Investigation of Algorithmic Solutions of Sudoku Puzzles

## Investigation of Algorithmic Solutions of Sudoku Puzzles

The game of Sudoku as we know it was first developed in the 1979 by a freelance puzzle named Howard Garn.(History) Sudoku is a game that is traditionally played on a $9 \times 9$ grid. The goal of the game is to have all numbers 1-9 in each column, row, and in each of the $93 \times 3$ grids that are put together to form the $9 \times 9$ grid, known as boxes. If the puzzle is not $9 \times 9$ the numbers that are used to fill in the blank spaces must match the number of spaces in both the column and row. Each puzzle begins with some values placed in the squares with the purpose of the puzzle solver to fill in the rest of the squares. Since Sudoku follows a set of rules following a set of logic, it is possible to represent this logic as an algorithm. Aspects of this algorithm can be used to understand more about Sudoku as a whole, such as the application of tricks to solve puzzles more efficiently.

Sudoku puzzles become more complex for each row and column added. The number of rows and columns in a given puzzle has to be a square or even number because this is the only way to have boxes with the same number of spaces. An example of this issue can be demonstrated by attempting to create a $5 \times 5$ Sudoku puzzle. Since this puzzle is neither a square number or even it is shown that the boxes of the $5 \times 5$ Sudoku puzzle are not able to have exactly 5 spaces without the boxes being uneven.

| 1 | 2 | 1 | 2 | 1 |
| :---: | :---: | :---: | :---: | :---: |
| 3 | 4 | 3 | 4 | 2 |
| 5 | 1 | 5 | 3 | 4 |
| 2 | 3 | 1 | 2 | 5 |
| 4 | 5 | 3 | 4 | 5 |

The exception to the rule that all even puzzles allow for evenly sized boxes is in a $2 x 2$ puzzle.

This is because in a $2 \times 2$ puzzle the only 2 solutions are as follows, and both of these possible solutions create boxes that are $1 \times 2$. All other even numbers create 4 boxes.

| 2 | 1 |
| :--- | :--- |
| 1 | 2 |


| 1 | 2 |
| :--- | :--- |
| 2 | 1 |

To understand how an algorithm for a puzzle can be created it is advantageous to start the investigation by analyzing the simplest puzzle possible. Since a $2 \times 2$ puzzle does not produce a valid puzzle, a $4 \times 4$ puzzle will be used instead.

| A1 | B1 | C1 | D1 |
| :---: | :---: | :---: | :---: |
| A2 | B2 | C2 | D2 |
| A3 | B3 | C3 | D3 |
| A4 | B4 | C4 | D4 |

The $4 \times 4$ grid has different locations that are represented by the coordinate system that represents horizontal changes as a change in letter and a vertical change by a change in number. This coordinate system is useful because it allows for the algorithm to identify specific locations on the puzzle. Other notation necessary to interpret the algorithm is the use of $x$ to define the state of a given space on the Sudoku board. When $x$ is referred to as undefined it means that a square has not been assigned a value.


Instead of showing the complete algorithm, it simply says repeat until location d4. The order at which you approach the spaces does not matter as long every space is analyzed. The possible solutions that are eliminated are those from the spaces that are in the same row, column, and box as the given space. This algorithm does not create a solution by itself, but rather tells the person completing the puzzle what possibilities can be eliminated from undefined squares, so some analysis is necessary after application of the algorithm. This algorithm also may require multiple applications until the puzzle is fully completed.This algorithm can be applied to any size Sudoku puzzle, with slight modifications. In order to do so, it is necessary to adjust the number of possible $x$ values, in the algorithm, according to the number of spaces in each row and column. It is also necessary to change the number of spaces analyzed and to also develop a labelling system that is appropriate to the size of the puzzle. This means that if a puzzle is $9 x 9$ then the number of $x$ values present would be 9 . This method is limited in special cases when limited information is provided and therefore creates an unsolvable state in the Sudoku puzzle. However this is a fringe case because if not all solutions are found on the first application of the algorithm, new information may reveal the true solutions on the second, third, fourth, etc. application of the algorithm. The method used in this algorithm is a process of elimination approach. Every time a specific value is given by the Sudoku puzzle that same value is eliminated from the possible solutions of the spaces in the same row, column, and box as the given value. A solution for the previously undefined spaces when all solutions but one are eliminated from a specific space.

