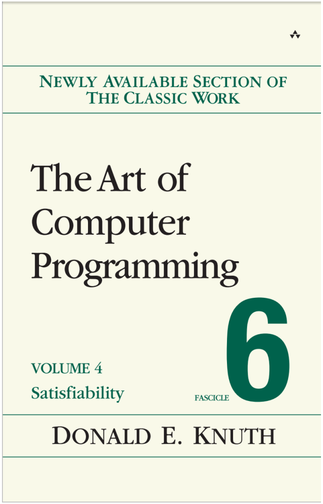
例えば，線形比較の符号化の場合

$$\sum_{i=1}^{n} a_i x_i \leq k \quad (a_i > 0)$$

- 全ての a_i が 1, x_i が 0-1 変数の場合 (**基数制約**) の符号化
  - この単純な場合でも，数多くの SAT 符号化が研究されている! [Warners, 1998, Bailleux+, 2003, Sinz, 2005, Chen, 2010, Roberto+, 2011, Abío+, 2013, Nguyen+, 2015]
- x_i が 0-1 変数の場合 (**擬似ブール制約**) の符号化 [Eén+, 2006, Bailleux+, 2006, Bailleux+, 2009, Roussel+, 2009, Abío+, 2012, Ogawa+, 2013, Tamura+, 2013, Philipp+, 2015, Sakai+, 2015]
- x_i **が有限整数領域上のドメインを持つ** 場合の符号化 [de Kleer, 1989, Kasif, 1990, Iwama+, 1994, Walsh, 2000, Gent, 2002, Gavanelli, 2007, Prestwich, 2009, Tamura+, 2009, 田村+, 2010, 丹生+, 2013, Soh+, 2017]

符号化の研究は SAT 分野において重要なテーマの 1 つ!

符号化は重要! from TAOCP [Knuth, 2015]



The Art of Computer Programming 4B 巻分冊 6 [Knuth, 2015] の 97 ページには以下のように記載されている.

“Thus the art of problem encoding turns out to be just as important as the art of devising algorithms for satisfiability.”

(例) 直接符号化と順序符号化

直接符号化 [de Kleer, 1989]

各整数変数 x とそのドメインの各値 i に対して，$x = i$ を表す命題変数 $p_{x=i}$ を用いる．各整数変数 x に対して用いる命題変数

$$p_{x=i} \qquad (\mathit{lb}(x) \le i \le \mathit{ub}(x))$$

命題変数 $p_{x=i}$ が $x = i$ の時かつその時に限って真となるように，以下の at-least-one 節と at-most-one 節を追加する．

$$p_{x=\mathit{lb}(x)} \vee \cdots \vee p_{x=\mathit{ub}(x)}$$

$$\neg p_{x=i} \vee \neg p_{x=j}$$

$$(\mathit{lb}(x) \le i < j \le \mathit{ub}(x))$$

順序符号化 [Tamura+, 2009]

各整数変数 x とそのドメインの各値 i に対して，$x \le i$ を表す命題変数 $p_{x\le i}$ を用いる．各整数変数 x に対して用いる命題変数

$$p_{x\le i} \qquad (\mathit{lb}(x) \le i < \mathit{ub}(x))$$

命題変数 $p_{x\le i}$ が $x \le i$ の時かつその時に限って真となるように，以下の節を追加する．

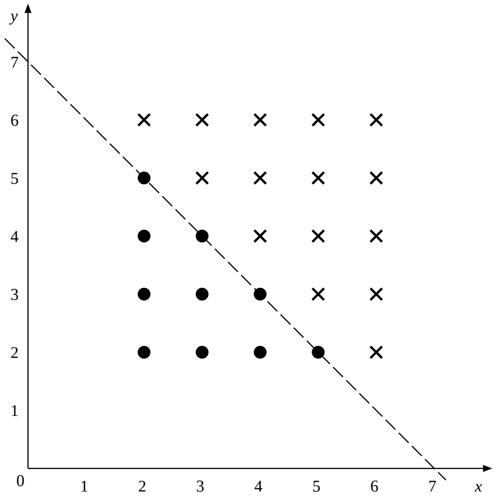
$$\neg p_{x\le i-1} \vee p_{x\le i}$$

$$(\mathit{lb}(x) < i < \mathit{ub}(x))$$

直接符号化 [de Kleer, 1989, Walsh, 2000]

制約 $x + y \leq 7$ $(x, y \in \{2,3,4,5,6\})$ は，違反点 (図中の $\times$ 点) を列挙することで以下の 15 節に符号化される．

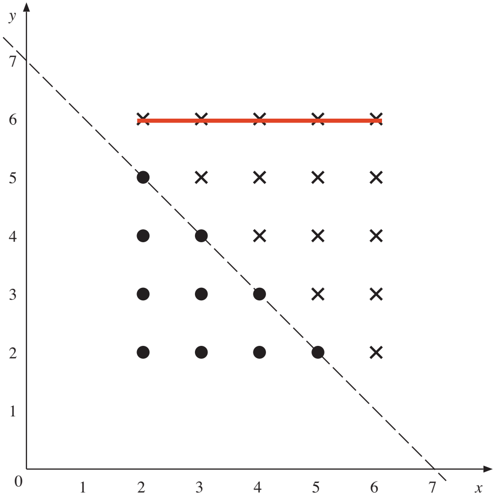
$$\neg(p_{x=2} \wedge p_{y=6})$$
$$\neg(p_{x=3} \wedge p_{y=5})$$
$$\neg(p_{x=3} \wedge p_{y=6})$$
$$\neg(p_{x=4} \wedge p_{y=4})$$
$$\neg(p_{x=4} \wedge p_{y=5})$$
$$\neg(p_{x=4} \wedge p_{y=6})$$
$$\neg(p_{x=5} \wedge p_{y=3})$$
$$\neg(p_{x=5} \wedge p_{y=4})$$
$$\neg(p_{x=5} \wedge p_{y=5})$$
$$\neg(p_{x=5} \wedge p_{y=6})$$
$$\neg(p_{x=6} \wedge p_{y=2})$$
$$\neg(p_{x=6} \wedge p_{y=3})$$
$$\neg(p_{x=6} \wedge p_{y=4})$$
$$\neg(p_{x=6} \wedge p_{y=5})$$
$$\neg(p_{x=6} \wedge p_{y=6})$$



