

Contents

Abstract	vi - vii
List of Figures	xiii - xix
List of Tables	xx
List of Abbreviations	xxi
Chapter 1: Introduction to Sudoku Problem	1 - 14
1.1 A Brief Background on Sudoku	3
1.1.1 Latin Square	4
1.1.2 Magic Square	4
1.1.3 Modern Sudoku	5
1.1.4 Sudoku and Medias	7
1.2 Difficulty Level of the Sudoku Puzzle	7
1.3 Motivations behind the Present Work	8
1.4 Objectives of the Present Work	9
1.5 Achievements out of the Work Done	10
1.6 Outline of the Thesis	14
Chapter 2: Existing Literature on Sudoku Puzzle: A Review	15 - 60
Overview	15
2.1 Nature of the Puzzle Problem	15
2.2 A Study on different Existing Sudoku Solving Techniques	17
2.2.1 Backtracking based methodology.....	20
2.2.1.1 Forward Checking	21
2.2.1.2 Constraint Propagation	21
2.2.1.3 Minimum Remaining Values	22

2.2.1.4 Hill Climbing	23
2.2.2 Elimination based Strategies	23
2.2.2.1 Unique Missing Candidate	24
2.2.2.2 Naked single	26
2.2.2.3 Hidden single	27
2.2.2.4 Lone ranger	27
2.3.2.5 Locked candidate	28
2.2.2.6 Twin	29
2.2.2.7 Triplet	29
2.2.2.8 Quad	30
2.2.2.9 X-Wing	30
2.2.2.10 XY-Wing	31
2.2.2.11 XYZ-Wing	32
2.2.2.12 Swordfish	34
2.2.2.13 Colouring	34
2.2.2.14 Forcing Chains	35
2.2.2.15 Tuleja's (or Mr. T's) Theorem	36
2.2.3 Soft Computing based Techniques	39
2.2.3.1 Stochastic Optimization Technique	39
2.2.3.2 Cultural Genetic Algorithm	42
2.2.3.3 Repulsive Particle Swarm Optimization	47
2.2.3.4 Quantum Simulated Annealing	49
2.2.3.5 Hybrid Genetic Algorithm and Simulated Annealing	52
2.2.3.6 Bee Colony Optimization	54
2.2.3.7 Artificial Immune System	56
2.3 Enumerating Sudoku Grids	57
2.4 Summary	59
Chapter 3: A New Sudoku Solver	61 - 89
Overview	61
3.1 Version 3.1 of the New Gussed Free Sudoku Solving Algorithm	63

3.2	Permutation Generation Technique	69
3.2.1	A brief Review on existing Permutation Generation Algorithm	70
3.2.1.1	Random Permutation Generation Algorithm	70
3.2.1.2	Fisher-Yates Shuffle	71
3.2.1.3	Permutation Tree	71
3.2.2	A Novel Permutation Generation Technique	73
3.3	Version 3.2 of the New Guessed Free Sudoku Solving Algorithm	76
3.3.1	The Algorithm (Version 3.2) at a Glance	80
3.3.2	A Brief Comparison between Version 3.1 and Version 3.2 of the Algorithm	83
3.3.3	Computational Complexity of the Algorithm.....	86
3.4	Summary.....	88

Chapter 4: A Novel Graph Theoretic Sudoku Solver 90 - 123

Overview	90	
4.1	GTSPS: A Graph Theoretic Approach for solving Sudoku Puzzle	90
4.2	Version 4.2 of the Algorithm: A Different Adaptation of GTSPS	101
4.3	The Algorithm at a Glance	103
4.4	Computational Complexity of the Algorithm	106
4.5	Experimental Results	109
4.5.1	Analysis of Experimental Results.....	114
4.5.2	Comparative Judgement among Different Sudoku Solvers	115
4.6	Relative Comparisons among the Versions of the Newly Devised Sudoku Solver.....	119
4.7	Summary	122

Chapter 5: Techniques for Generating Sudoku Instances 124 - 143

Overview	124	
5.1	Metrics of Difficulty Level.....	124
5.1.1	The Total Amount of Given Cells	125
5.1.2	The Lower Bound on the Number of Clues in Each Row, Column	125
5.2	Generating a Sudoku Instance from a Solved Sudoku Puzzle	126
5.2.1	Randomized Selection of Cell Location.....	127

