

Constraint Programming and Variant Sudoku

Nick Leaton

February 2022

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

Contents

1	Introduction	4
2	Sets	4
3	Variables	4
4	Constraints	5
4.1	Basic Constraints	5
4.2	Uniqueness	5
4.3	Rows	5
4.4	Columns	5
4.5	Boxes	6
4.6	Diagonals	6
4.7	Known Values	7
5	Sudoku Variants	7
5.1	Odd Even	7
6	Lines	7
6.1	German and Dutch Whispers	8
6.2	Arrows	8
6.3	Thermometer	9
6.4	FrozenThermometer	9
6.5	Palindromes	9
6.6	Renban	9
6.7	Between the Ends	10
6.8	Diagonal	10
6.9	Argyle	10
7	Magic Squares	10
8	Pairs	11
8.1	XV Sudoku	11
8.2	Greater Than Sudoku	12
8.3	Consecutive	12
8.4	Kropki	12
9	Chess Move	13
9.1	Fortress Sudoku	13
9.2	Anti-king	14
9.3	Anti-knight	14
9.4	Donkey Step	14
9.5	Monkey Step	15
9.6	Orthogonally Non Consecutive	15

10 Regional Sum	15
10.1 Killer	15
10.2 Little Killer	16
11 Formulations	17
11.1 Abs	17
11.2 Not equal	17
11.3 In Set	17
11.4 Not In Set	17

1 Introduction

For a standard Sudoku, we have 9 rows and 9 columns. We have 9 possible digits in each square. So lets define them as sets. For other sized grids, the sets will change. For the moment lets go with a 9x9 board.

2 Sets

$$Digits = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \quad (1)$$

$$Rows = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \quad (2)$$

$$Columns = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \quad (3)$$

We will need some variables. For each cell we will have 9 variables, each a boolean. If the digit 8 is in the cell the variable corresponding to 8 will be true, and all the others will be false. So we will have a set of boolean values where 0 is false and 1 is true

We can add in a set of all possible cells

$$\begin{aligned} Board &= \{(r, c)\} \\ \forall r &\in Rows \\ \forall c &\in Columns \end{aligned} \quad (4)$$

We need a set of booleans where we can use them for choices.

$$Boolean = \{0, 1\} \quad (5)$$

3 Variables

We will then have the following variables where the C conveys the concept of the choice in that cell.

$$\begin{aligned} &C_{rcv} \\ \forall r &\in Rows \\ \forall c &\in Columns \\ \forall v &\in Digits \\ &where C_{rcv} \in Boolean \end{aligned} \quad (6)$$

To make writing some constraints simpler add in some new variables for the values in a given cell. Since this is a derived value it can be any number. In practice since it will be the sum of integers, it will be an integer.

$$\begin{aligned}
& V_{rc} \\
& \forall r \in Rows \\
& \forall c \in Columns \\
& V_{rc} \in Digits
\end{aligned} \tag{7}$$

4 Constraints

4.1 Basic Constraints

To restrict the choices to just one in each cell, we need the following constraint.

$$\sum_{v \in Values} C_{rcv} = 1, \forall r \in Rows, \forall c \in Columns \tag{8}$$

For the constraint on the value variables, we add the following.

$$\sum_{v \in Values} v.C_{rcv} = V_{rc}, \forall r \in Rows, \forall c \in Columns \tag{9}$$

4.2 Uniqueness

Typically there are regions that impose a uniqueness condition. These work in a very similar way. If we have a set of the cells in the region

$$Region = \{(r, c)\}, \forall (r, c) \in Region \tag{10}$$

The we can impose the condition for uniqueness as follows. Not all regions will have a full set of digits, so the ≤ 1 condition needs to be applied. If the region has 9 cells, then all the sums will be 1.

$$\sum_{(r,c) \in Region} C_{rcv} \leq 1, \forall v \in Digits \tag{11}$$

4.3 Rows

Rows now form 9 regions.

$$Row(i) = \{(i, c)s\}, \forall c \in Column \tag{12}$$

4.4 Columns

Columns do the same

$$Column(i) = \{(r, i)\}, \forall r \in Rows \tag{13}$$

4.5 Boxes

In addition to the uniqueness restrictions on rows and columns, some puzzles have boxes that are 3x3. In a 9x9 sudoku there are 9 of them and we can number them Boxes are numbered left to right, top to bottom.