順序符号化 [Tamura+, 2009]

制約 $x + y \leq 7 \ (x, y \in \{2, 3, 4, 5, 6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

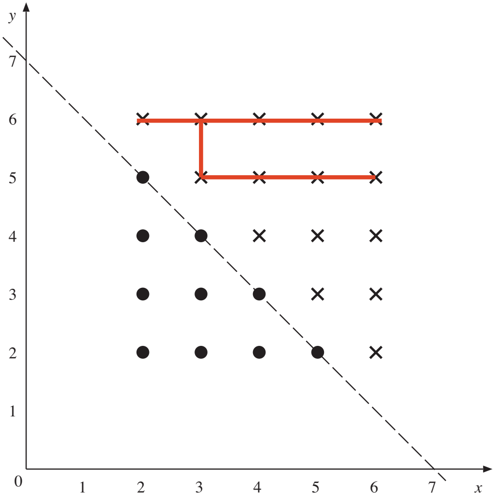
$$\neg p_{y \geq 6}$$
$$\neg(p_{x \geq 3} \land p_{y \geq 5})$$
$$\neg(p_{x \geq 4} \land p_{y \geq 4})$$
$$\neg(p_{x \geq 5} \land p_{y \geq 3})$$
$$\neg p_{x \geq 6}$$



順序符号化 [Tamura+, 2009]

制約 $x + y \le 7$ $(x, y \in \{2, 3, 4, 5, 6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

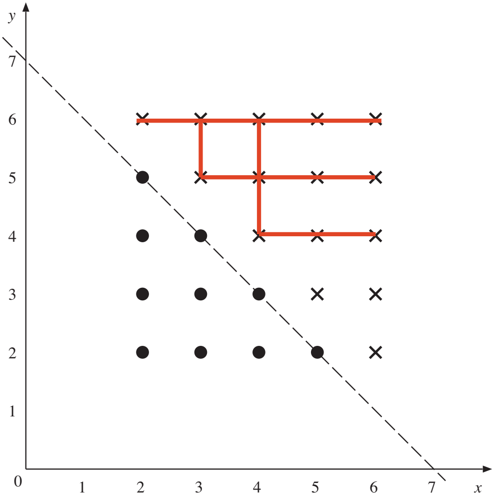
$$\neg p_{y\ge6}$$
$$\neg(p_{x\ge3} \land p_{y\ge5})$$
$$\neg(p_{x\ge4} \land p_{y\ge4})$$
$$\neg(p_{x\ge5} \land p_{y\ge3})$$
$$\neg p_{x\ge6}$$



順序符号化 [Tamura+, 2009]

制約 $x + y \leq 7$ $(x, y \in \{2,3,4,5,6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

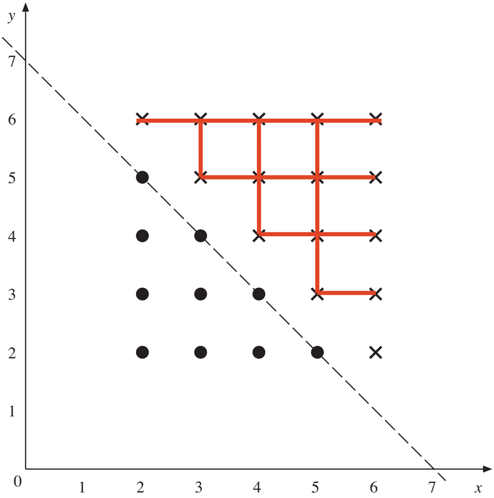
$$\neg p_{y \geq 6}$$
$$\neg(p_{x \geq 3} \wedge p_{y \geq 5})$$
$$\neg(p_{x \geq 4} \wedge p_{y \geq 4})$$
$$\neg(p_{x \geq 5} \wedge p_{y \geq 3})$$
$$\neg p_{x \geq 6}$$



順序符号化 [Tamura+, 2009]

制約 $x+y \leq 7$ $(x, y \in \{2,3,4,5,6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

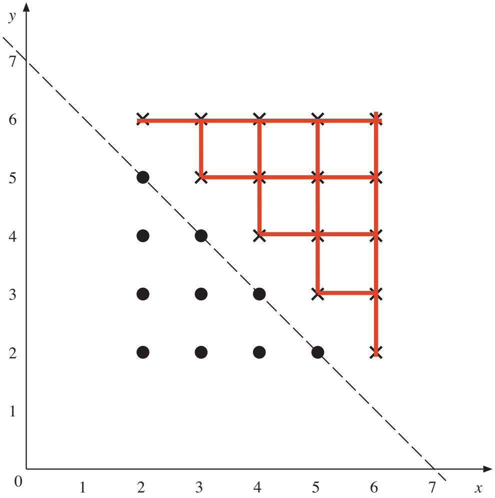
$$\neg p_{y\geq 6}$$
$$\neg(p_{x\geq 3} \land p_{y\geq 5})$$
$$\neg(p_{x\geq 4} \land p_{y\geq 4})$$
$$\neg(p_{x\geq 5} \land p_{y\geq 3})$$
$$\neg p_{x\geq 6}$$



