An Interactive Application to Solve Sudoku Variant Puzzles

Helmut Simonis, Luis Quesada

University College Cork, Cork, Ireland

ModRef 2026, July 2026

A World Leading SFI Research Centre

Insight SFI RESEARCH CENTRE FOR DATA ANALYTICS

HOST INSTITUTIONS

PARTNER INSTITUTIONS

FUNDED BY:

Key Points

- Solving Sudoku Variant Puzzles
- Wide variety of constraints besides `alldifferent`, over 200 handled
- Interested in propagation, not search
- Off-the-shelf solver, Choco
- Human interaction by stating constraints
- Solving 97% of Sudoku 2025 Grand-Prix puzzles
- Help humans to enjoy puzzles, not replace humans!

Insight ~~is~~ was one of the largest data research and innovation centres in Europe…

Insight Today

4
Co-Lead Universities
4 partner institutions

Built on **20** years of research in Data Analytics and AI

1250
Academics, Postdocs, PhDs, RAs

3900+
Scientific conference and journal papers

175+
Funded collaborations with industry partners

350+
Research Awards

16
Spin out companies
107 license agreements

135+
H2020 consortia, 500+ collaborations, 40+ countries

1,137+ school visits, 28,000 students

398
PhDs graduated

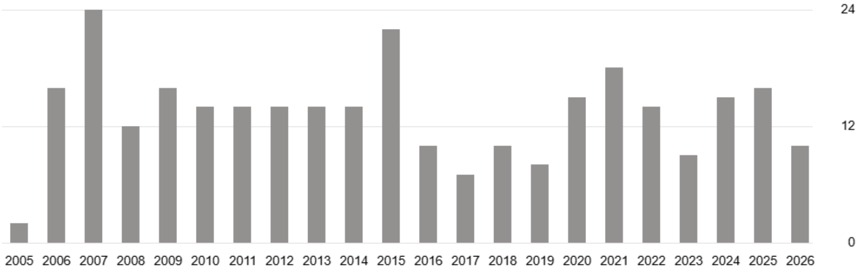
New and Improved: Rinn Artificial Intelligence: Research & Innovation in Data Science and AI

Rinn Artificial Intelligence is designed to serve as a national hub for research and innovation in data science and AI. The programme has two overarching goals: (i) to advance foundational cutting-edge research in Data Science and AI, and (ii) to pursue open, international, and inter- and transdisciplinary research aimed at studying, designing, and implementing strategies to address societal challenges.

rinn
Artificial
Intelligence

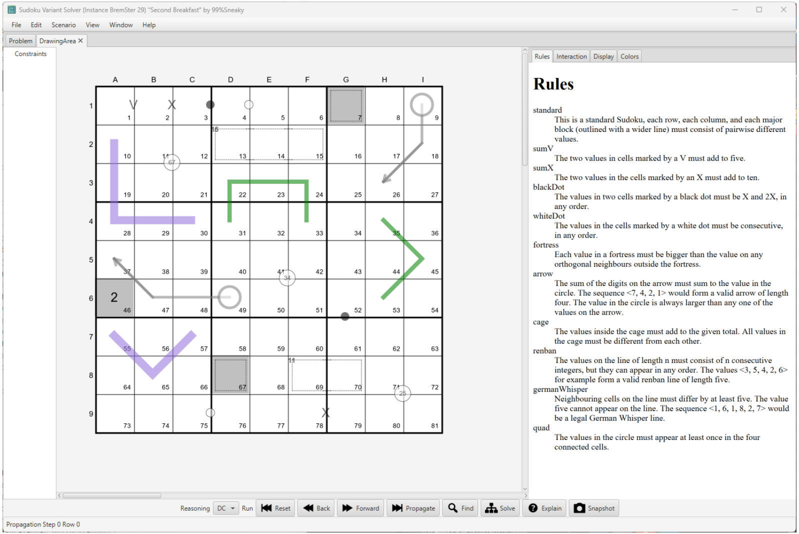
Going back 21 years, ModRef 2005

- H. Simonis. Sudoku as a Constraint Problem. ModRef 2005. Sitges, Spain.
- Classification of propagation strength needed to solve regular Sudokus
- Over 300 citations on Google Scholar

