数独パズルの計算機による解析について

北本, 卓也
山口大学

出版情報：COE Lecture Note. 49, pp.78-82, 2013-08-09. 九州大学マス・フォア・インダストリ研究所
バージョン：
権利関係：

数独パズルの計算機による解析について
On the Analysis of Sudoku Puzzles by Computers

北本　卓也

Takuya Kitamoto*

山口大学

Yamaguchi University

Abstract

This paper presents a method to analyze Sudoku puzzles by computers. Generally speaking, it is not so difficult to solve Sudoku puzzles by computers, and many programs to solve Sudoku puzzles are avaiable. However, most of the programs use recursion and backtracking, which is significantly different from methods used by a human. Hence, a human-like method to solve a Sudoku puzzle is unknown even if the solution of the puzzle is computed by compute programs.

We created a computer program which solves Sodoku program with human-like methods. The program employs three basic techniques and solve almost all Sudoku puzzle in published books and magazines.

1　はじめに

数独パズルと呼ばれるバズルが流行しており、書店に行くとそのパズルの雑誌や書籍がたくさんおいてある。このパズルは日本だけでなく、世界中で流行しており、世界大会も行われている。

計算機を用いて、この数独パズルを解く試みが行われている。数学的な手法を用いた方法としては、Boolean Groebner Bases を用いた方法（[1], [2]）や、マルコフ基底を用いた方法（ [3]）などがあるが、世の中に出ている数独パズルを解くフリーソフトのほとんどは「仮置き」と呼ばれる単純な方法を用いている。「仮置き」を用いればどんなパズルも解くことができるが、その方法は通常、人間が手でパズルを解く時の手法とは異なっている（人が手でパズルを解くときには、仮置きは論理的でなく避けるべき手法だと考えられている）。このため、コンピュータソフトを用いればどんなパズルでもその答えはわかるが、どうやればそのパズルが解けるかまではわからない。

そこで、なるべく人間が行うものに近い方法でパズルを解くプログラムを作成した。本稿では、そのプログラムの解法アルゴリズムと、それを用いて実際のパズルを解かせた結果を報告する。

2　数独パズルとは？

数独パズルは 9×9 マスからなるパズルであり、各マスには 1 〜 9 までの数字が入る。また、各列、各行、各サブブロック（9×9 マスを 9 つの 3×3 のマスの塊に分けたもの）には 1 〜 9 までの数字が 1 つずつ入る。あらかじめ、いくつかのマスが数字で埋められた 9×9 マスが与えられたとき、上のルールのもとでまだ埋まっていないマスの数字を決めていくパズルである。

*kitamoto@yamaguchi-u.ac.jp

3　計算機による数独パズルの表現

プログラム内では、数独パズルを 9×9 の行列（行列の各要素はパズルの各マスに当たる）で表し、行列の各要素を 1〜9 までの数字を要素とする集合（例えば $\{1,3,8,9\}$）とする。この集合の要素は、そのマスの数字として可能性がある数字の集合であり、先ほど挙げたルールを用いてこの数字を削除していくことがパズルを解くことである。最終的に集合の要素が 1 つになると、そのマスの数字が確定する。全てのマスの数字が確定すれば（すなわち、全てのマスの集合の要素数が 1 になれば)、パズルが解けたことになる。以下では、$\mathrm{A}_{i,j}$ で (i,j) マスの数字の候補を表すことにする。

4　解法アルゴリズム

３つの基本テクニックを用いてパズルを解いていき、それでも解けない場合には、仮置きを用いる。

4.1　基本テクニック 1 — n 国同盟

これは、一般に 2 国同盟、3 国同盟と呼ばれているテクニックの一般化であり、[4] で解説されている方法である。まず、2 国同盟について説明する。例で説明した方が早いので例で説明する。今、$\mathrm{A}_{1,1} = \mathrm{A}_{1,4} = \{1,2\}$ であったとする。このとき、$\mathrm{A}_{1,2}, \mathrm{A}_{1,3}, \mathrm{A}_{1,5}, \mathrm{A}_{1,6}, \mathrm{A}_{1,7}, \mathrm{A}_{1,8}, \mathrm{A}_{1,9}$ から $\{1,2\}$ を取り除くことができる。何故ならば $\mathrm{A}_{1,1} = \mathrm{A}_{1,4} = \{1,2\}$ とすると、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}$ のどちらかが 1 でもう一方が 2 である。$\mathrm{A}_{1,1} \sim \mathrm{A}_{1,9}$ には 1,2 が 1 回ずつしか出ないため、$\mathrm{A}_{1,2}, \mathrm{A}_{1,3}, \mathrm{A}_{1,5}, \mathrm{A}_{1,6}, \mathrm{A}_{1,7}, \mathrm{A}_{1,8}, \mathrm{A}_{1,9}$ が 1, 2 を含む可能性がなくなるからである。3 国同盟も同様である。この場合は 3 つのマスが $\{1,2\}$, $\{2,3\}$, $\{1,3\}$ の場合にも使えるので 2 国同盟よりも気づきにくい。