順序符号化 [Tamura+, 2009]

制約 $x + y \leq 7$ $(x, y \in \{2,3,4,5,6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

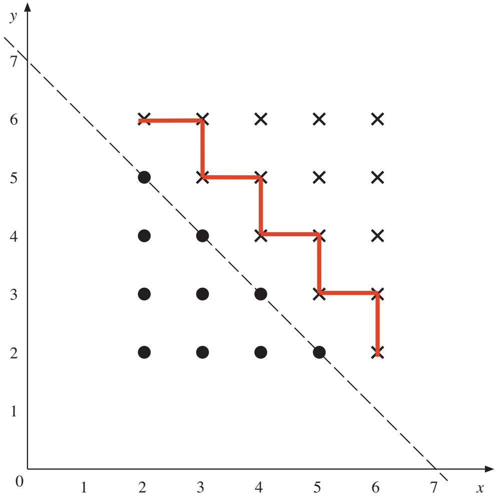
$$\neg p_{y\geq 6}$$
$$\neg(p_{x\geq 3} \land p_{y\geq 5})$$
$$\neg(p_{x\geq 4} \land p_{y\geq 4})$$
$$\neg(p_{x\geq 5} \land p_{y\geq 3})$$
$$\neg p_{x\geq 6}$$



順序符号化 [Tamura+, 2009]

制約 $x + y \leq 7$ $(x, y \in \{2, 3, 4, 5, 6\})$ は，違反する範囲を表すことで以下の 5 節に符号化される．

$$\neg p_{y \geq 6}$$
$$\neg(p_{x \geq 3} \land p_{y \geq 5})$$
$$\neg(p_{x \geq 4} \land p_{y \geq 4})$$
$$\neg(p_{x \geq 5} \land p_{y \geq 3})$$
$$\neg p_{x \geq 6}$$



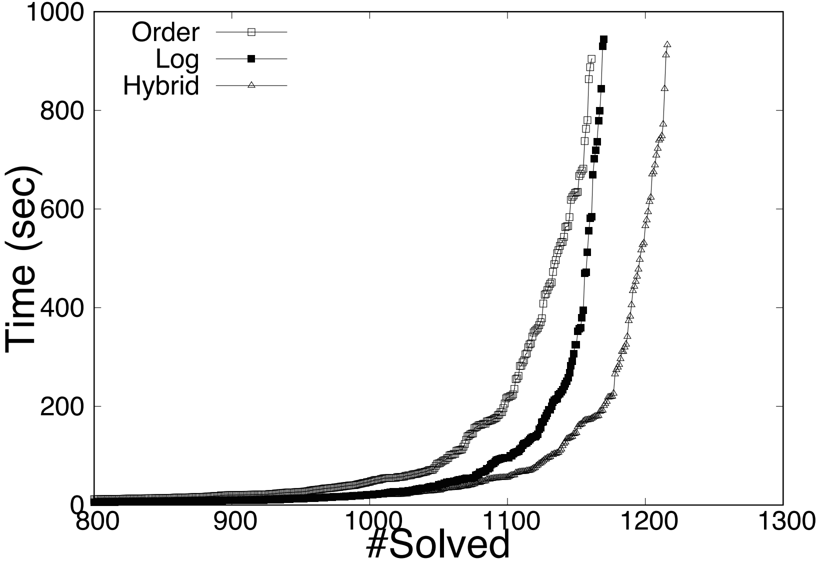
ハイブリッド符号化 [Soh+, 2017]

- 順序符号化 [Tamura+, 2009] (実装: Sugar ▸ web)
  - 整数の順序関係を用いており，線形比較 $\sum a_i x_i \geq b$ の符号化に適している．
  - 順序符号化を実装した Sugar は 2008 年，2009 年に CSP ソルバー競技会のグローバル部門で優勝した．
  - 理論的にも他の符号化にはない良い性質を持っていることが証明されている [Petke+, 2011].
  - しかし，問題の規模が大きくなると性能が悪くなる．
- 対数符号化 [Iwama+, 1994]
  - 整数の 2 進数表現を用いておりコンパクトな SAT 符号化が可能だが，線形比較について一般に順序符号化より性能が悪い．
  - 問題の規模が大きくても解くことができる．

ハイブリッド符号化 [Soh+, 2017]

- 順序符号化 [Tamura+, 2009] (実装: Sugar web)
  - 整数の順序関係を用いており，線形比較 $\sum a_i x_i \geq b$ の符号化に適している．
  - 順序符号化を実装した Sugar は 2008 年，2009 年に CSP ソルバー競技会のグローバル部門で優勝した．
  - 理論的にも他の符号化にはない良い性質を持っていることが証明されている [Petke+, 2011].
  - しかし，問題の規模が大きくなると性能が悪くなる．
- 対数符号化 [Iwama+, 1994]
  - 整数の 2 進数表現を用いておりコンパクトな SAT 符号化が可能だが，線形比較について一般に順序符号化より性能が悪い．
  - 問題の規模が大きくても解くことができる．
- **ハイブリッド符号化** [Soh+, 2017] (実装: Diet-Sugar web)
  - 順序符号化と対数符号化を融合した SAT 符号化法．
  - 各整数変数は順序符号化もしくは対数符号化のどちらか一つで符号化され，各制約は両方の符号化変数を含むことができる．