5.2.2	Sequential Selection of Cell Location	128
5.2.2.1	Wandering along S (or Zigzag) Path	128
5.2.2.2	Wandering from Left to Right or the Reverse	128
5.2.2.3	Symmetrical Removal of Values from Rows	129
5.2.2.4	Symmetrical Removal of Values from Columns	130
5.2.2.5	Symmetrical Removal of Values from Minigrids	130
5.2.3	Flowchart at a Glance for the <i>Digging Hole</i> Strategy	132
5.3	Generation of Sudoku Instances by Transformation of Puzzles	132
5.3.1	Digit Exchanging	132
5.3.1.1	Replacement of all digits	134
5.3.2	Rotation	135
5.3.2.1	Rotation by 90 degree	135
5.3.2.2	Rotation by 180 degree	136
5.3.2.3	Flipping Vertical Rotation	137
5.3.2.4	Flipping horizontal rotation	138
5.3.3	Rows-in-a-Band Exchanging	138
5.3.4	Columns-in-a-Stack Exchanging	139
5.3.5	Band-Exchanging	140
5.3.6	Stack-Exchanging	141
5.3.7	A Combination of All Six Methods	141
5.4	Verifying the Uniqueness of the Solution	142
5.5	Summary	143

Chapter 6: Application of Solving Sudoku Instances 144- 172

Overview	144
6.1 Application Domains of Solving Sudoku Instances	144
6.2 A Novel Biometric Template Encryption Scheme using Sudoku Puzzle	147
6.2.1 Biometric System Vulnerabilities	149
6.2.1.1 Intrinsic Limitations	149
6.2.1.2 Adversary Attacks	150
6.2.2 Existing Biometric Template Encryption Scheme	150

6.2.3 Biometric Template Encryption Scheme using Sudoku	153
6.2.4 Biometric Template Decryption Technique	156
6.2.5 Experimental Results: Analysis and Discussion	157
6.3 A Novel Steganographic Scheme Using Sudoku	159
6.3.1 Existing Steganographic Scheme using Sudoku	161
6.3.1.1 Chang et al.'s method	161
6.3.2 A New Steganographic Scheme using Sudoku	164
6.3.2.1 Embedding of an 8×8 Sudoku matrix in a Cover Image	164
6.3.2.2 Embedding of Hidden Message	166
6.3.2.3 Checking the Integrity of the Cover Image	168
6.3.2.4 Hidden Message Extraction	168
6.3.2.5 Experimental Results and Analysis	171
6.4 Summary	172
Chapter 7: Conclusion and Future Scope	173 - 178
7.1 Status of the Problems Prior to this Work	173
7.2 Summary of the Work Done	174
7.3 Future Scope of the Work	177
Bibliography	179 - 184
Appendix A: Definitions of Some Important Terms	185 - 188
Appendix B: List of Publications	189 - 190

List of Figures

Figure Number	Figure Description	Page Number
1.1	Figure 1.1: (a) An instance of the Sudoku problem. (b) A solution of the Sudoku instance shown in Figure 1.1(a), where a digit / symbol occurs exactly once in each row, column, and minigrid..	2
1.2	(a) A Sudoku puzzle P. (b) Two solutions of the Sudoku puzzle P, in which the valid permutations that produce the solutions differ only in the first and second minigrid.	3
1.3	A Latin square where the numbers 0 through 3 are present in each row and column uniquely..	4
1.4	A Magic square where the numbers 1 through 9 are present exactly once in a 3×3 grid such that the sum of the digits in each row, column, and diagonal (or crossway of length three) equals to 15.	5
1.5	A pre-Sudoku in the newspaper <i>Le Siècle</i> published on November 19, 1892 [14].	5
1.6	A pre-Sudoku published in <i>La France</i> on July 6, 1895 [14].	6
2.1	The structure of a 9×9 Sudoku puzzle (problem) with its nine minigrids of size 3×3 each as numbered (in grey outsized font) 1 through 9.	19
2.2	A size two (2×2) Sudoku puzzle for executing constraint propagation.. ...	22
2.3	A size three (3×3) Sudoku puzzle with minimum remaining values.....	23
2.4	(a) An instance of a Sudoku puzzle. (b) Potential values in each blank cell are inserted based on the given clues of the Sudoku instance in Figure 2.4(a); here green digits are naked singles. (c) Consequent naked singles (green digits) found from Figure 2.4(b).	24-25
2.5	A Sudoku puzzle with unique missing candidate in cell [5,3].....	25
2.6	An instance of Sudoku puzzle, where hidden singles can be found in cells [7,2] and [4,9].	26
2.7	An example row of a Sudoku puzzle with a lone ranger 3 in the second cell.....	27
2.8	A Sudoku puzzle with probable locked candidates in the last row of	