$$Boxes = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \quad (14)$$

So first we need to go from the box number to the centre cell of each box.

$$Centre(i) = ((i - 1) \text{ div } 3 + 2, (i - 1) \text{ mod } 3 + 2), \forall (r, c) \in Region \quad (15)$$

We can now generate the Region of all centre cells of boxes

$$Centers(i) = \{Centre(i)\}, \forall i \in Regions \quad (16)$$

We can also generate the Regions for each box. First we create a set of offsets

$$Offsets = \{-1, 0, 1\} \quad (17)$$

Then

$$Boxes(i) = \{Centre(i) + (x, y), x \in Offsets, y \in Offsets\}, \forall i \in Regions \quad (18)$$

4.6 Diagonals

We need two regions. One for the top left to bottom right diagonal, and one for the bottom left to top right.

$$TLBR = \{(x, x), \forall x \in Rows\} \quad (19)$$

$$\sum C_{rcv} = 1, \forall (r, c) \in TLBR \quad (20)$$

$$BLTR = \{(x, 10 - x), \forall x \in Rows\} \quad (21)$$

$$\sum C_{rcv} = 1, \forall (r, c) \in BLTR \quad (22)$$

This sets up the abstract model for standard Sudokus.

4.7 Known Values

What we need to do now is make the model concrete by add in known digits.

They are easy to constrain. If the required digit in a cell (r,c) is D, we can impose this condition.

$$V_{rc} = D \quad (23)$$

So far that enables the solution of standard Sudokus.

5 Sudoku Variants

There are lots of Sudoku variants. Here are some of the variants and how to add constraints that enable them to be solved.

5.1 Odd Even

Some puzzles restrict the choices that can be placed in cells to subsets of the digits. For eaxample that they must be even or odd. So we can define those sets.

$$Odds = \{1, 3, 5, 7, 9\} \quad (24)$$

$$Evens = \{2, 4, 6, 8\} \quad (25)$$

Then for a given cell (r,c) that has to be even or odd we can add one of the following constraints. Not that it is \notin as the set condition. This is to force the choice of that cell to have an odd value in the case of an even cell is not allowed, and vice versa for even digits in odd cells.

$$C_{rcv} = 0, \forall v \notin Evens \quad (26)$$

$$C_{rcv} = 0, \forall v \notin Odds \quad (27)$$

6 Lines

There are sudoku puzzles based on lines where the elements on the lines are subject to restrictions. Lines are represented as an ordered list

$$line = [(r, c)] \forall (r, c) \in Line \quad (28)$$

$$n = length(line) \quad (29)$$

6.1 German and Dutch Whispers

A German Whisper line is a line in the puzzle. Adjacent digits on the line must have a difference of at least 5. That precludes a 5 appearing on the line since the adjacent number would have to be a 0 or a 10, and neither are valid numbers. We can then add a constraint that disallows any 5 on the line. This condition is not strictly necessary, but it does cut down choices.

$$C_{r,c,5} = 0, \forall (r, c) \in Line \quad (30)$$

For the interior cells on a German Whisper line, then 4 and 6 should be excluded if the cells are in a region that imposes uniqueness. That is more difficult to implement since it involves working out if the three cells are in the same region or regions. So it is probably not worth implementing

A Dutch Whisper line is the same as a German Whisper line, except the difference has to be at least 4.

So if there are n cells on the line and the required difference is D , we can add these constraints

$$abs(V_{line[i]} - V_{line[i+1]}) \geq D, \forall i \in \{1 \dots n - 1\} \quad (31)$$

If the line forms a loop, then add an extra condition

$$abs(V_{line[1]} - V_{line[n]}) \geq D \quad (32)$$

6.2 Arrows

Arrow restrictions have a head of the line which is the sum of the other cells on the line. When the length of the line is n , for a simple one cell total we have.

$$V_{line[1]}^m = \sum_{i=2}^n V_{line[i]} \quad (33)$$

For an arrow with multiple cells that make the head, we have the list of cells, and a count of the number of cells in the head. If the number of cells in the head is m , and we have n cells in total, including the head on the arrow, we have this restriction. We can create the totals by adding up the hundreds digit, tens digit and unit digit.

$$\sum_{i=1}^m 10^{m-i} V_{line[i]} = \sum_{i=m+1}^n V_{line[i]} \quad (34)$$

The simple variant can be implemented by using the the more generic and setting $m = 1$.

