



Research Article

# A Binary Integer Programming Approach to find a Sudoku Matrix

Paramacutty Paramadevan \*

Department of Mathematics, Faculty of Science, Eastern University, Sri Lanka

Received: May 01, 2022; Revised: July 11, 2022; Accepted: July 27, 2022

**Abstract:** In the basic Sudoku game, players must fill the entries of a  $n \times n$  matrix, except some given entries, under the conditions that each row, each column, and each  $m \times m$  sub matrix contains integers 1 through  $n$  exactly once. In this research paper, in addition to the basic game, conditions and solution process for three advanced versions, such as Sudoku  $X$ , Four Square Sudoku and Four Pyramid Sudoku were also studied. In this study, the above-mentioned conditions are converted into appropriate mathematical forms as constraints of the Integer Linear Programming Problem using Pascal Programming Language, such a way that the objective function and the constraints suit as an input mathematical model for the LINGO mathematical optimization software. Here, an unlimited version of the LINGO software has been used with the built-in Branch and Bound algorithm as the number of constraints is very high. Finally, solutions to the given Sudoku problems obtained from the solutions acquired through LINGO software.

**Keywords:** Sudoku, Integer linear programming, Pascal programming, LINGO, Branch and Bound algorithm

## 1. Introduction

Sudoku is an interesting logic-based combinatorial number-placement puzzle game that ends up when finding a matrix whose elements are natural numbers that are less than or equal to a fixed number  $n$ . The general condition of this game is that a player has to fill a  $n \times n$  matrix with some given entries so that each row, each column, and each  $m \times m$  sub-matrix contains integers 1 through  $n$  exactly once.

Even though the Sudoku game was initiated in the United States of America (Garns, 1979), it became famous in Japan during the 1980s. Nowadays, we can observe that children and adults are showing interest in solving this game appearing in newspapers and online (Riley et al., (n.d); Pegg Jr, 2005).

Playing this game has several benefits: improving concentration, reducing anxiety and stress, promoting healthy mindsets, developing problem-solving skills, improving memory and thinking skills, etc.

In a common Sudoku problem, the basic rules are as follows:

- fill each element of a given  $n \times n$  matrix with natural numbers from 1 to  $n$  in which some elements already given
- each row contains numbers from 1 to  $n$  exactly once
- each column contains numbers from 1 to  $n$  exactly once

- certain sub matrices contain numbers from 1 to  $n$  exactly once.

In most common Sudoku problems (Bammel and Rothstein, 1975), the fixed number  $n$  is assumed to be 9, and the said sub-matrices are the nine  $3 \times 3$  sub-matrices located in specific nine double-lined locations.

In the fundamental Sudoku problem, the level of difficulty of the game depends on the number of givens (already filled entries), given values, and their locations (Bartlett et al., 2008). Solving the problem becomes more accessible when the number of givens is more than 30; meanwhile, it is observed experimentally that a problem with less than 17 givens has not been solved so far. Also, the difficulty level will increase in the advanced versions of the game. Two fundamental Sudoku problems are given in Figure 1, and their solution matrices are given in Section 3. In both problems given in Figure 1, out of 81 entries of the  $9 \times 9$  square matrix, 38 entries are given; our task is to fill the rest of the entries in such a way that the above basic conditions should be satisfied.

Some of the most popular advanced versions are Sudoku  $X$ , Four Square Sudoku, and Four Pyramid Sudoku. In these versions, there are additional specific conditions imposed to be satisfied.

In Sudoku  $X$ , the symbol  $X$  indicates the diagonal and anti diagonal entries of the square matrix. In this version, in addition to the basic conditions, the numbers from 1 to 9 have to appear in each diagonal exactly once. Two examples of Sudoku  $X$  games are given in Figure 2, and their solutions are given in Section 3. In



