Sudoku as a Constraint Satisfaction Problem

Sean Murphree
Dept of Computer Science
University of California, Irvine
Irvine, CA 92617, USA

Abstract:
Purpose of this paper is to approach and solve Sudoku as a Constraint Satisfaction Problem and to compare various heuristics and their effects on solving the problem. Heuristics compared include Backtracking with Forward Checking and Minimum Remaining Value, Arc Consistency, and Arc Consistency Pre-Processing.

Introduction:

In this paper I approached the classic Sudoku problem as a constraint satisfaction problem. Sudoku problems consist of 9 rows and 9 columns, this 9x9 square space is further subdivided into 9 3x3 squares. Problems have some of the original 81 squares filled in with numbers in the range of 1 to 9. The remaining squares must then be filled in with numbers from the same range, such that no row, column, or 3x3 square contains a repetition of the same number; they must all be unique.

Constraint satisfaction parallels Sudoku problems well. We can create AllDif constraints, one for each row, column and 3x3 square. From that we have 27 different constraints, each on 9 variables, such that the variables cannot take on the value of any other variable in the constraint.

The goal if this paper is to contrast and compare different CSP search heuristics including Backtracking with Forward Checking, Minimum Remaining Value, Arc Consistency and Arc Consistency as Pre-Processing.

Methods:

The heuristics used in this paper are Backtracking Search with Forward Checking as outlined by the book [Russel & Norvig, pg 215, 216], as well as Minimum Remaining Value, Arc Consistency and Arc Consistency as Pre-Processing.

To collect timing data, time was taken before loading problems, before searching for a solution but after loading and pre-processing, and at the end of the search. To collect an average run time, each test case was executed 100 times and the total run time was divided by 100. This lead to a tremendous improvement in time, on the scale of 10 times faster. This is due to allocation costs of Java.

Nodes are considered as generated (and thus the counter incremented) after the test for being in a failed state which was described in the Backtracking algorithm in Russel & Norvig. This location was chosen do to the fact that if we are in a failed state, we won't recursively call backtracking but we did attempt the state which is considered a node. Also, if the node is not a failed state, we will recursively call the Backtracking algorithm which will eventually unwind back to the location following it's call, which is where the node counter increment takes place. Thus, this location allows for it to count both failed and non-failed variable assignment states as a node generation.

Results:

Benchmarks were performed on a 2.1Ghz dual core computer with 4GB of ram running Windows Vista 32bit. The results are included in the table below for each of the heuristics implemented, along with their total time, search time, number of nodes generated, and effective branching factor.

Heuristics Used	Total Time (msecs)	Search Time (msecs)	Nodes Generated	Effective Branching Factor
Test Case 1				
BT-FC	4	3	10	
BT-FC-MRV	3	2	6	
BT-ACP-MRV	2	1	3	
BT-AC-MRV	4	3	1	
Test Case 2				
BT-FC	141	139	674	
BT-FC-MRV	123	122	578	
BT-ACP-MRV	166	164	811	
BT-AC-MRV	247	245	417	
Test Case 3				
BT-FC	94	93	414	
BT-FC-MRV	62	61	270	
BT-ACP-MRV	12	10	36	
BT-AC-MRV	114	113	165	

Discussion:

Most of the results followed what was to be expected. However, in Test Case 2 using ACP with MRV, we can see that more nodes were generated than were generated using the different heuristics. This was unexpected given the idea that ACP will narrow down domains, and MRV would select good values for variables. But this just goes to show that there are some Sudoku problems where MRV is not the best heuristic. Test Case 2 may be improved by other heuristics such as a degree heuristic, to reduce the branching factor on future choices.

References:

1. Russel, Stuart, and Peter Norvig. *Artificial Intelligence A Modern Approach*. 3rd ed. New York: Prentice Hall, 2010. Print.

Supplementary Information:

The following is a list of variable domains after running ACP to support that ACP is implemented correctly. Variables are given by their index in a single dimension array in row major order.