順序符号化，対数符号化，ハイブリッド符号化の比較 (CSC2009 ベンチマーク 1458 問で評価) [Soh+, 2017]

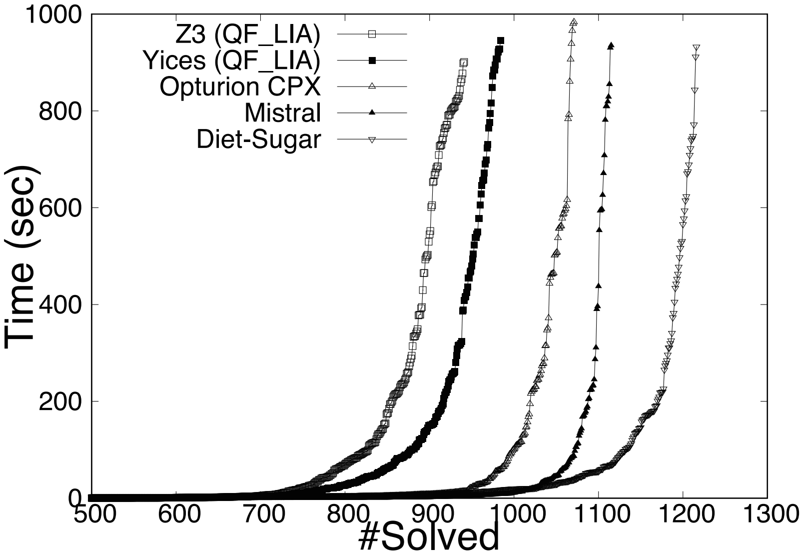


Diet-Sugar と CSP/SMT ソルバーとの比較

比較したソルバー

- **Mistral** (version 1.550)
  - ピュアな (SAT 型でない) CSP ソルバー．2009 年の CSP ソルバー競技会の優勝ソルバー．
- **Opturion CPX** (version 1.0.2)
  - CSP ソルバーと SAT ソルバーのハイブリッドソルバー．2015 年の Minizinc チャレンジ "Fixed/Free" カテゴリの優勝ソルバー．
- **Yices** (version 2.4.2) — QF_LIA
  - 2015 年の SMT ソルバー競技会 "Quantified Free Linear Integer Arithmetic (QF_LIA)" 部門の優勝ソルバー．
- **z3** (version 4.3.2) — QF_LIA
  - 2015 年の SMT ソルバー競技会 QF_LIA 部門の優勝ソルバー．

Diet-Sugar と CSP/SMT ソルバーとの比較 (CSC2009 ベンチマーク 1458 問で評価) [Soh+, 2017]



おわりに

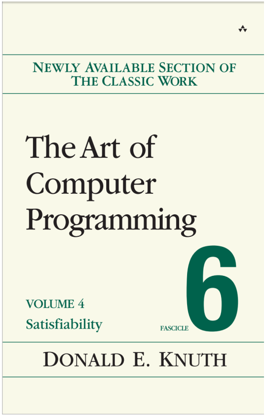
- 本発表では，以下の説明を行った．
  - SAT 型システムの成功事例と近年の話題
  - SAT ソルバーの求解性能の進化
  - SAT ソルバーの機能の進化
  - SAT ソルバーの利用に必須である符号化技術

SAT の面白いところ

- SAT ソルバーは，教科書通りの充足可能性を判定するプログラムではなく，インクリメンタル SAT，Certified UNSAT, UNSAT コアなど様々な機能を提供するようになっている．
- SAT 型システムもそれらを問題解法に利用する研究が行われている．
- SAT ソルバーを使った問題解法は決して「ソルバーまかせ」ではなく，問題のモデリング，符号化，SAT ソルバーの機能の活用などを深く検討しないと良い解法にはならない．
- しかし，それらがうまくはまった時には既存の専用解法を大きく上回ることも珍しくない．

参考情報

SAT の教科書



The Art of Computer Programming
4B 巻分冊 6 [Knuth, 2015]

Handbook of Satisfiability
[Biere+, 2009]

SAT 技術の進化と応用 ～パズルからプログラム検証まで～, 情報処理 57 巻 8 号 (2016 年 8 月号)

- **SAT 技術の進化** [番原+, 2016]
- **SAT とパズル ～問題をいかに SAT ソルバーで解くか～** [田村+, 2016]
- **SAT とラムゼー数** ～数学の未解決問題への挑戦～ [藤田+, 2016]
- **SAT と AI** [井上, 2016]
- **SAT ソルバーの最近の進展** [鍋島+, 2016]
- **MaxSAT：SAT の最適化問題への拡張 ～MaxSAT ソルバーの活用法～** [越村+, 2016]
- **SMT ソルバーによるプログラム検証** [石井+, 2016]

コンピュータソフトウェアの解説

- **SAT ソルバ・SMT ソルバの技術と応用** [梅村, 2010]
- **SAT 問題と他の制約問題との相互発展** [酒井+, 2015]
- **SAT 型制約プログラミングシステムと周辺技術** [宋+, 2017]

特集「最近の SAT 技術の発展」, 人工知能学会誌 25 巻 1 号 (2010 年 1 月号)