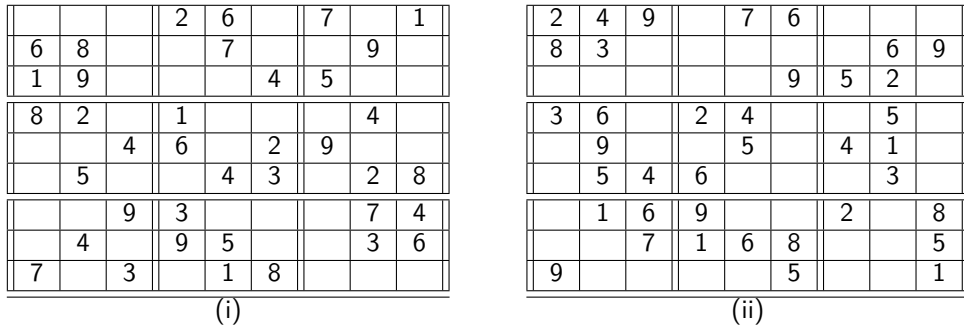


Figure 1: Examples of fundamental Sudoku problem

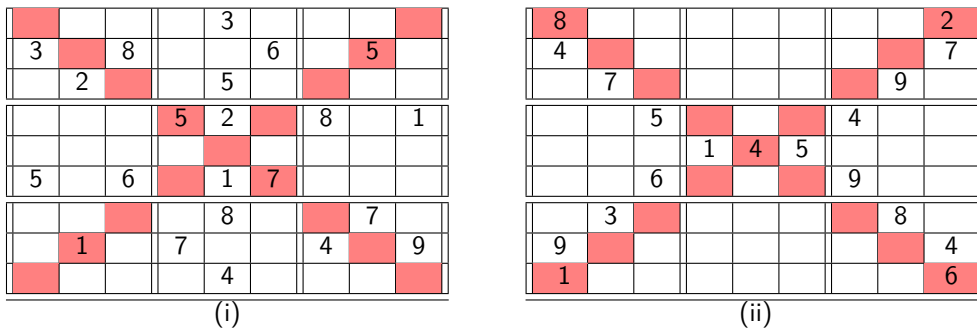


Figure 2: Examples of Sudoku X

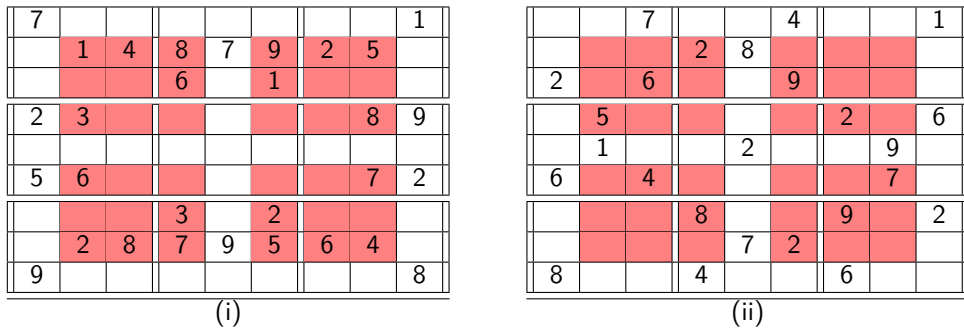


Figure 3: Examples of Four Square Sudoku

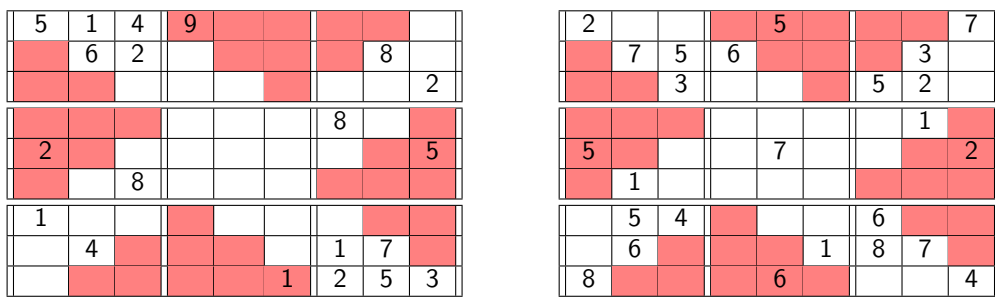


Figure 4: Examples of Four Pyramid Sudoku

Figure 2, out of 81 entries only, 22 entries are given. Our task is to fill the rest of the entries of the matrix in such a way that, in addition to the above basic conditions, the numbers from 1 to 9 have to appear in each diagonal exactly once.

In the Four Square Sudoku game, there are four specific highlighted squares. In this game, in addition to the basic conditions, nine entries of each square are filled so that each digit 1 to 9 appears exactly once. We may consider these four squares as another set of four  $3 \times 3$

sub-matrices that are different from those already considered in the basic conditions. Two examples of Four Square Sudoku games are given in Figure 3, and their solutions are given in Section 3. In Figure 3, out of 81 entries, 25 entries are given. Our task is to fill the rest of the matrix entries so that in addition to the above basic conditions, the numbers from 1 to 9 have to appear in each of the specific four squares exactly once.