Test Case 1:

Var[0] Domain Values: [3, 9]
Var[1] Domain Values: [6]
Var[2] Domain Values: [3, 9]
Var[3] Domain Values: [1]
Var[4] Domain Values: [2, 7, 8]
Var[5] Domain Values: [4]
Var[6] Domain Values: [2, 7, 8]
Var[7] Domain Values: [5]
Var[8] Domain Values: [7, 8, 9]
Var[9] Domain Values: [1, 4, 9]
Var[10] Domain Values: [1, 7, 9]
Var[11] Domain Values: [8]
Var[12] Domain Values: [3]
Var[13] Domain Values: [2, 7]
Var[14] Domain Values: [5]
Var[15] Domain Values: [6]
Var[16] Domain Values: [2, 4, 9]
Var[17] Domain Values: [7, 9]
Var[18] Domain Values: [2]
Var[19] Domain Values: [3, 5, 7]
Var[20] Domain Values: [3, 4, 5]
Var[21] Domain Values: [6]
Var[22] Domain Values: [7, 8]
Var[23] Domain Values: [9]
Var[24] Domain Values: [4, 7, 8]
Var[25] Domain Values: [3, 4, 8]
Var[26] Domain Values: [1]
Var[27] Domain Values: [8]
Var[28] Domain Values: [1, 2, 5, 9]
Var[29] Domain Values: [1, 5, 9]
Var[30] Domain Values: [4]
Var[31] Domain Values: [3]
Var[32] Domain Values: [7]
Var[33] Domain Values: [2, 5]
Var[34] Domain Values: [1, 2, 9]
Var[35] Domain Values: [6]
Var[36] Domain Values: [1, 4, 9]
Var[37] Domain Values: [1, 9]
Var[38] Domain Values: [6]
Var[39] Domain Values: [8]
Var[40] Domain Values: [5]
Var[41] Domain Values: [2]
Var[42] Domain Values: [3]
Var[43] Domain Values: [1, 9]
Var[44] Domain Values: [7, 9]
Var[45] Domain Values: [7]
Var[46] Domain Values: [2, 3, 5]
Var[47] Domain Values: [3, 5]
Var[48] Domain Values: [9]
Var[49] Domain Values: [6]
Var[50] Domain Values: [1]
Var[51] Domain Values: [2, 5, 8]
Var[52] Domain Values: [2, 8]
Var[53] Domain Values: [4]
Var[54] Domain Values: [5]
Var[55] Domain Values: [1, 8, 9]
Var[56] Domain Values: [1, 9]

```
Var[57] Domain Values: [7]
Var[58] Domain Values: [1, 4]
Var[59] Domain Values: [3]
Var[60] Domain Values: [4, 8]
Var[61] Domain Values: [4, 6, 8]
Var[62] Domain Values: [2]
Var[63] Domain Values: [1, 3]
Var[64] Domain Values: [1, 3, 8]
Var[65] Domain Values: [7]
Var[66] Domain Values: [2]
Var[67] Domain Values: [1, 4]
Var[68] Domain Values: [6]
Var[69] Domain Values: [9]
Var[70] Domain Values: [4, 8]
Var[71] Domain Values: [5, 8]
Var[72] Domain Values: [6]
Var[73] Domain Values: [4]
Var[74] Domain Values: [2]
Var[75] Domain Values: [5]
Var[76] Domain Values: [9]
Var[77] Domain Values: [8]
Var[78] Domain Values: [1]
Var[79] Domain Values: [7]
Var[80] Domain Values: [3]
```