n 国同盟は、これを n マスに拡張したものである。具体的には、次のように述べることができる。$j_1, \cdots, j_n$ を 1 〜 9 までの数字から取った n (<9) 個の数字の組だとするとき、

$$\mathrm{A}_{i,j_1} \cup \cdots \cup \mathrm{A}_{i,j_n} = \{k_1, \cdots, k_n\} \text{　ならば、} \mathrm{A}_{i,r}\ (r \neq j_1, \cdots, j_n) \text{ から } k_1, \cdots, k_n \text{ を削除できる。} \tag{1}$$

上は行に対する n 国同盟を述べているが、もちろん、各列、各サブブロックにも同じことが言える。

4.2　基本テクニック 2 — AB 簡約

これは、一般に X-Wing や Swordfish と呼ばれているテクニックの一般化である。X-Wing について例を用いて説明する。今、$1 \notin \mathrm{A}_{1,j}$ $(j = 2,3,5,6,7,8,9)$ かつ $1 \notin \mathrm{A}_{2,j}$ $(j = 2,3,5,6,7,8,9)$ が成り立っているとすると、$\mathrm{A}_{i,1}$ $(i = 3, \cdots, 9)$ と $\mathrm{A}_{i,4}$ $(i = 3, \cdots, 9)$ から 1 を除くことができる。これは次のような理由からである。与えられた条件より $\mathrm{A}_{1,1}, \mathrm{A}_{1,4}$ のうち 1 つが 1 であり、$\mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のうち 1 つが 1 である。よって、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}, \mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のうち、２つが 1 である。$\mathrm{A}_{i,1}$ $(i = 1, \cdots, 9)$ と $\mathrm{A}_{i,4}$ $(i = 1, \cdots, 9)$ のうち、1 であるものは２つであり、$\mathrm{A}_{1,1}, \mathrm{A}_{1,4}, \mathrm{A}_{2,1}, \mathrm{A}_{2,4}$ のいずれかなので、$\mathrm{A}_{i,1}$ $(i = 3, \cdots, 9)$, $\mathrm{A}_{i,4}$ $(i = 3, \cdots, 9)$ は 1 ではない。これを一般化すると、次のようになる。

$$\text{領域 } \mathrm{A}, \mathrm{B} \text{ が同じ個数の } k \text{ を含むならば　} k \notin \mathrm{A} \cap \mathrm{B}^C \Rightarrow k \notin \mathrm{A}^C \cap \mathrm{B} \tag{2}$$

先の例は、$k = 1$, $\mathrm{A} = \mathrm{A}_{1,1} \cup \cdots \cup \mathrm{A}_{1,9} \cup \mathrm{A}_{2,1} \cup \cdots \cup \mathrm{A}_{2,9}$, $\mathrm{B} = \mathrm{A}_{1,1} \cup \cdots \cup \mathrm{A}_{9,1} \cup \mathrm{A}_{1,4} \cup \cdots \cup \mathrm{A}_{2,4}$ としたものである。ちなみに 領域 A, B として、それぞれ３つの行と３つ列を取ったものが、Swordfish と呼ばれているテクニックである。また、領域の取り方としては、行、列の他にもサブブロックを取ることができる。

4.3　基本テクニック 3 — チェーン

これは、一般に Simple Chain や XY Chain と呼ばれているテクニックの一般化である。これらのチェーン系のテクニックは上級のテクニックとみなされており、他のテクニックが使えない場合も有効なことが多いが、考え方は仮置きに近い。

例を挙げる。$A_{1,1} = \{1, 2, 3\}$, $A_{1,4} = \{1, 2\}$, $A_{4,4} = \{1, 2\}$, $A_{4,2} = \{1, 2\}$, $A_{3,2} = \{1, 2\}$ とすると、$A_{1,1}$ から 1, 2 を除くことができる（すなわち、$A_{1,1} = \{3\}$）。何故ならば、今、仮に $A_{1,1} = \{1\}$ と仮定すると、$A_{1,4} = \{2\}$ である（第 1 行に 1 は 1 つだけ）。以下、同様に $A_{4,4} = \{1\}$, $A_{4,2} = \{2\}$, $A_{3,2} = \{1\}$ となり、$A_{1,1}$ のサブブロックに 1 が 2 つある（$A_{1,1} = A_{3,2} = \{1\}$）ことになってしまう。$A_{1,1} = \{2\}$ と仮定しても同様である。この例での