- **SAT ソルバーの基礎** [井上+, 2010]
- **高速 SAT ソルバーの原理** [鍋島+, 2010]
- **制約最適化問題と SAT 符号化** [田村+, 2010]
- **SMT: 個別理論を取り扱う SAT 技術** [岩沼+, 2010]
- **モデル列挙とモデル計数** [長谷川+, 2010]
- $*-$**SAT: SAT の拡張** [平山+, 2010]
- **SAT によるプランニングとスケジューリング** [鍋島, 2010]
- **SAT によるシステム検証** [番原+, 2010]

その他の情報

- **Constraint Solvers Catalog** ▸ web
  - 制約ソルバーを実装言語，モデリング言語，利用可能な制約等の様々な観点でまとめたカタログサイト．
- **Satisfiability modulo theories** ▸ web
  - Wikipedia の SMT のページ．SMT ソルバーを背景理論や実装言語でまとめた表が記載されている．
- **私のブックマーク (SAT)** ▸ web
  - SAT に関するプロジェクト，ソルバー，競技会，ハンドブック・解説・スライドなどの情報が URL と共に紹介されたブックマーク集．
- **CSPSAT プロジェクト** ▸ web
  - 2009 年にスタートした日本国内のプロジェクト．現在 CSPSAT3 が進行中．
- **発表者らのソフトウェア**
  - SAT 型制約ソルバー Sugar ▸ web
  - SAT 型制約ソルバー Diet-Sugar ▸ web
  - SAT 型制約プログラミングシステム Scarab ▸ web
  - Scarab チュートリアル ▸ web

参考文献

Reference I

[Abío+, 2013] Abío, Ignasi, Nieuwenhuis, Robert, Oliveras, Albert , and Rodríguez-Carbonell, Enric (2013).
A parametric approach for smaller and better encodings of cardinality constraints.
In *Principles and Practice of Constraint Programming - 19th International Conference, CP 2013, Uppsala, Sweden, September 16-20, 2013. Proceedings*, pages 80–96.

[Abío+, 2012] Abío, Ignasi, Nieuwenhuis, Robert, Oliveras, Albert, Rodríguez-Carbonell, Enric , and Mayer-Eichberger, Valentin (2012).
A new look at BDDs for pseudo-Boolean constraints.
Journal of Artificial Intelligence Research, 45:443–480.

Reference II

[Roberto+, 2011] Roberto As¥́in, Robert Nieuwenhuis, Albert Oliveras , Enric Rodr¥́iguez-Carbonell (2011).
Cardinality networks: a theoretical and empirical study.
Constraints, 16(2):195–221.

[Audemard+, 2009] Audemard, Gilles and Simon, Laurent (2009).
Predicting learnt clauses quality in modern SAT solvers.
In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pages 399–404.

[Bailleux+, 2003] Bailleux, Olivier and Boufkhad, Yacine (2003).
Efficient CNF encoding of Boolean cardinality constraints.
In *Proceedings of the 9th International Conference on Principles and Practice of Constraint Programming (CP 2003), LNCS 2833*, pages 108–122.

Reference III

[Bailleux+, 2006] Bailleux, Olivier, Boufkhad, Yacine , and Roussel, Olivier (2006).
A translation of pseudo Boolean constraints to SAT.
Journal on Satisfiability, Boolean Modeling and Computation, 2(1-4):191–200.

[Bailleux+, 2009] Bailleux, Olivier, Boufkhad, Yacine , and Roussel, Olivier (2009).
New encodings of pseudo-Boolean constraints into CNF.
In *Proceedings of the 12th International Conference on Theory and Applications of Satisfiability Testing (SAT 2009), LNCS 5584*, pages 181–194.

[番原+, 2016] 番原 睦則 , 鍋島 英知 (2016).
SAT 技術の進化.
情報処理, 57(8):704–709.

Reference IV

[Banbara+, 2013] Banbara, Mutsunori, Soh, Takehide, Tamura, Naoyuki, Inoue, Katsumi , and Schaub, Torsten (2013).
Answer set programming as a modeling language for course timetabling.
TPLP, 13(4-5):783–798.

[番原+, 2010] 番原 睦則 , 田村 直之 (2010).
SAT によるシステム検証.
人工知能学会誌, 25(1):122–129.

[Bayardo Jr.+, 1997] Bayardo Jr., Roberto J. and Schrag, Robert (1997).
Using CSP look-back techniques to solve real-world SAT instances.
In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI 1997)*, pages 203–208.

[Biere, 2009] Biere, Armin (2009).
Bounded model checking.
In *Handbook of Satisfiability*, pages 457–481. IOS Press.

Reference V

[Biere+, 2009] Biere, Armin, Heule, Marijn, van Maaren, Hans , and Walsh, Toby, editors (2009).
Handbook of Satisfiability, volume 185 of *Frontiers in Artificial Intelligence and Applications (FAIA)*. IOS Press.

[Chen, 2010] Chen, Jingchao (2010).
A New SAT Encoding of the At-Most-One Constraint.
In *The 9th International Workshop on Constraint Modelling and Reformulation (ModRef 2010) at the 16th International Conference on the Principles and Practice of Constraint Programming (CP 2010)*, pages 1–8.

[Cook, 1971] Cook, Stephen A. (1971).
The complexity of theorem-proving procedures.
In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing (STOC 1971)*, pages 151–158.

Reference VI

[Crawford+, 1994] Crawford, James M. and Baker, Andrew B. (1994).
Experimental results on the application of satisfiability algorithms to scheduling problems.
In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI 1994)*, pages 1092–1097.

[Davis+, 1962] Davis, Martin, Logemann, George , and Loveland, Donald W. (1962).
A machine program for theorem-proving.
Communications of the ACM, 5(7):394–397.