In the Four Pyramid Sudoku game, in addition to the basic requirements, four specific highlighted pyramids must also contain each digit 1 to 9 exactly once. Two examples of Four Pyramid Sudoku games are given in Figure 4, and their solutions are also given in Section 3. Figure 4, also out of 81 entries, 25 entries are given. We should fill other matrix entries so that, in addition to the above basic conditions, the numbers from 1 to 9 must appear in each of the specific four pyramids exactly once.

This research paper discusses the fundamental Sudoku problem and three advanced version types. The rest of the article is organized as follows: mathematical models are formulated in Section 2, Section 3 illustrates how we can get solutions for the above problems, Section 4 concludes the article, and references are given at the end of the article.

### 2. Mathematical Model

Let us first define the decision variables of the Binary Integer Programming Problem (BIPP). The decision variables are defined in the association of the element of the Sudoku matrix, which is in the  $p^{th}$  row and  $q^{th}$  column. Let

$$a_{p,q,r} = \begin{cases} 1 & \text{if } (p,q)^{th} \text{ element is } r \\ 0 & \text{otherwise} \end{cases}$$

Note that the integers  $p, q$  and  $r$  less than or equal to the given integer  $n$ .

In any BIPP, we usually have an objective function from which we obtain the optimal solution to the problem. We are interested in obtaining a feasible solution here, so we do not need to pay much attention to the objective function.

Let us formulate the constraints of the BIPP. Constraints are nothing, but they are the mathematical form of the conditions said above, which are to be satisfied by the final Sudoku matrix.

1. Only one  $r$  in each row  $\sum_{q=1}^n a_{p,q,r} = 1,$   
 $p = 1, 2, \dots, n; r = 1, 2, \dots, n.$
2. Only one  $r$  in each column  $\sum_{p=1}^n a_{p,q,r} = 1,$   
 $p = 1, 2, \dots, n; r = 1, 2, \dots, n.$

3. Only one  $r$  in each sub matrix

$$\sum_{q=mt-m+1}^{mt} \sum_{p=ms-m+1}^{ms} a_{p,q,r} = 1,$$

$$r = 1, 2, \dots, n; s = 1, 2, \dots, n; t = 1, 2, \dots, n.$$

4. Only one of the integers from 1 to  $n$  will be the element of the final matrix  $\sum_{r=1}^n a_{p,q,r} = 1,$   
 $p = 1, 2, \dots, n; q = 1, 2, \dots, n$

5. Final matrix should contains given elements in the respective places.  
That is the  $(p,q)^{th}$  element is  $r$  for given  $p, q.$   
ie.  $a_{p,q,r} = 1$  for given  $p, q$  and  $r.$

6. Each  $a_{p,q,r}$  is a binary variable  
 $a_{p,q,r} \in \{0, 1\} \forall p = 1, 2, \dots, n; q = 1, 2, \dots, n; r = 1, 2, \dots, n.$

7. In addition to the above six constraints, only one of the following three set of constrain should be included to get the mathematical model for the respective advanced version. For our convenient 9 is substituted instead of  $n$  in the following constraints.

- (a) i.  $\sum_{p=1}^9 a_{p,p,r} = 1, r = 1, 2, \dots, 9.$

- ii.  $\sum_{p=1}^9 a_{p,(n+1-p),r} = 1, r = 1, 2, \dots, 9.$

- (b)  $\sum_{p=l}^{l+2} \sum_{q=k}^{k+2} a_{p,q,r} = 1, l = 2, 6; k = 2, 6;$   
 $r = 1, 2, \dots, 9.$

- (c) i.  $\sum_{p=1}^3 \sum_{q=3+p}^{9-p} a_{p,q,r} = 1, r = 1, 2, \dots, 9.$

- ii.  $\sum_{q=1}^3 \sum_{p=1+q}^{7-q} a_{p,q,r} = 1, r = 1, 2, \dots, 9.$

- iii.  $\sum_{p=7}^9 \sum_{q=11-p}^{p-3} a_{p,q,r} = 1, r = 1, 2, \dots, 9.$

- iv.  $\sum_{q=7}^9 \sum_{p=13-q}^{q-1} a_{p,q,r} = 1, r = 1, 2, \dots, 9.$

Pascal programming language is used to generate the above constraints in such a way that these constraints suit an input mathematical model for the LINGO mathematical optimization software. Once the constraints are formulated, accumulating all the constraints, a mathematical optimization model associated with the Sudoku problem is formed. This mathematical model involves 729 binary decision variables and 324 constraints. That is 81 constraints associated with each condition 1,2,3, and 4. The number of constraints associated with condition 5 is equal to the number of givens 25.

