

SATソルバを利用した お絵かきロジックの 問題作成支援ツール

長坂哲 伊藤寛之¹ 酒井正彦
草刈圭一郎 西田直樹 坂部俊樹

名古屋大学 大学院情報科学研究科

2010年03月01日

¹現在, (株) 富士通アドバンスソリューションズ

お絵かきロジック (1/2)

- マスの各行と各列における制約から元の二値画像を復元するパズル
 - 各行と各列の制約は、連続する黒マスの数の並び
- 問題例とその解：

		2	1	1	2				
		1	1	1	1	3	2		
		4	6	2	1	1	2	2	1
4									
2	2								
2	2								
8									
2									
2	2								
2	2								
4									

問題 (制約)

		2	1	1	2				
		1	1	1	1	3	2		
		4	6	2	1	1	2	2	1
4									
2	2								
2	2								
8									
2									
2	2								
2	2								
4									

解 (絵)

お絵かきロジック (2/2)

- 問題作成

- 解が一意であることが必要

- 一意にならない問題例：

	1	1
1	■	
1		■

	1	1
1		■
1	■	

- 問題自動生成が困難

- 解 (絵) の美しさが問題の評価の一つ
- 美しさは人間の感性に頼らなければならない

- 手作業での問題作成が困難

- 一意性となる問題の作成が困難

内容

- お絵かきロジック問題の解生成・一意性判定ツール (LogicSolver) の作成
 - SAT ソルバによるアプローチ
- お絵かきロジック問題作成支援ツール (LogicProblemMaker) の作成
 - LogicSolver を利用
 - 一意性の判定
 - 一意にならない原因となるマスの表示
 - 任意の画像からの問題作成

SAT ソルバ

- 充足可能性判定ツール (SAT ソルバ)
 - 入力は和積標準形 (CNF) の論理式
 - CNF の例 : $(x \vee y) \wedge \neg x \wedge (\neg x \vee z)$
 - 出力は充足可能, または, 充足不能
 - 充足可能な場合は解の一つを表示
 - 上の解の例 : x : 偽, y : 真, z : 偽
 - 近年高速化
 - 数 100 万の変数を持つ論理式も判定可能な場合がある
- お絵かきロジックの問題作成支援に SAT ソルバは有効かどうか为目的の一つ

問題から論理式へのコーディング (1/2)

- 各マスに対して論理変数 ($x_1 \sim x_{15}$) を用意

		3	3	2	1	1
2	1	x_1	x_2	x_3	x_4	x_5
	3	x_6	...			
	5				...	x_{15}

- 各論理変数 x_n : マスの状態

- x_n が真 \Rightarrow 黒マス
- x_n が偽 \Rightarrow 白マス

- 各行、列ごとにコーディング
- 例：

— 一行目をコーディング

2	1	x_1	x_2	x_3	x_4	x_5
---	---	-------	-------	-------	-------	-------

問題から論理式へのコーディング (2/2)

- 一行目をコーディング
- 組み合わせは3通り

2	1	x_1	x_2	x_3	x_4	x_5
---	---	-------	-------	-------	-------	-------

2	1	■	■	□	■	□
---	---	---	---	---	---	---

2	1	■	■	□	□	■
---	---	---	---	---	---	---

2	1	□	■	■	□	■
---	---	---	---	---	---	---

- 組み合わせの論理式表現
 - $$\begin{aligned}
 & (x_1 \wedge x_2 \wedge \neg x_3 \wedge x_4 \wedge \neg x_5) \vee (x_1 \wedge x_2 \wedge \neg x_3 \wedge \neg x_4 \wedge x_5) \vee \\
 & (\neg x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4 \wedge x_5)
 \end{aligned}$$
- 組み合わせ爆発が発生

コーディングの改良 (1/3)

- 改良前の変数に加え新たな変数の導入
 - 制約の各数に対して塗る位置を表す変数の導入
 - y_{ij} : i 番目の制約の j 番目の塗る候補位置を表す
- 一行目を再びコーディング

2	1	x_1	x_2	x_3	x_4	x_5
---	---	-------	-------	-------	-------	-------

2 が 1 番目の制約,

1 が 2 番目の制約

塗り方はそれぞれ 2 通り

$$\begin{cases} 2 : y_{11}, y_{12} \\ 1 : y_{21}, y_{22} \end{cases}$$

		y_{11} :真		y_{21} :真	
2	1	■	■	■	□

		y_{11} :真		y_{22} :真	
2	1	■	■	□	■

		y_{12} :真		y_{22} :真	
2	1	□	■	■	■

コーディングの改良 (2/3)

$$\begin{cases} 2: & y_{11}, y_{12} \\ 1: & y_{21}, y_{22} \end{cases}$$

2	1	x_1	x_2	x_3	x_4	x_5
---	---	-------	-------	-------	-------	-------

		y_{11} :真		y_{21} :真	
2	1	■	■	□	■
		y_{11} :真		y_{22} :真	
2	1	■	■	□	■
		y_{12} :真		y_{22} :真	
2	1	□	■	■	■