	minigrd 6 (and here the locked candidates are 3 and 5 in cells [6,7] and [6,8]), in the first column of minigrd 8 (and here the locked candidates are 9 and 3 in cells [8,4] and [9,4]), and so on..	27
2.9	(a) A partial Sudoku instance with the presence of a twin comprising 2 and 3 in cells [2,5] and [2,6]. (b) Elimination of probable values based on the twin from the second row and from the second minigrd... ..	28
2.10	Example rows of Sudoku puzzles with different varieties of triplet. (a) A triplet of <i>Variety# 1</i> . (b) A triplet of <i>Variety# 2</i> . (c) A triplet of <i>Variety# 3</i>	29
2.11	An example row of a Sudoku puzzle with quad comprising digits 1, 2, 4, and 7 present in columns four, six, seven, and eight... ..	30
2.12	A Sudoku puzzle with X-wing comprising digit 2 present at the crossings of rows three and seven, and columns two and eight..	31
2.13	A Sudoku puzzle with XY-wing, where $x = 5$, $y = 1$, and $z = 8$	32
2.14	A Sudoku puzzle structure with XY-wing, sharing minigrd.	32
2.15	A Sudoku puzzle with XYZ-wing occurred at the fourth and eighth column.... ..	33
2.16	A Sudoku puzzle with swordfish occurred at the first, fifth, and seventh column.	33
2.17	A Sudoku puzzle with multi-colouring for cells [2, 4], [2,6], [3,2], [3,4], and [5,6].	34
2.18	A Sudoku puzzle where we can apply the forcing chain method for confirming the value of cell [8,4].... ..	35
2.19	An example of solved Sudoku puzzle used for <i>Tuleja's theorem</i>	36
2.20	Assignment of numbers in cells in rows belonging to other minigrds of a Sudoku puzzle following <i>Tuleja's theorem</i>	37
2.21	(a) A Sudoku puzzle instance with pair. (b) A Sudoku puzzle instance without pair, used to illustrate <i>Tuleja's theorem</i>	38
2.22	An instance of a Sudoku Puzzle having 47 solution spaces (or empty cells).	40
2.23	Flowchart of an algorithm based on Bee colony.	55

2.24	Flowchart of an algorithm based on Artificial Immune System (AIS). ...	56
2.25	A partial Sudoku puzzle that consists of triples as a part to study the first case.	58
2.26	A partial Sudoku puzzle that contains pairs as a part to study the second case.	58
2.27	(a) A partial Sudoku puzzle consisting of pairs comprising digits 6 and 9. (b) The same partial Sudoku puzzle after mutual exchange of 6 and 9 in each pair..	59
3.1	(a) An instance of the Sudoku problem. (b) A solution of the Sudoku instance shown in Figure 3.1(a)..	61
3.2	The structure of a 9×9 Sudoku puzzle (problem) with its nine minigrids of size 3×3 each as numbered (in grey outsized font) 1 through 9.....	63
3.3	The permutation tree for three natural numbers 1, 2, and 3 following the algorithm of Arnow [42].	72
3.4	The permutation tree for three distinct colours following the algorithm of Latif [43].	72
3.5	The permutation tree for generating only valid permutations of the missing digits in minigrid 3 of the Sudoku instance shown in Figure 3.1(a).	75
3.6	(a) A <i>zigzag</i> way of considering the minigrids starting with minigrid 1, following a row-major sequence, and ending with minigrid 9. (b) A <i>spiral</i> way of considering the minigrids starting with minigrid 1, following a row-major sequence, and ending with minigrid 5. (c) A <i>semi-spiral</i> way of considering the minigrids starting with minigrid 1, following a partial row-major sequence, and ending with minigrid 3.... ..	77
3.7	(a) A <i>zigzag</i> way of considering the minigrids starting with minigrid 1, following a column-major sequence, and ending with minigrid 9. (b) A <i>spiral</i> way of considering the minigrids starting with minigrid 1, following a column-major sequence, and ending with minigrid 5. (c) A <i>semi-spiral</i> way of considering the minigrids starting with minigrid 1, following a partial column-major sequence, and ending with minigrid 7.....	79
4.1	A graph G_S that contains nine vertices, and for the sake of simplicity we draw it in a 3×3 fashion, where the three vertices in a row form a	