6.3 Thermometer

Thermometer come in two flavours. The first is a strict thermometer where the values must increase along the line.

$$V_{line[i]} < V_{line[i+1]}, \forall i \in \{1 \dots n-1\} \quad (35)$$

6.4 FrozenThermometer

The second is such that they do not need to increase, but must obey any uniqueness condition for the cells region. For the non-strict case we have

$$V_{line[i]} \leq V_{line[i+1]}, \forall i \in \{1 \dots n-1\} \quad (36)$$

6.5 Palindromes

Palindromes are lines where the cell on the line matches the corresponding cell from the other end so digits read the same no matter which way you read it.

$$V_{line[i]} = V_{line[n-i]}, \forall i \in \{1 \dots \lfloor n/2 \rfloor\} \quad (37)$$

This works for circular palindrome lines.

Query. Do all circular palindromes have to have an even number of cells?

6.6 Renban

A renban puzzle consists of a line. Numbers on the line must form a sequence of integers with no gaps. The numbers can appear in any order. The constraints are to find the lowest number on the line, the highest or upper number on the line. The difference between the upper and lower bounds must be the line length less one.

$$lower \leq V_{line[i]}, \forall i \quad (38)$$

$$upper \leq V_{line[i]}, \forall i \quad (39)$$

Upper is always greater than lower, or equal if there is just one cell on the line which doesn't happen. We can then take differences and we will always have a positive difference.

$$upper - lower = n - 1 \quad (40)$$

Lastly, the values on the line must be unique

$$\sum_{d \in Digits} C_{d, line[i]} = 1, \forall i \quad (41)$$

6.7 Between the Ends

A between the ends line has two ends with numbers in them. The digits on the line must be between the two numbers. There are no uniqueness constraints other than the constraints imposed by other rules. The problem that needs to be solved is to get a lower value of the circles and the upper value of the circles. You don't know a priori which is the higher. Then the constraint for the numbers is easy. It's also clear to work out that ones and nines cannot be on the line.

$$V_{line[i]} < U, \forall i \notin \{1, n\} \quad (42)$$

$$V_{line[i]} > L, \forall i \notin \{1, n\} \quad (43)$$

6.8 Diagonal

In a diagonal sudoku, either one or two diagonal lines are added. Along each diagonal, digits must be unique.

6.9 Argyle

7 Magic Squares

Magic Squares are a 3 by 3 set of squares where the sum of any row, column or diagonal are the same. The total must be 15. Another fact is that the centre square must be 5.

The cells are labelled in the same way as boxes are numbered. A function returns the ordered list of the cells in the magic square.

$$magic(r, c) = [(r + x, r + y), \forall y \in Offsets, \forall x \in Offsets] \quad (44)$$

A magic square is specified by its centre, (r,c). The centre of a magic square must be 5.

$$V_{magic[5]} = 5 \quad (45)$$

It's not necessary, but corners of magic squares must be even, and edges must be odd.

$$C_{magic[i]v} = 0, \forall v \in Evens, \forall i \in \{1, 3, 7, 9\} \quad (46)$$

and

$$C_{magic[i]v} = 0, \forall v \in Odds, \forall i \in \{2, 4, 6, 8\} \quad (47)$$

The constraint sums are now implemented.

$$\sum_{i \in \{1,2,3\}} V_{magic[i]} = 15 \quad (48)$$

$$\sum_{i \in \{4,5,6\}} V_{magic[i]} = 15 \quad (49)$$

$$\sum_{i \in \{7,8,9\}} V_{magic[i]} = 15 \quad (50)$$

$$\sum_{i \in \{1,4,7\}} V_{magic[i]} = 15 \quad (51)$$

$$\sum_{i \in \{2,5,8\}} V_{magic[i]} = 15 \quad (52)$$

$$\sum_{i \in \{3,6,9\}} V_{magic[i]} = 15 \quad (53)$$

$$\sum_{i \in \{1,5,9\}} V_{magic[i]} = 15 \quad (54)$$

$$\sum_{i \in \{3,5,7\}} V_{magic[i]} = 15 \quad (55)$$

8 Pairs

Some of the variants look at the restriction between two different cells.