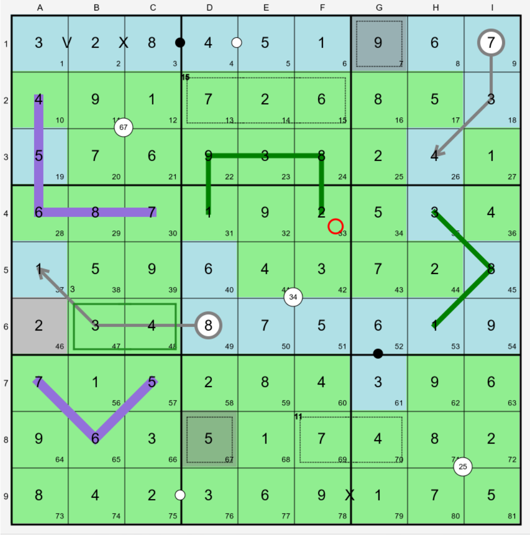


Outline

Running Example ("Second Breakfast" by "99%Sneaky")



What do the constraints mean?



Morphology

- Structural
  - Node(15)
    - Single(7)
    - Neighbors(4)
      - Orthogonal(2)
      - Diagonal(1)
    - Ray(4)
  - Pair(10)
  - Triple(2)
  - Quad(9)
    - Collection(3)
    - Ordered(6)
  - Line(41)
    - Collection(9)
    - Partial(15)
    - Ordered(17)
      - Pairs(7)
      - Subsequence(7)
  - MultiLine(2)
  - Area(15)
  - Outside(26)
  - Global(34)
    - Structure(19)
    - Strict(12)
    - Aggregate(3)

Outline

The Problem

Modelling

Manual Constraints

Results

Summary

Modelling Choices

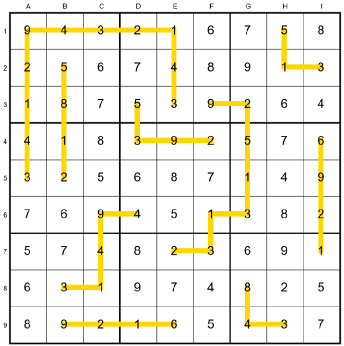
- One-to-one link to global constraint
- Conjunction of basic constraints
- Disjunction
- Tablification
    - Achieve domain consistency of basic constraint

Example: Renban

- Meaning: Values on a line form a collection of distinct, consecutive integers
- Global constraint: `alldifferent_consecutive_values`
- $\textbf{renban}(X) \equiv \min(X, \textbf{Min}) \wedge \max(X, \textbf{Max}) \wedge \textbf{Max} - \textbf{Min} = |X| - 1$
- Table: 2 values 16 tuples, 8 values 80,640 tuples

Example: GreatestLine (Line (Collection) Constraint)

- The largest value on the line is greater than the sum of any other two cells on the line



Implementation: GreatestLine

- Using `sort` global constraint, plus inequality
- `sort(X,Y)`: Y is sorted X
- Quite difficult to express with basic constraints alone

$$\textbf{greatestLine}([X_1, X_2, \ldots, X_n]) \equiv sort([X_1, X_2, \ldots, X_n], [Y_1, Y_2, \ldots, Y_n]) \wedge Y_n \geq Y_{n-1} + Y_{n-2}$$

Insight

Example: Parity Sum (Outside Constraint)

- The clue outside the grid gives the sum of the initial cells in the grid, up to and including the first parity change.
- A one-digit outside clue could reference just the first cell, since that would be the first odd/even number in that direction.

