

AAA Project

Solving Sudoku using the backtracking algorithm

800107 - Daniel Ng See Cheong
839521 - Uzair Moolla

September 12, 2016

1 Introduction

Sudoku is a numerical based logic puzzle game. The idea is to fill in an $n \times n$ grid with numbers in a specific manner. The most commonly known grid usually consists of square blocks with 3 rows and 3 columns. These individual blocks are then arranged in a similar manner again, with 3 blocks along rows and 3 along columns, producing a 9×9 matrix. An example of this described matrix can be seen in Figure 1. We need to fill each of the blocks with one of these numbers 1, 2, 3, 4, 5, 6, 7, 8, 9. The rules for filling these blocks are as follows:

- Within each 3×3 block we can only have single occurrence of each number.
- Along each 9 block row we can only have a single occurrence of each number.
- Along each 9 block column we can only have a single occurrence of each number.

An example of a Sudoku Puzzle completed using the rules stated above can be seen in Figure 2.

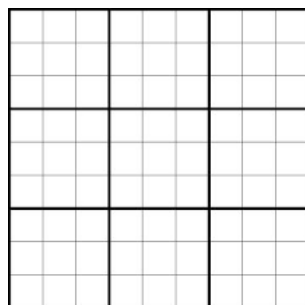


Figure 1: Sample of a 9×9 Blank Sudoku Board

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

Figure 2: Sample of a 9×9 Completed Sudoku Board

Solving sudoku puzzles are a fun exercise for many. However we would like to find a suitable algorithm which can be used to solve sudoku puzzles correctly as well as efficiently. There are various possible methods which can be used to solve sudoku puzzles. We will be focussing on the *Backtracking Algorithm* in our experiments.

#Fill in facts#

2 Aims

We are going to use the backtracking algorithm to solve sudoku puzzles of size $n \times n$. We will attempt to solve these sudoku puzzles in as little time as possible. Once we have the algorithm working correctly we will perform both empirical as well as theoretical analysis on it. We aim to find the best, average and worst case complexities of the algorithm being used.

#add some stuff#

3 Summary of Theory

4 Pseudo code

Algorithm 1 Backtracking

```
function NAME(params)
  if empty then
    #fill in#
  else
    continue
  return something
```

5 Experimental Methodology

6 Presentation of results

7 Interpretation of results

8 Conclusion

9 References

10 Acknowledgements