[de Kleer, 1989] de Kleer, Johan (1989).
A comparison of ATMS and CSP techniques.
In *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI 1989)*, pages 290–296.

Reference VII

[Eén+, 2003] Eén, Niklas and Sörensson, Niklas (2003).
An extensible SAT-solver.
In *Proceedings of the 6th International Conference on Theory and Applications of Satisfiability Testing (SAT 2003), LNCS 2919*, pages 502–518.

[Eén+, 2006] Eén, Niklas and Sörensson, Niklas (2006).
Translating pseudo-Boolean constraints into SAT.
Journal on Satisfiability, Boolean Modeling and Computation, 2(1-4):1–26.

[Fu+, 2006] Fu, Zhaohui and Malik, Sharad (2006).
On solving the partial MAX-SAT problem.
In *Proceedings of the 9th International Conference on Theory and Applications of Satisfiability Testing (SAT 2006), LNCS 4121*, pages 252–265.

Reference VIII

[藤田+, 2016] 藤田 博 , 越村 三幸 (2016).
SAT とラムゼー数 ～数学の未解決問題への挑戦～.
情報処理, 57(8):716–719.

[Garey+, 1979] Garey, Michael R. and Johnson, David S. (1979).
Computers and Intractability: A Guide to the Theory of NP-Completeness.
W. H. Freeman and Company, New York.

[Gavanelli, 2007] Gavanelli, Marco (2007).
The log-support encoding of CSP into SAT.
In *Proceedings of the 13th International Conference on Principles and Practice of Constraint Programming (CP 2007), LNCS 4741*, pages 815–822.

Reference IX

[Gebser+, 2012] Gebser, Martin, Kaufmann, Benjamin , and Schaub, Torsten (2012).
Conflict-driven answer set solving: From theory to practice.
Artif. Intell., 187:52–89.

[Gelder, 2007] Gelder, Allen Van (2007).
Verifying propositional unsatisfiability: Pitfalls to avoid.
In *Proceedings of the 10th International Conference on Theory and Applications of Satisfiability Testing (SAT)*, pages 328–333.

[Gent, 2002] Gent, Ian P. (2002).
Arc consistency in SAT.
In *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI 2002)*, pages 121–125.

Reference X

[Jürgen+, 2004] Jürgen Giesl, Ren￥'e Thiemann, Peter Schneider-Kamp , and Stephan Falke (2004).
Automated termination proofs with aprove.
In *Rewriting Techniques and Applications, 15th International Conference, RTA 2004, Aachen, Germany, June 3-5, 2004, Proceedings*, pages 210–220.

[長谷川+, 2010] 長谷川 隆三, 藤田 博 , 越村 三幸 (2010).
モデル列挙とモデル計数.
人工知能学会誌, 25(1):96–104.

[Heule+, 2013] Heule, Marijn, Jr., Warren A. Hunt , and Wetzler, Nathan (2013).
Trimming while checking clausal proofs.
In *Formal Methods in Computer-Aided Design, FMCAD 2013, Portland, OR, USA, October 20-23, 2013*, pages 181–188.

Reference XI

[Heule+, 2016] Heule, Marijn J. H., Kullmann, Oliver , and Marek, Victor W. (2016).
Solving and verifying the boolean pythagorean triples problem via cube-and-conquer.
In *Theory and Applications of Satisfiability Testing - SAT 2016 - 19th International Conference, Bordeaux, France, July 5-8, 2016, Proceedings*, pages 228–245.

[鍋島+, 2012] 鍋島 英知, 岩沼 宏治 , 井上 克巳 (2012).
Glueminisat 2.2.5: 単位伝搬を促す学習節の積極的獲得戦略に基づく高速 SAT ソルバー.
コンピュータソフトウェア, 29(4):211–230.

[平山+, 2010] 平山 勝敏 , 横尾 真 (2010).
∗-SAT: SAT の拡張.
人工知能学会誌, 25(1):105–113.

Reference XII

[Hoos, 1999] Hoos, Holger H. (1999).
SAT-encodings, search space structure, and local search performance.
In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI 1999)*, pages 296–303.

[井上, 2016] 井上 克巳 (2016).
SAT と AI.
情報処理, 57(8):720–723.

[井上+, 2010] 井上 克巳 , 田村 直之 (2010).
SAT ソルバーの基礎.
人工知能学会誌, 25(1):57–67.

[石井+, 2016] 石井 大輔 , 上田 和紀 (2016).
SMT ソルバーによるプログラム検証.
情報処理, 57(8):734–737.

Reference XIII

[Iwama+, 1994] Iwama, Kazuo and Miyazaki, Shuichi (1994).
SAT-variable complexity of hard combinatorial problems.
In *Proceedings of the IFIP 13th World Computer Congress*, pages 253–258.

[岩沼+, 2010] 岩沼 宏治 , 鍋島 英知 (2010).
SMT: 個別理論を取り扱う SAT 技術.
人工知能学会誌, 25(1):86–95.

[Jackson, 2006] Jackson, Daniel (2006).
Software Abstractions - Logic, Language, and Analysis.
MIT Press.

Reference XIV

[Kaivola+, 2009] Kaivola, Roope, Ghughal, Rajnish, Narasimhan, Naren, Telfer, Amber, Whittemore, Jesse, Pandav, Sudhindra, Slobodová, Anna, Taylor, Christopher, Frolov, Vladimir A., Reeber, Erik , and Naik, Armaghan (2009).
Replacing testing with formal verification in intel coretm i7 processor execution engine validation.
In *Proceedings of the 21st International Conference on Computer Aided Verification (CAV 2009), LNCS 5643*, pages 414–429.