Test Case 2:

```
Var[0] Domain Values: [1, 3, 6, 7]
Var[1] Domain Values: [3, 6, 7, 8]
Var[2] Domain Values: [1, 3, 6, 7, 8]
Var[3] Domain Values: [1, 7]
Var[4] Domain Values: [1, 5, 8]
Var[5] Domain Values: [4]
Var[6] Domain Values: [9]
Var[7] Domain Values: [2, 3, 5, 7, 8]
Var[8] Domain Values: [1, 2, 3, 5, 7, 8]
Var[9] Domain Values: [1, 4, 6, 7, 9]
Var[10] Domain Values: [4, 6, 7, 8, 9]
Var[11] Domain Values: [5]
Var[12] Domain Values: [3]
Var[13] Domain Values: [2]
Var[14] Domain Values: [1, 7, 8]
Var[15] Domain Values: [6, 7, 8]
Var[16] Domain Values: [7, 8]
Var[17] Domain Values: [1, 7, 8]
Var[18] Domain Values: [2]
Var[19] Domain Values: [3, 7, 8, 9]
Var[20] Domain Values: [1, 3, 7, 8]
Var[21] Domain Values: [1, 7, 9]
Var[22] Domain Values: [1, 5, 8, 9]
Var[23] Domain Values: [6]
Var[24] Domain Values: [5, 7, 8]
Var[25] Domain Values: [4]
Var[26] Domain Values: [1, 3, 5, 7, 8]
Var[27] Domain Values: [8]
Var[28] Domain Values: [3, 7, 9]
Var[29] Domain Values: [4]
```

```
Var[30] Domain Values: [1, 2, 7, 9]
Var[31] Domain Values: [1, 3, 5, 9]
Var[32] Domain Values: [1, 2, 3, 5, 7]
Var[33] Domain Values: [2, 5, 7]
Var[34] Domain Values: [6]
Var[35] Domain Values: [2, 5, 7]
Var[36] Domain Values: [3, 7, 9]
Var[37] Domain Values: [5]
Var[38] Domain Values: [2, 3, 7]
Var[39] Domain Values: [2, 4, 7, 9]
Var[40] Domain Values: [6]
Var[41] Domain Values: [2, 3, 7, 8]
Var[42] Domain Values: [2, 4, 7, 8]
Var[43] Domain Values: [1]
Var[44] Domain Values: [2, 4, 7, 8]
Var[45] Domain Values: [6, 7]
Var[46] Domain Values: [1]
Var[47] Domain Values: [2, 6, 7]
Var[48] Domain Values: [2, 4, 7]
Var[49] Domain Values: [4, 5, 8]
Var[50] Domain Values: [2, 5, 7, 8]
Var[51] Domain Values: [3]
Var[52] Domain Values: [2, 5, 7, 8]
Var[53] Domain Values: [9]
Var[54] Domain Values: [1, 3, 4, 5, 7]
Var[55] Domain Values: [2]
Var[56] Domain Values: [1, 3, 7]
Var[57] Domain Values: [8]
Var[58] Domain Values: [1, 3, 4]
Var[59] Domain Values: [1, 3]
Var[60] Domain Values: [4, 5, 7]
Var[61] Domain Values: [3, 5, 7, 9]
Var[62] Domain Values: [6]
Var[63] Domain Values: [3, 4, 5, 6]
Var[64] Domain Values: [3, 4, 6, 8]
Var[65] Domain Values: [3, 6, 8]
Var[66] Domain Values: [2, 4, 6]
Var[67] Domain Values: [7]
Var[68] Domain Values: [9]
Var[69] Domain Values: [1]
Var[70] Domain Values: [2, 3, 5, 8]
Var[71] Domain Values: [2, 3, 4, 5, 8]
Var[72] Domain Values: [1, 3, 4, 6, 7]
Var[73] Domain Values: [3, 4, 6, 7, 8]
Var[74] Domain Values: [9]
Var[75] Domain Values: [5]
Var[76] Domain Values: [1, 3, 4]
Var[77] Domain Values: [1, 2, 3]
Var[78] Domain Values: [2, 4, 7, 8]
Var[79] Domain Values: [2, 3, 7, 8]
Var[80] Domain Values: [2, 3, 4, 7, 8]
```