	triangle and the three vertices in a column form a triangle..	92
4.2	A 35-clue Sudoku instance P that we consider for developing the graph theoretic technique.	93
4.3	The graph G that is obtained for the 35-clue Sudoku instance P of Figure 4.2..	95
4.4	The linked list representation of graph G in Figure 4.3.... ..	96
4.5	The modified graph G obtained by repetitively deleting vertices (along with their edges) with degree three or less starting from the graph, G in Figure 4.3.... ..	97
4.6	The finally modified graph G , obtained by repetitively deleting vertices (along with their edges) based on the compatibility among the corresponding permutations along rows and columns starting from the intermediate modified graph, G in Figure 4.5..	98
4.7	The solution of the 35-clue Sudoku instance P shown in Figure 4.2 that we considered for developing the graph theoretic version of the algorithm <i>GTSPS</i>	100
4.8	The graph G^* , derived from the finally modified graph G (in Figure 4.5) by combining row-wise and column-wise matching permutations, where the matched permutations in a row (or in a column) form a super-vertex of size three each, and the two such super-vertices are connected by an edge only if the related permutations also match to each other for all the three orthogonal minigrids..	102
4.9	(a) A Sudoku puzzle P. (b) Two of the valid solutions of P..... ..	109
4.10	Comparative bar chart based on average CPU times (in milliseconds)..... ..	113
4.11	Comparative bar chart based on average numbers of iterations..	114
5.1	A solved Sudoku puzzle... ..	126
5.2	An Extremely Easy Sudoku instance created after randomly digging holes in the solved Sudoku puzzle shown in Figure 5.1.	127
5.3	Cells are dug in rowwise direction while (a) wandering along ‘S’ path and (b) moving from the left to right direction.	128

5.4	A Sudoku puzzle instance is created by symmetric removal of values (once from top-left and then from bottom-right) from the solved puzzle shown in Figure 5.1... ..	129
5.5	Flowchart for the <i>Digging Hole</i> strategy.	131
5.6	An instance of Sudoku puzzle.	133
5.7	A new instance is generated from the Sudoku instance shown in Figure 5.6 after interchanging of 1 and 9.	133
5.8	A new instance is generated from the Sudoku instance shown in Figure 5.6 after exchanging three pairs of values.	134
5.9	A new Sudoku instance is generated after replacement of all the digits for the Sudoku instance shown in Figure 5.6.	134
5.10	A new Sudoku instance generated after left rotation of 90 degree of the Sudoku instance shown in Figure 5.6.	135
5.11	A new Sudoku instance is generated after right rotation of 90 degree of the Sudoku instance of Figure 5.6.... ..	136
5.12	A new Sudoku instance is generated after rotation of 180 degree of Sudoku instance shown in Figure 5.6.	136
5.13	A new Sudoku instance is generated after flipping vertically of the Sudoku instance shown in Figure 5.6.	137
5.14	A New Sudoku instance is generated after flipping horizontally of the Sudoku instance shown in Figure 5.6.	137
5.15	Concept of band in Sudoku puzzle.	138
5.16	A new Sudoku instance is created after exchanging the values present in rows 1 and 2 in band 1 for the Sudoku instance shown in Figure 5.6.. ...	139
5.17	Concept of stack in Sudoku puzzle.	139
5.18	A new Sudoku instance is created after exchanging the values of column 1 and column 2 of the Sudoku instance shown in Figure 5.6.. ...	140
5.19	A new Sudoku instance is created after exchanging the values in band 1 with that of band 3 for the Sudoku instance shown in Figure 5.6..	140
5.20	A new Sudoku instance is created after exchanging the values in stack 1	