For some of these we need a complete set of pairs on the board.

$$starts = Rows - \{9\} \quad (56)$$

$$RowPairs = \{Pair((r, c), (r + 1, c), \forall r \in starts, \forall c \in starts\} \quad (57)$$

$$ColumnPairs = \{Pair((r, c), (r, c + 1), \forall r \in starts, \forall c \in starts\} \quad (58)$$

$$Pairs = RowPairs \cup ColumnPairs \quad (59)$$

8.1 XV Sudoku

TODO Image of XV Sudoku

XV Sudoku consists of a pair of orthogonally adjacent cells. If a Roman V is between the cells they add up to 5. If an X is between the cells they total 10. There are only two ways to make a 5, 1 and 4, 2 and 3. So we can add a new set to give the solver a helping hand. For the two cells they can be defined as a line with two elements

$$vset = \{1, 2, 3, 4\} \quad (60)$$

$$C_{rcv} = 0, \forall v \notin vset, \forall (r, c) \in Pair \quad (61)$$

Then we need the total for the V variant

$$\sum_{(r,c) \in Pair} V_{rc} = 5 \quad (62)$$

or for the X variant

$$\sum_{(r,c) \in Pair} V_{rc} = 10 \quad (63)$$

In some cases XV sudoku puzzles include the rule that all XV cells are given. In which case you are not allowed any non marked cells that sum to 5 or 10. This involves a \neq test, and that needs an addition boolean variable for each pair.

TODO implement !=

8.2 Greater Than Sudoku

Some puzzles contain greater than signs between two cells. The first of the pair must be less than the second of the pair.

$$V_{pair[1]} < V_{pair[2]} \quad (64)$$

This is a subset of Thermo Sudoku

8.3 Consecutive

Typical represented by a white dot on the border between two cells. The requirement is that the two cells are consecutive. So if one of the cells is a 4 the other cell can only be a 3 or 5.

The constraint is easy.

$$abs(V_{pair[1]} - V_{pair[2]}) = 1 \quad (65)$$

This is a subset of Renban Sudoku

8.4 Kropki

Here it is typical that a white dot is placed on the border. The requirement is that the one of the values of the pair is twice the value of the other. That means that 5, 7 and 9 can never appear as digits on a Kropki pair. There are only 8 possible combinations.

We can eliminate 5, 7 and 9.

Pair[1]	Pair[2]
1	2
2	1
2	4
3	6
4	2
4	8
6	3
8	4

$$Kropki = \{1, 2, 3, 4, 6, 8\} \quad (66)$$

$$C_{rcv} = 0, \forall v \notin Kropki, \forall (r, c) \in Pair \quad (67)$$

For the exact answer, introduce 8 binary variables corresponding to each of the possible combinations k_b

$$kropkis = \{12, 21, 24, 36, 42, 48, 63, 84\} \quad (68)$$

Constrain the set to be unique, or use a SOS variable

$$\sum_{b \in kropkis} k_b = 1 \quad (69)$$

$$b.k_b = 10.V_{r_1c_1} + V_{r_2c_2} \quad (70)$$

9 Chess Move

There are a selection of restrictions based of chess moves. The typical implementation is to have a set of possible offsets from a cell. The target cells are then generated from they (r,c) of the cell, and check to see if they are on the board.

9.1 Fortress Sudoku

In a fortress sudoku the cell that is the fortress is larger than the up to 4 orthogonally connected cells. If the fortress is (r,c) then

$$orthogonaloffsets = \{(-1, 0), (0, -1), (0, 1), (1, 0), (1, 1)\} \quad (71)$$

$$\begin{aligned} V_{rc} &> V_{r+x, c+y} \\ \forall (x, y) &\in orthogonaloffsets, \\ &if(r+x, c+y) \in Board \end{aligned} \quad (72)$$

This a variant of Thermo Sudoku

9.2 Anti-king

In an anti-king puzzle, no number may repeat a king's move away from another identical number.