[Kasif, 1990] Kasif, Simon (1990).
On the parallel complexity of discrete relaxation in constraint satisfaction networks.
Artificial Intelligence, 45(3):275–286.

Reference XV

[Katebi+, 2011] Katebi, Hadi, Sakallah, Karem A. , and Silva, João P. Marques (2011).
Empirical study of the anatomy of modern sat solvers.
In *Proceedings of the 14th International Conference on Theory and Applications of Satisfiability Testing (SAT 2011), LNCS 6695*, pages 343–356.

[Kautz+, 1992] Kautz, Henry A. and Selman, Bart (1992).
Planning as satisfiability.
In *Proceedings of the 10th European Conference on Artificial Intelligence (ECAI 1992)*, pages 359–363.

[Knuth, 2015] Knuth, Donald E. (2015).
The Art of Computer Programming, Volume 4B.
Addison-Wesley Professional.

Reference XVI

[Konev+, 2014] Konev, Boris and Lisitsa, Alexei (2014).
A SAT attack on the erdős discrepancy conjecture.
In *Theory and Applications of Satisfiability Testing - SAT 2014 - 17th International Conference, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 14-17, 2014. Proceedings*, pages 219–226.

[越村+, 2016] 越村 三幸 , 藤田 博 (2016).
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, 57(8):730–733.

[Larrabee, 1992] Larrabee, Tracy (1992).
Test pattern generation using boolean satisfiability.
IEEE Trans. on CAD of Integrated Circuits and Systems, 11(1):4–15.

Reference XVII

[Le Berre+, 2009] Le Berre, Daniel and Rapicault, Pascal (2009).
Dependency management for the eclipse ecosystem: Eclipse p2, metadata and resolution.
In *Proceedings of the 1st International Workshop on Open Component Ecosystems*, IWOCE '09, pages 21–30.

[Luby+, 1993] Luby, Michael, Sinclair, Alistair , and Zuckerman, David (1993).
Optimal speedup of Las Vegas algorithms.
Information Processing Letters, 47(4):173–180.

[Marques-Silva+, 1999] Marques-Silva, João P. and Sakallah, Karem A. (1999).
GRASP: A search algorithm for propositional satisfiability.
IEEE Transactions on Computers, 48(5):506–521.

Reference XVIII

[Morgado+, 2013] Morgado, António, Heras, Federico, Liffiton, Mark H., Planes, Jordi , and Marques-Silva, João (2013).
Iterative and core-guided maxsat solving: A survey and assessment.
Constraints, 18(4):478–534.

[Moskewicz+, 2001] Moskewicz, Matthew W., Madigan, Conor F., Zhao, Ying, Zhang, Lintao , and Malik, Sharad (2001).
Chaff: Engineering an efficient SAT solver.
In *Proceedings of the 38th Design Automation Conference (DAC 2001)*, pages 530–535.

[鍋島, 2010] 鍋島 英知 (2010).
SAT によるプランニングとスケジューリング.
人工知能学会誌, 25(1):114–121.

Reference XIX

[鍋島+, 2016] 鍋島 英知, 岩沼 宏治 , 井上 克巳 (2016).
SAT ソルバーの最近の進展.
情報処理, 57(8):724–729.

[鍋島+, 2010] 鍋島 英知 , 宋 剛秀 (2010).
高速 SAT ソルバーの原理.
人工知能学会誌, 25(1):68–76.

[Nguyen+, 2015] Nguyen, Van-Hau and Mai, Son Thai (2015).
A new method to encode the at-most-one constraint into SAT.
In *Proceedings of the Sixth International Symposium on Information and Communication Technology, Hue City, Vietnam, December 3-4, 2015*, pages 46–53.

Reference XX

[則武+, 2013] 則武 治樹, 番原 睦則, 宋 剛秀, 田村 直之 , 井上 克巳 (2013).
パッキング配列問題の制約モデリングと SAT 符号化.
コンピュータソフトウェア, 31(1):116–130.

[Ogata+, 2004] Ogata, Shougo, Tsuchiya, Tatsuhiro , and Kikuno, Tohru (2004).
Sat-based verification of safe petri nets.
In *Automated Technology for Verification and Analysis: Second International Conference, ATVA 2004, Taipei, Taiwan, ROC, October 31-November 3, 2004. Proceedings*, pages 79–92.

Reference XXI

[Ogawa+, 2013] Ogawa, Toru, Liu, Yangyang, Hasegawa, Ryuzo, Koshimura, Miyuki , and Fujita, Hiroshi (2013).
Modulo based CNF encoding of cardinality constraints and its application to maxsat solvers.
In *2013 IEEE 25th International Conference on Tools with Artificial Intelligence, Herndon, VA, USA, November 4-6, 2013*, pages 9–17.

[Petke+, 2011] Petke, Justyna and Jeavons, Peter (2011).
The order encoding: From tractable csp to tractable sat.
In *Proceedings of the 14th International Conference on Theory and Applications of Satisfiability Testing (SAT 2011), LNCS 6695*, pages 371–372.

[Philipp+, 2015] Philipp, Tobias and Steinke, Peter (2015).
Pblib – a library for encoding pseudo-boolean constraints into cnf.
In Heule, Marijn and Weaver, Sean, editors, *Theory and Applications of Satisfiability Testing – SAT 2015*, volume 9340 of *Lecture Notes in Computer Science*, pages 9–16. Springer International Publishing.