Test Case 3:

```
Var[0] Domain Values: [1, 3, 4, 5]
Var[1] Domain Values: [4, 5, 6]
Var[2] Domain Values: [9]
```

```
Var[3] Domain Values: [1, 3, 4, 6]
Var[4] Domain Values: [2]
Var[5] Domain Values: [8]
Var[6] Domain Values: [3, 4, 6, 7]
Var[7] Domain Values: [1, 4, 5, 7]
Var[8] Domain Values: [1, 3, 4, 6, 7]
Var[9] Domain Values: [1, 2, 3, 4, 5]
Var[10] Domain Values: [8]
Var[11] Domain Values: [1, 2, 3, 5]
Var[12] Domain Values: [1, 3, 4, 6]
Var[13] Domain Values: [1, 6, 7]
Var[14] Domain Values: [1, 3, 4, 6, 7]
Var[15] Domain Values: [9]
Var[16] Domain Values: [1, 2, 4, 5, 7]
Var[17] Domain Values: [1, 2, 3, 4, 6, 7]
Var[18] Domain Values: [1, 2, 3, 4]
Var[19] Domain Values: [7]
Var[20] Domain Values: [1, 2, 3]
Var[21] Domain Values: [1, 3, 4, 6]
Var[22] Domain Values: [5]
Var[23] Domain Values: [1, 3, 4, 6, 9]
Var[24] Domain Values: [3, 4, 6, 8]
Var[25] Domain Values: [1, 2, 4, 8]
Var[26] Domain Values: [1, 2, 3, 4, 6, 8]
Var[27] Domain Values: [2, 7]
Var[28] Domain Values: [3]
Var[29] Domain Values: [8]
Var[30] Domain Values: [9]
Var[31] Domain Values: [6, 7]
Var[32] Domain Values: [2, 4, 6, 7]
Var[33] Domain Values: [1]
Var[34] Domain Values: [4, 7]
Var[35] Domain Values: [5]
Var[36] Domain Values: [2, 5, 7, 9]
Var[37] Domain Values: [2, 5, 9]
Var[38] Domain Values: [2, 5, 7]
Var[39] Domain Values: [1, 2, 3, 4, 6, 8]
Var[40] Domain Values: [1, 6, 7, 8]
Var[41] Domain Values: [1, 2, 3, 4, 6, 7]
Var[42] Domain Values: [3, 4, 6, 7, 8]
Var[43] Domain Values: [4, 7, 8]
Var[44] Domain Values: [3, 4, 6, 7, 8]
Var[45] Domain Values: [6]
Var[46] Domain Values: [1]
Var[47] Domain Values: [4]
Var[48] Domain Values: [3, 8]
Var[49] Domain Values: [7, 8]
Var[50] Domain Values: [5]
Var[51] Domain Values: [2]
Var[52] Domain Values: [9]
Var[53] Domain Values: [3, 7, 8]
Var[54] Domain Values: [1, 2, 3, 5, 7, 8, 9]
Var[55] Domain Values: [2, 5, 9]
Var[56] Domain Values: [1, 2, 3, 5, 7]
Var[57] Domain Values: [1, 2, 5, 8]
Var[58] Domain Values: [4]
Var[59] Domain Values: [1, 2, 9]
```

```
Var[60] Domain Values: [7, 8]
Var[61] Domain Values: [6]
Var[62] Domain Values: [1, 2, 7, 8, 9]
Var[63] Domain Values: [1, 2, 4, 5, 7, 8, 9]
Var[64] Domain Values: [2, 4, 5, 9]
Var[65] Domain Values: [6]
Var[66] Domain Values: [1, 2, 5, 8]
Var[67] Domain Values: [1, 8, 9]
Var[68] Domain Values: [1, 2, 9]
Var[69] Domain Values: [4, 7, 8]
Var[70] Domain Values: [3]
Var[71] Domain Values: [1, 2, 4, 7, 8, 9]
Var[72] Domain Values: [1, 2, 4, 8, 9]
Var[73] Domain Values: [2, 4, 9]
Var[74] Domain Values: [1, 2]
Var[75] Domain Values: [7]
Var[76] Domain Values: [3]
Var[77] Domain Values: [1, 2, 6, 9]
Var[78] Domain Values: [5]
Var[79] Domain Values: [1, 2, 4, 8]
Var[80] Domain Values: [1, 2, 4, 8, 9]
```