- (1) y_{1j}, y_{2j} のそれぞれで, ちょうど一つが真
 $y_{11} \vee y_{12}, y_{21} \vee y_{22}$ (少なくとも一つが真)
 $\neg y_{11} \vee \neg y_{12}, \neg y_{21} \vee \neg y_{22}$ (高々一つが真)
- (2) y_{ij} が真ならば, ある $y_{(i+1)k}$ が真 (ただし $j \leq k$)
 $y_{11} \rightarrow (y_{21} \vee y_{22}), y_{12} \rightarrow y_{22}$
- (3) 新たな変数 ($y_{11}, y_{12}, y_{21}, y_{22}$) とマス ($x_1 \sim x_5$) との関係
 $x_1 \leftrightarrow y_{11}, x_2 \leftrightarrow (y_{11} \vee y_{12}), x_3 \leftrightarrow y_{12}, x_4 \leftrightarrow y_{21},$
 $x_5 \leftrightarrow y_{22}$

コーディングの改良 (3/3)

- 改良後の論理式

$$\frac{(y_{11} \vee y_{12}) \wedge (y_{21} \vee y_{22}) \wedge (\neg y_{11} \vee \neg y_{12}) \wedge (\neg y_{21} \vee \neg y_{22})}{(1)} \wedge$$

$$\frac{(y_{11} \rightarrow (y_{21} \vee y_{22})) \wedge (y_{12} \rightarrow y_{22})}{(2)} \wedge$$

$$\frac{(x_1 \leftrightarrow y_{11}) \wedge (x_2 \leftrightarrow (y_{11} \vee y_{12})) \wedge (x_3 \rightarrow y_{12}) \wedge (x_4 \rightarrow y_{21}) \wedge (x_5 \rightarrow y_{22})}{(3)}$$

- 全ての行と列にコーディングを適用

実装・実験 (1/3)

- 論理式から CNF への変換
 - Tseitin 変換を利用 (新たな変数の導入により爆発せずに CNF へ変換)
- 実装
 - CNF 生成 : SML/NJ , SAT ソルバ : MiniSat v1.14
- 実験環境
 - CPU: Intel Pentium III-S 1.26GHz Dual,
Memory: 2GB, OS: FreeBSD 6.2-RELEASE
- 実験内容
 - 既存の問題 (3 種類) を改良前と後の方法で求解
 - 問題のサイズ :
(1) 30×30 , (2) 50×40 , (3) 70×180

実装・実験 (2/3)

- 実験結果

- サイズ : 30 × 30

	節の数	変数の数	CNF 生成 (秒)	SAT ソルバ (秒)
改良前	6,592,894	1,014,377	62	423
改良後	29,922	3,337	0.2	0.1

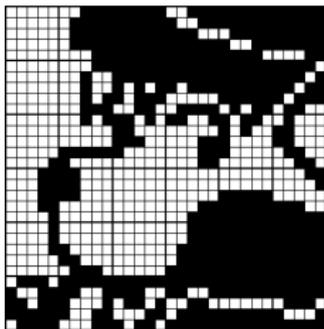
- サイズ : 50 × 40

	節の数	変数の数	CNF 生成 (秒)	SAT ソルバ (秒)
改良前	–	–	error	–
改良後	114,148	9,443	0.9	0.4

- サイズ : 70 × 180

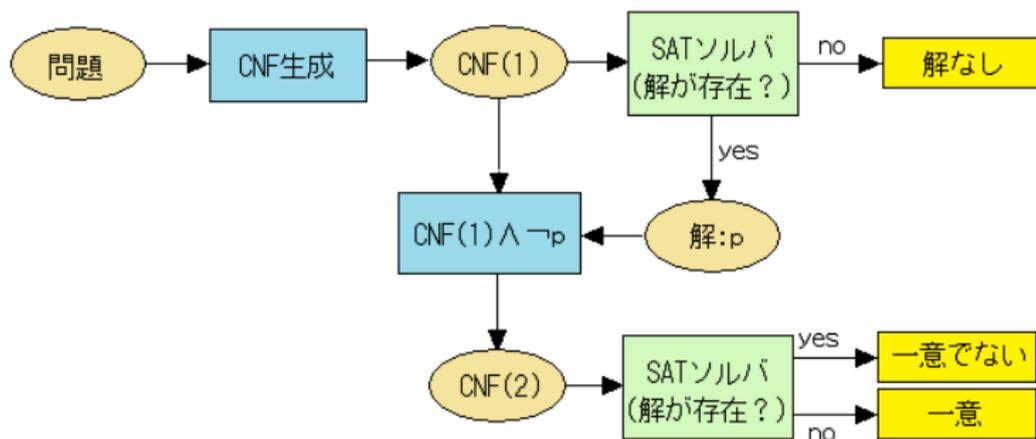
	節の数	変数の数	CNF 生成 (秒)	SAT ソルバ (秒)
改良前	–	–	error	–
改良後	2,767,447	122,006	32	7.9