$$A_{1,1} \to A_{1,4} \to A_{4,4} \to A_{4,2} \to A_{3,2} \to A_{1,1}$$

のつながりをチェーンと呼ぶが、実際のプログラムでは、再帰的にこのようなチェーンを探し、矛盾が生じないかをチェックしている。チェーンの検索は、同じ行、同じ列、同じサブブロックにある同じ数字、もしくは同じマスの別の数字の候補に対して再帰呼び出しを用いて行なっているが、無制限に検索すると時間がかかりすぎるため、再帰呼び出しの深さに制限を設けている（今の所は 10 に設定）。

5　実験

前節の解法アルゴリズムを実行するプログラムを *Mathematica* で実装した。数独の問題を入力しやすいように図 1 にあるような問題入力用のインターフェイスを備えている。

出版されている数独パズルに関する書籍のうち、難しい問題が掲載されていると思われる書籍 [5] - [9] を選び、このプログラムで解答させた所、基本テクニック 1 〜 3 の範囲で問題が解け、仮置きを必要とするものはなかった。もちろん、全てのパズルが仮置きなしで解けるわけではなく、図 2 の「世界で最も難しいと言われているパズル」をはじめ、仮置きを必要とするパズルもたくさんある。特に海外のサイトにあるパズルには難しいものが多いようである。例えば [10] には、本稿のプログラムでは仮置きを必要とするパズルがたくさんある。

6　結論

人間に近い方法で数独パズルを解くプログラムを作成し、実際の問題を解いた。そのプログラムでは、まず 3 つの基本テクニックでパズルを解き、それでは解けない場合には仮置きを用いる。出版されている書籍にある問題は仮置きなしで解くことができたが、インターネットのサイト（特に海外のもの）には、仮置きを必要とするものが多く存在する。今回のプログラムでは、問題を解く際の各テクニックの利用状況がわかるので、次のような応用が考えられる。

- 与えられた問題の難易度判定
- 与えられた問題の解法の解説作成
- 与えられた問題の面白さの判定

今後は、これらの応用とともに、問題作成などについても取り組んでいきたい。

図 1: 問題入力用のインターフェイス

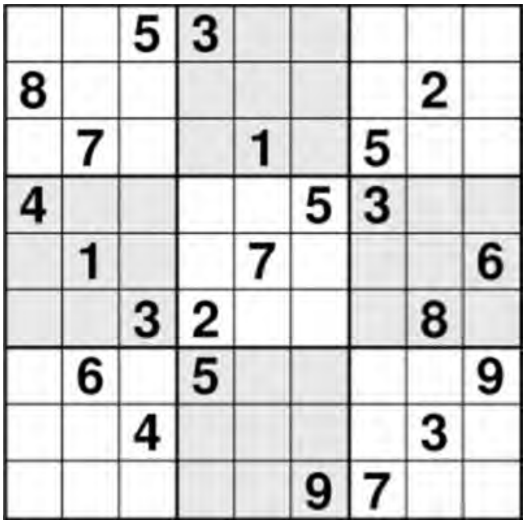

図 2: 世界で最も難しいと言われているパズル

参　考　文　献

[1] 井上秀太郎, 佐藤洋祐, 鈴木晃, 鍋島克輔: ブーリアン・グレブナ基底を使った数独の解法, 数理解析研究所講究録, **1666**, pp. 1-5, 2009.

[2] 井上秀太郎, 佐藤洋祐: グレブナー基底を使った数独の難易度判定と問題作成, 数理解析研究所講究録, **1785**, pp. 51-56, 2012.

[3] 中山洋将: Asir でのマルコフ基底計算とマルコフ基底によるパズルの解の列挙, *Risa/Asir Conference 2012*, 神戸, 2012.

[4] J. F. Crook: A Pencil-and-Paper Algorithm for Solving Sudoku Puzzles, *Notices of the American Mathematical Society*, **56**(4), pp. 460-468, 2009.

[5] 西尾徹也: ナンプレ超上級編 29, 世界文化社, 2012.

[6] ニコリ: ポケット数独 11 上級篇, ソフトバンククリエイティブ, 2012.

[7] たかせあきひこ: 難問ナンプレ 252 題 vol.15, 白夜ムック, 2012.

[8] ナンプレ研究会: 究極難解ナンプレ 8, 晋遊舎, 2013.

[9] 川崎光徳: 超難問ナンプレ 130 選, 永岡書店, 2012.

[10] http://www.sudokuwiki.org/sudoku.htm