TODO Diagram

The orthogonal move will not be possible because of the row and column restrictions, but if included a good preprocessor will remove them.

$$\begin{aligned} kingoffsets = \{ & (-1, -1), (-1, 0), (-1, 1), (0, -1), \\ & (0, 1), (1, -1), (1, 0), (1, 1) \} \end{aligned} \quad (73)$$

For the restrictions we use the choice variables

$$\begin{aligned} \sum_{v \in Values} C_{rcv} &= 0, \\ \forall (x, y) \in kingoffsets, \\ \forall (r, c) \in Board \\ if(r + x, c + y) &\in Board \end{aligned} \quad (74)$$

9.3 Anti-knight

For an anti knight move restriction is is the same as the anti-king except the offsets are different.

To do Diagram

$$\begin{aligned} knightoffsets = \{ & (-2, 1), (-1, 2), (1, 2), (2, 1), \\ & (-2, -1), (-1, -2), (1, -2), (2, -1) \} \end{aligned} \quad (75)$$

For the restrictions we use the choice variables

$$\begin{aligned} \sum_{v \in Values} C_{rcv} &= 0, \\ \forall (x, y) \in knightoffsets, \\ \forall (r, c) \in Board \\ if(r + x, c + y) &\in Board \end{aligned} \quad (76)$$

9.4 Donkey Step

The same digit cannot appear in a cell within 2 diagonal steps from itself.

$$donkeyoffsets = \{(-2, -2), (2, -2), (2, 2), (-2, 2), (1, 1)\} \quad (77)$$

$$\begin{aligned}
&abs(V_{rc} - V_{r+x, c+y}) \geq 2, \\
&\forall(x, y) \in donkeyoffsetsets, \\
&\quad \forall(r, c) \in Board \\
&if(r + x, c + y) \in Board
\end{aligned} \tag{78}$$

9.5 Monkey Step

The same digit cannot appear in a cell within 2 diagonal steps from itself.

$$monkeyoffsetsets = \{(-3, 1), (-1, 3), (1, 3), (3, 1), (-3, -1), (-1, -3), (1, -3), (3, -1)\} \tag{79}$$

$$\begin{aligned}
&abs(V_{rc} - V_{r+x, c+y}) \geq 2, \\
&\forall(x, y) \in monkeyoffsetsets, \\
&\quad \forall(r, c) \in Board \\
&if(r + x, c + y) \in Board
\end{aligned} \tag{80}$$

9.6 Orthogonally Non Consecutive

For orthogonally non consecutive rules, we need to make sure that if a 3 is in cell that any orthogonally adjacent cells are not 2 or 4. That means the absolute difference between the two cells must be greater than or equal to two. We start off with the offsets, then implement the rules.

TODO Diagram

$$orthogonaloffsetsets = \{(-1, 0), (0, -1), (0, 1), (1, 0), (1, 1)\} \tag{81}$$

$$\begin{aligned}
&abs(V_{rc} - V_{r+x, c+y}) \geq 2, \\
&\forall(x, y) \in orthogonaloffsetsets, \\
&\quad \forall(r, c) \in Board \\
&if(r + x, c + y) \in Board
\end{aligned} \tag{82}$$

An alternative implementation is to consider that there are effectively four Renban lines of length two.

10 Regional Sum

10.1 Killer

In a killer sudoku, a region has a total denoted by S. The sum of the cells values in that region must be the same as the total. Typically numbers cannot repeat inside that region. Note that a number does not have to appear in the region

TODO Diagram

$$region = \{(r, c) | (r, c) \in Region\} \quad (83)$$

Then the total is easy.

$$\sum_{(r,c) \in region} V_{rc} = S \quad (84)$$

Uniqueness again is simple.

$$\sum_{(r,c) \in region} C_{rcv} \leq 1, \forall v \in Digits \quad (85)$$

10.2 Little Killer

TODO Diagram

In a little killer sudoku, the same applies. The region is defined by a diagonal line labelled on the side with the total. There is no explicit uniqueness restriction. Numbers can repeat but there are the restrictions for uniqueness in boxes. The implementation is the same as that for a killer sudoku.