Another possible method for reaching a solution would be to assign every space in the same row, column, and box of the given value, every possible solution for each undefined square. After analyzing every space, possible solutions will be eliminated because new information will contradict old information and create a more refined list of possibilities. The issue with this method is the fact that the amount of information present from the start of this solution method is much more than the information present at the start of the elimination method and therefore is a less practical method of
solution. This is because when a space is labelled as undefined, it will be given the possibility of being any number in the number set of that specific puzzle.

The algorithm as is, attempts to solve Sudoku in a sort of brute force manner where it ignores special case patterns of Sudoku and rather attempts to apply the same basic rule over and over until a final solution is reached. These special case patterns are known as tricks, and can allow for more efficient solving of a Sudoku puzzle. A popular Sudoku trick is to use what is known as a hidden single, in order to get the solution to a particular empty space.(Johnson)Normally a hidden single is spotted when there are multiple spaces in the same row, column, or box, that contain the same possible solutions, except that in one of the spaces there is one extra possible solution. An example of this is as follows:

| 4 | 7 | 1 | 5 |
| :--- | :--- | :--- | :--- |
| 3 | 8 | 1 | 5 |
| 2 | 1 | 1 | 5 |
|  | 9 |  | 9 |

In this $3 \times 3$ box each space on the right side of the box has the possibility of being either a 1,5 , or 9 but the middle space can also be a 6 . Through the use of the rule of hidden singles the space that also has a possibility of being a six is guaranteed to be a six. If tricks are factored into a Sudoku algorithm it is more advantageous to utilize the method that does not simply eliminate solutions. This is because most established Sudoku tricks are written for this method. It is most advantageous to only apply this strategy when applying the algorithm after the first application. This is because obvious solutions will be removed from the equation as well as hidden singles will be revealed because of the information
present in the puzzle. This is also because an algorithm to look for hidden singles will be applied after the base algorithm in order to eliminate possible solutions.

In conclusion, it is possible to create an algorithm for solving Sudoku is possible by solely utilizing the basic rules of Sudoku puzzles to complete, however to create more efficient methods it is advantageous to account for the puzzle as a whole. This is achieved through a number of checks that look for specific patterns in puzzles such as hidden singles. This method of adding to a base algorithm can be extended to factor in more tricks. This entirely depends on the complexity of puzzles being solved, as algorithms that factor for more tricks will be able to complete more complex puzzles, or puzzles that contain only minimal initial information.

## Works Cited

Johnson, Angus. "Solving Sudoku." Solving Sudoku. N.p., 2005. Web. 26

Feb. 2016.
"The History of Sudoku." Play Free Sudoku a Popular Online Puzzle Game. GameHouse, n.d. Web. 25 Feb. 2016.

## Assessment criteria:

Criterion

## Comments *

## Achievement level

The introduction seemed solid, but the paper never truly explored the
A Communication
aim of the paper. A simple algorithm needed more depth, with examples. The paper did adequately communicate the basics.

B Mathematical presentation

While the paper added a nice chart to show the flow of the algorithm, there was a lack of other representations to help the reader. In one case, the table was done sloppily and could have been done better.

C Personal engagement

Little to no mention was made of why the topic was chosen. There only superficial exploration, and the paper did not give the feel of an exploration, but more of an informational text.

D Reflection
Short mention was made near the end, but no relevant conclusions
were drawn, nor mentions of how an algorithm like this could be
applied elsewhere.

E $\quad \begin{aligned} & \text { Use of } \\ & \text { mathematics }\end{aligned}$

No relevant mathematics or calculations could be found. In fact, the only mathematics anywhere in the paper was counting rows and columns, and eliminating numbers from a grid.
Teacher Moderator Senior

| $0-4$ |  |
| :---: | :---: |
| 1 | $\square$ |

0-3
1


| $0-4$ |  |
| :--- | :--- |
| 2 | $\square$ |


| $0-3$ |  |
| :--- | :--- |
| 1 | $\square$ |

1

 $\square$ | $0-6$ |
| :---: |
| 0 |
|  |

Total:

$\square$
$\square$