	and stack 3 present in the Sudoku instance shown in Figure 5.6.	141
5.21	(a) A given Sudoku instance. (b) A newly generated Sudoku instance after application of several methods of transformation.	142
6.1	Sketch generation and template reconstruction of biometric information.	148
6.2	Categorization of biometric template protection scheme.	151
6.3	Authentication mechanism when the biometric template is protected using feature transformation approach (courtesy to [60]).	152
6.4	Authentication mechanism using biometric cryptosystem (courtesy to [60]).	153
6.5	Biometric template encryption process: The biometric template is divided and placed over a region of 9×9 blocks..	155
6.6	The process of key generation.	155
6.7	Merging of Server key with User key.	156
6.8	Sample E94 database.	157
6.9	Histogram of the template before embedding Sudoku.	158
6.10	Histogram of the template after embedding Sudoku.	158
6.11	(a) An instance of 9×9 Sudoku problem. (b) A solution of the Sudoku instance shown in Figure 6.11(a). (c) The tile matrix (T) for creating the reference matrix (M) by subtracting 1 from each of the values in each cell of the solved Sudoku grid obtained in Figure 6.11(b).	162
6.12	Pixel tuple from carrier image.	162
6.13	(a) An instance of 8×8 Sudoku puzzle. (b) A solution of the Sudoku instance.	164
6.14	Embedding of the 8×8 Sudoku solution into the cover image.	165
6.15	An 18×18 reference matrix (M) is created using tile matrix (T), introducing the solved Sudoku puzzle shown in Figure 6.11(c).	165
6.16	The proposed Steganographic scheme.	169
6.17	A sample cover image.	170

6.18	Stego image after embedding the sample cover image shown in Figure 6.17, in an 8×8 Sudoku and secret message..	170
6.19	Histogram of the cover image, shown in Figure 6.17.	171
6.20	Histogram of the stego image, shown in Figure 6.18.	171

List of Tables

Table Number	Table Name	Page Number
1.1	Number of clues given in a Sudoku puzzle in defining the level of difficulty of a Sudoku instance.	8
1.2	The lower bound on the number of clues given in each row and column of a Sudoku instance for each corresponding level of difficulty.	8
4.1	Comparison based on average number of iterations and CPU time for Sudoku instances with multiple solutions..	110
4.2	Comparison based on average number of iterations and CPU time for Sudoku instances of difficulty level ‘Easy’..	110
4.3	Comparison based on average number of iterations and CPU time for Sudoku instances with difficulty level ‘Medium’..	111
4.4	Comparison based on average number of iterations and CPU time for Sudoku instances with difficulty level ‘Hard’..	112
4.5	Comparison based on average number of iterations and CPU time for Sudoku instances with difficulty level ‘Evil’..	112
4.6	Comparison based on average number of nodes in the initial graph structure obtained using of the proposed <i>Graph Theoretic solver (GTSPS)</i>	113
4.7	The novelty of <i>GTSPS</i> , the guessed free Sudoku solver developed in this thesis in comparison to other existing Sudoku solvers for different measuring issues.	117-118
4.8	A table of comparison among the different versions of the devised Sudoku solver for a Sudoku instance P of size $n \times n$, p and x being the average number of blank cells and the average number of invalid permutations computed for a minigrad.	121
5.1	The amount ranges of givens in each difficulty level of Sudoku puzzle.	125
5.2	The lower bound on the number of clues in each row and column for each difficulty level.	125

List of Abbreviations

AIS	Artificial Immune System
BCO	Bee Colony Optimization
CGA	Cultural Genetic Algorithm
DNA	Deoxyribo Nucleuc Acid
GA	Genetic Algorithm
HGASA	Hybrid Genetic Algorithm and Simulated Annealing
LSB	Least Significant Bit
NP	Non-deterministic Polynomial
QSA	Quantum Simulated Annealing
RGB	Red Green Blue
RPSO	Repulsive Particle Swarm Optimization
SMS	Short Message Service
TSP	Travelling Salesman Problem