In a quadruple sudoku at a junction of cells, there will be up to four digits specified. For example if the digits 678 appear in a circle, there must be at least one 6, one 7 and one 8 in the surrounding four cells. Digits can repeat.

The cells form a 4 cell region, R. If the digits are $D = \{a, b, c, d\}$ then we can add these constraints.

$$\sum_{(r,c) \in R} V_{rcv} \geq 1, \forall v \in D \quad (86)$$

- Disjoint Groups
- Clones
- Sandwich
- Skyscraper
- Jigsaw
- Alternating Parity along a line
- Snakes
- Water
- Rossini
- Indexing
- Pointing Evens Clover CTC 2022-01-27

- Liar
- nadir CrackingTheCryptic
- BALANCING SUDOKU
- Slingshot A number in an arrow cell shows how many cells away the digit from which the arrow originates appears.
- Unique Sums Each cage sums to a unique number
- X-Factor Clues outside the grid show the product of the first X digits reading up, where X is the first digit.
- Taxicab Digits cannot appear a taxicab distance away from themselves

Preprocessing removes the redundant conditions. For example the value variables. Putting these in makes the model readable and easier to code.

Preprocessing with book keeping.

Implementation

Pyomo

Yaml choices

Examples

Performance

Preprocessing reductions

Unique solutions

Branching counts

Book keeping

11 Formulations

11.1 Abs

To implement an absolute function, $z = |f(x)|$ we need to reformulate into two expressions: $z = x \text{ if } x \geq 0 \text{ and } -z = x \text{ if } x < 0$

11.2 Not equal

11.3 In Set

11.4 Not In Set

Anti-Knight. No digit may be a knight's move away from an identical digit. Anti-King. No digit may be a king's move away from an identical digit. Anti-Queen. No digit that follows the rule may be a queen's move away from an identical move. Thermometers. In a thermometer, the lowest digit is at the bulb end and strictly increases along its length. Non Consecutive. No digit can be orthogonally adjacent to a consecutive digit. Magic Square. In the marked

3x3 box, every row, column and diagonal must sum to the same number. Killer Sudoku: Cells are grouped into cages with dotted outlines. The small number in the top left of each cage gives the sum of all digits inside it. Cages cannot contain repeated digits.

Square Quadruples. Each circled number gives between 1 and 4 digits that must be included in the surrounding four cells. Digits can repeat subject to other restrictions. Diagonals: The cells along a marked diagonal must contain the digits 1 to 9. Taxicab. The taxicab distance between two cells is the minimum number of orthogonal moves needed to travel from one cell to the other. [Manhattan distance] Kropki. A black dot between two cells means a digit on one side is double the digit on the other side. [Not all black dots are given] Complete or incomplete Arrowheads. An arrowhead between two cells points at the lower digit. Arrow Sum: The digit in the circle on an arrow is the sum of the other digits on the arrow. [Unique?] Multi digit Arrow Sum. Little Killer. Each clue with an arrow outside the grid shows the sum of the digits along the direction of the arrow. Digits can repeat subject to other restrictions. Jousting knights: Every digit must be a knights move away from at least one identical digit. Dominos: The sum of any two orthogonally adjacent digits can never be a number that is divisible by 5. 5,10,15 Killer Sudoku Hidden Sum. All cages sum to the same number that is to be determined. Palindromes. Market lines are palindromes. The digits along the line read the same in both directions. SumX. Orthogonally adjacent digits do not sum to 10 unless there is an X between them SumV. Orthogonally adjacent digits do not sum to 5 N-Sums: N-Products: Consecutive Lines: Sandwich Sums Odd Digits Even Digits Diagonal Pair Sums. Not all circles given CTC p111 Galactic Thermos. 159 Sudoku GAS 238 – Battenburg Normal sudoku rules apply. Every time a blue square appears at the intersection of four cells, the odd and even digits in those four cells must be in a 'checkerboard' pattern (two odd digits diagonally opposite each other, and two even digits diagonally opposite each other). Every time a white square appears at an intersection of four cells, the odd and even digits in those four cells must NOT be in a 'checkerboard' pattern. Frozen Thermometer