[Pipatsrisawat+, 2007] Pipatsrisawat, Knot and Darwiche, Adnan (2007).
A lightweight component caching scheme for satisfiability solvers.
In *Proceedings of the 10th International Conference on Theory and Applications of Satisfiability Testing (SAT 2007), LNCS 4501*, pages 294–299.

[Prestwich, 2003] Prestwich, Steven David (2003).
SAT problems with chains of dependent variables.
Discrete Applied Mathematics, 130(2):329–350.

Reference XXIII

[Prestwich, 2009] Prestwich, Steven David (2009).
CNF encodings.
In *Handbook of Satisfiability*, pages 75–97. IOS Press.

[Roussel+, 2009] Roussel, Olivier and Manquinho, Vasco M. (2009).
Pseudo-Boolean and cardinality constraints.
In Biere, Armin, Heule, Marijn, van Maaren, Hans , and Walsh, Toby, editors, *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*, pages 695–733. IOS Press.

[酒井+, 2015] 酒井 政裕 , 今井 健男 (2015).
SAT 問題と他の制約問題との相互発展.
コンピュータソフトウェア, 32(1):103–119.

Reference XXIV

[Sakai+, 2015] Sakai, Masahiko and Nabeshima, Hidetomo (2015).
Construction of an ROBDD for a pb-constraint in band form and related techniques for pb-solvers.
IEICE Transactions, 98-D(6):1121–1127.

[Selman+, 1996] Selman, Bart, Kautz, Henry , and Cohen, Bram (1996).
Local search strategies for satisfiability testing.
In Johnson, David J. and Trick, Michael A., editors, *Cliques, Coloring, and Satisfiability: the Second DIMACS Implementation Challenge*, volume 26 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 521–532. American Mathematical Society.

Reference XXV

[Sinz, 2005] Sinz, Carsten (2005).
Towards an optimal CNF encoding of Boolean cardinality constraints.
In *Proceedings of the 11th International Conference on Principles and Practice of Constraint Programming (CP 2005), LNCS 3709*, pages 827–831.

[Soh+, 2017] Soh, Takehide, Banbara, Mutsunori , and Tamura, Naoyuki (2017).
Proposal and evaluation of hybrid encoding of CSP to SAT integrating order and log encodings.
International Journal on Artificial Intelligence Tools, 26(1):1–29.

[宋+, 2017] 宋 剛秀, 番原 睦則 , 田村 直之 (2017).
SAT 型制約プログラミングシステムと周辺技術.
コンピュータソフトウェア, 34(1):67–80.

Reference XXVI

[Soh+, 2014] Soh, Takehide, Berre, Daniel Le, Roussel, Stéphanie, Banbara, Mutsunori , and Tamura, Naoyuki (2014).
Incremental sat-based method with native boolean cardinality handling for the hamiltonian cycle problem.
In *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA 2014), LNAI 8761*, pages 684–693.

[Soh+, 2010] Soh, Takehide and Inoue, Katsumi (2010).
Identifying necessary reactions in metabolic pathways by minimal model generation.
In *ECAI 2010 - 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 2010, Proceedings*, pages 277–282.

Reference XXVII

[Tamura+, 2013] Tamura, Naoyuki, Banbara, Mutsunori , and Soh, Takehide (2013).
PBSugar: Compiling pseudo-boolean constraints to SAT with order encoding.
In *Proceedings of the 25th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2013), IEEE*, pages 1020–1027.

[田村+, 2016] 田村 直之, 宋 剛秀 , 番原 睦則 (2016).
SAT とパズル 〜問題をいかに SAT ソルバーで解くか〜.
情報処理, 57(8):710–715.

[Tamura+, 2009] Tamura, Naoyuki, Taga, Akiko, Kitagawa, Satoshi , and Banbara, Mutsunori (2009).
Compiling finite linear CSP into SAT.
Constraints, 14(2):254–272.

[田村+, 2010] 田村 直之, 丹生 智也 , 番原 睦則 (2010).
制約最適化問題と SAT 符号化.
人工知能学会誌, 25(1):77–85.

[丹生+, 2013] 丹生 智也, 田村 直之 , 番原 睦則 (2013).
位取り記数法に基づく整数有限領域上の制約充足問題のコンパクトかつ効率的な sat 符号化.
コンピュータソフトウェア, 30(1):211–230.

[梅村, 2010] 梅村 晃広 (2010).
SAT ソルバ・SMT ソルバの技術と応用.
コンピュータソフトウェア, 27(3):24–35.

[Velev+, 2009] Velev, Miroslav N. and Gao, Ping (2009).
Efficient SAT techniques for relative encoding of permutations with constraints.
In *Australasian Conference on Artificial Intelligence*, pages 517–527.

Reference XXIX

[Walsh, 2000] Walsh, Toby (2000).
SAT v CSP.
In *Proceedings of the 6th International Conference on Principles and Practice of Constraint Programming (CP 2000)*, pages 441–456.

[Warners, 1998] Warners, Joost P. (1998).
A linear-time transformation of linear inequalities into conjunctive normal form.
Inf. Process. Lett., 68(2):63–69.

[Wetzler+, 2014] Wetzler, Nathan, Heule, Marijn , and Jr., Warren A. Hunt (2014).
Drat-trim: Efficient checking and trimming using expressive clausal proofs.
In *Theory and Applications of Satisfiability Testing - SAT 2014 - 17th International Conference, Held as Part of the Vienna Summer of Logic,*

Reference XXX

VSL 2014, Vienna, Austria, July 14-17, 2014. Proceedings, pages 422–429.