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

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

Figure 5: Solutions of fundamental Sudoku problem

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

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

Figure 6: Solutions of Sudoku X

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

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

Figure 7: Solutions of Four Square Sudoku

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

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

Figure 8: Solutions of Four Pyramid Sudoku

This problem was fed in the LINGO software and solved by using an Intel(R) Core(TM)i5 with a 4GB RAM computer. Finally, a solution to the Sudoku problem was obtained from the solution of the optimization problem using the LINGO. Here we used the Branch and Bound algorithm and unlimited version of the LINGO platform as the number of constraints is very high.

### 3. Results and discussions

The following general techniques may help him when a player tries to find a solution to any Sudoku problem.

- scanning in one direction
- scanning in both direction

- Searching for a single number
- eliminating numbers from rows, columns, and sub-matrices
- searching for missing numbers in rows and columns
- eliminating squares using naked pairs in rows, columns and sub-matrices

The solutions to fundamental Sudoku problems given in Figure 1 are shown in Figure 5. From Figure 5, we can observe that in the  $9 \times 9$  matrix, each row, each column, and each  $3 \times 3$  sub-matrix contains each integer 1 through 9 exactly once.

In Sudoku  $X$ , in addition to the 325 constraints considered in the fundamental Sudoku, 18 constraints are associated with the condition 7(a). The solutions to Sudoku  $X$  problems given in Figure 2 are shown in Figure 6. From Figure 6, we can observe that in addition to the above basic conditions, the numbers from 1 to 9 appeared in both diagonals exactly once. We can also notice some numbers repeatedly appeared in the diagonals in the solutions of fundamental Sudoku problems, where we had not considered the specific condition for Sudoku  $X$ .

In Four Square Sudoku, in addition to the 325 constraints considered in the fundamental Sudoku, 18 constraints are associated with the condition 7(b). The solutions to the Four Square Sudoku problems given in Figure 3 are shown in Figure 7. From Figure 7, we can observe that in addition to the above basic conditions, the numbers from 1 to 9 appeared in each of the specific four squares exactly once.

In Four Pyramid Sudoku, in addition to the 325 constraints considered in the fundamental Sudoku, 36 constraints are associated with condition 7(c). The solutions to the Four Pyramid Sudoku problems given in Figure 4 are shown in Figure 8. From Figure 8, we can observe that in addition to the above basic conditions, the numbers from 1 to 9 appeared in each of the specific four pyramids exactly once.

Further, in the introduction, it is emphasized that players can gain specific skills while playing the Sudoku game.

Then a question may arise: Why do we need an automated approach to solve this problem? One answer may be that when we play a Sudoku game, we may get stuck at some instance at several attempts and decide to leave the game with doubt about whether a solution exists or not. In this situation, this approach will help the player confirm whether an answer exists.

#### 4. Conclusion

This research paper developed the optimization models for fundamental Sudoku problems and its three advanced versions: Sudoku  $X$ , four square Sudoku, and four pyramid Sudoku. A process for finding a solution was also proposed by using two platforms, namely, Pascal and LINGO. In the first step, the Objective function and constraints were generated one by one using the pascal programming language. Secondly, the generated constraints were compiled with an Objective function to form a mathematical optimization model in such a way that it supports an input model for LINGO. Finally, the solution to our original Sudoku problem was obtained using the solution obtained from LINGO.

#### References

- Bammel, S.E. and Rothstein, J., 1975. The number of  $9 \times 9$  Latin squares. *Discrete Mathematics*, 11(1):93-95.
- Bartlett, A., Chartier, T.P., Langville, A.N. and Rankin, T.D., 2008. An integer programming model for the Sudoku problem. *Journal of Online Mathematics and its Applications*, 8(1).
- Garns, H., 1979. Number place. *Dell Pencil Puzzles & Word Games*, May, p. 6.
- Pegg Jr, E., 2005. Sudoku Variations. *MAA Online*, <https://www.maa.org/ed-pegg-jrs-math-games> (Accessed: 03.01.2022).
- Riley, P., Taalman, L. and Calvin, R., (n.d.). Brainfreeze Puzzles. *Brainfreezepuzzles.com*, <http://www.brainfreezepuzzles.com> (Accessed: 03.01.2022).