実装・実験 (3/3)

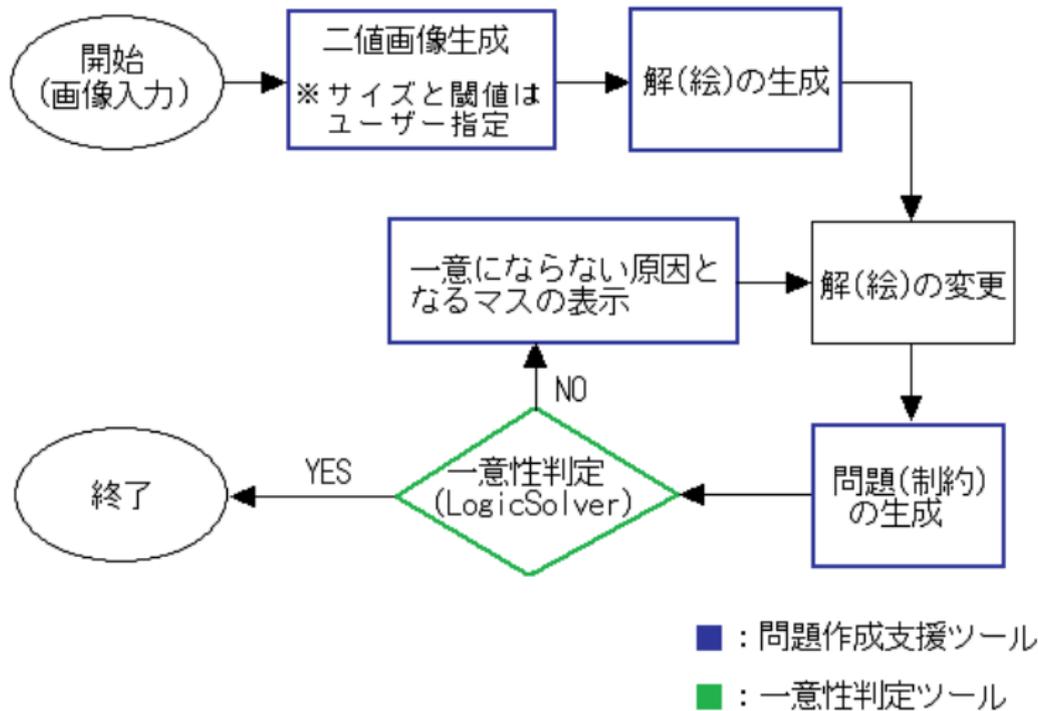


一意性判定

- CNF 生成と SAT ソルバでの判定を 2 回実行
 - 2 回目の CNF 生成では, 1 回目の SAT ソルバの解を利用
 - 2 回目の判定が充足不能 \Rightarrow 一意性を持つ



問題作成支援ツールの設計



問題作成支援ツールの実装

- 開発環境
 - Visual C++
- 入力画像のファイル形式
 - BMP, JPEG, EXIF, GIF, PNG, TIFF
- デモ

比較・評価

- 比較 (フリーツール5種類)
 - 一意性判定機能
 - 判定機能を持つ他ツールが存在
 - 本ツールより高速な機能を持つ他ツールが存在
 - 正確な判定機能は自ツールだけ
 - 画像からの問題生成機能
 - 二値画像からの問題生成機能を持つ他ツールが存在
 - カラー画像が入力可能なのは本ツールのみ
- 評価
 - 本ツールはお絵かきロジックの問題作成支援に有効
 - 高速, かつ, 正確な一意性判定機能
 - カラー画像を入力とした問題生成機能

まとめ・今後の課題

- まとめ

- お絵かきロジック問題の一意性判定ツール，
および，問題作成支援ツールの作成
 - SAT ソルバの利用
- SAT ソルバはお絵かきロジックの問題作成支援
に有効
 - 高速，かつ，正確に一意性判定可能

- 今後の課題

- 一意性判定の高速化
 - 新たな CNF へのコーディング方法の提案
 - 他の SAT ソルバとの比較
- 問題作成支援ツールの機能拡張
 - 一意性のための機能拡張
- カラーロジックへの対応