

# ヒントの少ない数独問題の確率的探索

古川 湧 山本 修身

名城大学 理工学部 情報工学科

## 1. はじめに

数独は、 $9 \times 9$  マスから成るペンシルパズルの 1 つであり、マスには 1 から 9 のいずれかの数字がはいり、同じ行と列、9 つに分けられた  $3 \times 3$  のブロックの中に同じ数字が出現しないように数を入れる。また、初期にあらかじめ与えられている数（ヒント）を変えてはいけぬ。この条件の下で解が唯一であるものが数独の問題と定義される。本研究ではヒントの数が最小の数独の問題を求めようを試みた。数独の問題は 16 個以下のヒントでは構成することができないことが知られている [1]。すでにヒント数 17 の問題はいくつか求められているが、本研究では確率的な手法により自動的にこのような問題を見つけ出すことを目標としている。

## 2. 提案する探索アルゴリズム

まず、ヒント数が 16 以下の十分にヒントを減らした状態を用意する。この状態から徐々にヒントを増やし、ヒント数 17 の時点で解が唯一になるようなヒントを探索する。ヒントを追加したときの解の個数を最小化するため、複数存在する解を求めて最も出現回数の少なかったマスと数字を調べ、新しいヒントとして埋める。これを解が唯一になるまで繰り返して、ヒント数の少ない問題を探る方法を試みた。ヒント数を減らした問題の複数解を求めるときに以下の二つの探索方法を行った。

### 2.1 バックトラックを用いた探索

バックトラックは、与えられた問題に数字を総当たり的に埋めていき、解がないと分かれば解のある根まで戻って探索するという木探索アルゴリズムである。これを使い現在の問題からすべての解を求めることができるが、ヒント数を過剰に減らすと解の数が膨大になり、現実的な時間では解けない欠点がある。

### 2.2 Simulated Annealing(SA) を用いた探索

SA は、現在の状態から処理を行い次の状態に遷移したときに好ましい方向に向かった場合に温度変数  $T$  を下げ最適解を見つけるアルゴリズムである [2]。本実験では問題の空白に適当な数字を入れ、それらの数字を入れ替え、行と列、 $3 \times 3$  のブロックに存在する数字の種類が多くなるほど温度が下がるような関数を使用した。SA は現在の問題のヒント数が過剰に少なく膨大な解があると予想される問題において、ランダムに 1,000 個の解を求める探索である。

### 2.3 探索アルゴリズムの概要

予想される解の数によって解の探索方法を使い分ける。バックトラックと SA を用いた探索アルゴリズムの概略を図 1 に示す。関数や変数はそれぞれ  $H$ : ヒントの集合、 $S$ : 解の集合、 $\#H$ :  $H$  の要素数、 $th$ : バックトラックか SA を選択するヒント数の閾値、 $SASolution(H)$ : SA を用いて 1,000 個の解を求める関数、 $BTSolution(H)$ : バックトラックを用いてすべての解を求める関数、 $LeastPosAndNum(S)$ : 最も出現回数の少ない場所と数字を渡す関数である。

## 3. 探索の結果

SA とバックトラックを用いて問題を探した結果、54 分かけて図 2 に示すようなヒント数 18 の問題を探した。

```
function SearchWithHints(H, th)
while true do
  if(#H <= th) S = SASolution(H)
  else S = BTSoluton(H)

  if(#S == 1) return H
  else if (#S == 0) return FAIL

  (pos, no) = LeastPosAndNum(S)
  H = H ∪ {(pos, no)}
done
end
```

図 1: 本アルゴリズムの概略。

		8						
6				3	7			1
			1					9
								6
	9							
		5			4	2		8
		9						3
				4				
		2			8			

図 2: 生成したヒント数 18 の問題。

ことができた。実行した C 言語のソースコードは 600 行になり、実行環境は OS: Windows10Pro, コンパイラ: gcc 6.3.0 (MinGW), CPU: Intel i5-6200U@2.40GHz, メモリ: 8GB を用いた。

## 4. まとめ

初期配置におけるヒント数の少ない問題を SA とバックトラックを用いて探索し、ヒント数 18 の問題を探することに成功した。課題として初期に決定するヒントの選り方や、探索手法を変えることでヒント数 17 の問題を生成できないか模索する必要がある。

### 参考文献

- [1] G. McGuire, B. Tugemann, G. Civario There is no 16-clue Sudoku: Solving the Sudoku minimum number of clues problem via hitting set enumeration. *Experimental Mathematics*, 23:2, pp. 190-217 (2014)
- [2] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi: Optimization by simulated annealing. *Science*, New Series, Vol. 220, No. 4598, pp. 671-680 (1983)