		5	13		3						
		A	B	C	D	E	F	G	H	I	
3	1	1	2	8	3	4	9	6	7	5	
13	2	4	6	3	5	1	7	2	8	9	
	3	7	5	9	6	8	2	3	4	1	
3	4	3	4	6	2	5	8	1	9	7	
	5	5	1	7	4	9	3	8	2	6	
	6	9	8	2	7	6	1	5	3	4	4
	7	2	3	4	1	7	5	9	6	8	
	8	8	7	1	9	2	6	4	5	3	12
	9	6	9	5	8	3	4	7	1	2	3
							4		12	5	

Insight

Parity Sum Code for Choco

```
case ParitySum -> {
  IntVar[] x = lineVars;
  int v = oc.getValue();
  x[0].eq(v).or(
   x[0].mod(2).ne(x[1].mod(2)).and(x[0].add(x[1]).eq(v)),
   x[1].mod(2).ne(x[2].mod(2)).and(x[0].add(x[1],x[2]).eq(v)),
   x[2].mod(2).ne(x[3].mod(2)).and(x[0].add(x[1],x[2],x[3]).eq(v)),
   x[3].mod(2).ne(x[4].mod(2)).and(x[0].add(x[1],x[2],x[3],x[4]).eq(v)),
   x[4].mod(2).ne(x[5].mod(2)).and(x[0].add(x[1],x[2],x[3],x[4],x[5]).eq(v)),
   x[5].mod(2).ne(x[6].mod(2)).and(x[0].
        add(x[1],x[2],x[3],x[4],x[5],x[6]).eq(v))).post();
  }
```

Table Sizes (Examples)

	Number of Variables							
Table	2	3	4	5	6	7	8	9
Sum+AllDifferent	72	504	3,024	15,120	60,480	181,440	362,880	*
Sum	81	729	6,561	59,049	-	-	-	-
Renban	16	42	144	600	2,880	15,120	80,640	*
Between	-	168	672	3,192	16,800	94,488	556,512	3,390,072
Magic Square	-	-	-	-	-	-	-	41

Outline

The Problem

Modelling

Manual Constraints

Results

Summary

Manual Constraints

- Propagation alone is not sufficient for many instances
- Allow human to control solution process
- More than just fixing or removing values
- User interface components to create constraints on the fly

Proposed Manual Constraints

in$(< X_1, X_2, ..., X_n >, < V_1, V_2, ..., V_k >)$ We restrict the domain of all X_i variables to the set V.

mustContain$(< X_1, X_2, ..., X_n >, < V_1, V_2, ..., V_k >)$ For each value V_j, one of the X_i variables must take that value. This leads to a disjunction, but more reasoning can be performed.

mustNotContain$(< X_1, X_2, ..., X_n >, < V_1, V_2, ..., V_k >)$ None of the values V_j are possible for any of the X_i variables. We can remove these values from their domains.

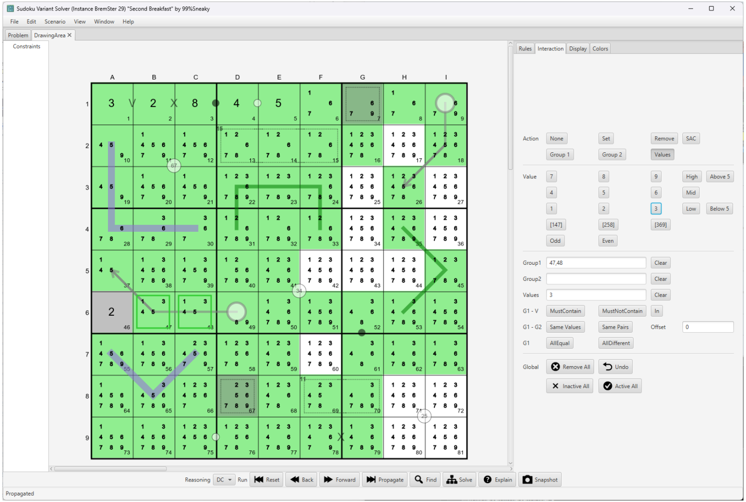
sameValues$(< X_1, X_2, ..., X_n >, < Y_1, Y_2, ..., Y_n >)$ The values in vectors X and Y must be the same. This often arises when considering the interaction of two `alldifferent` constraints.

samePairs$(< X_1, X_2, ..., X_n >, < Y_1, Y_2, ..., Y_n >, C)$ The values must match pairwise, i.e. $X_i = Y_i + c$. In most scenarios offset c is zero.

allEqual$(< X_1, X_2, ..., X_n >)$ All variables X_i must take the same values. This is a constraint often needed when a lot of initial reasoning is performed to find equivalence classes of cells in the grid.

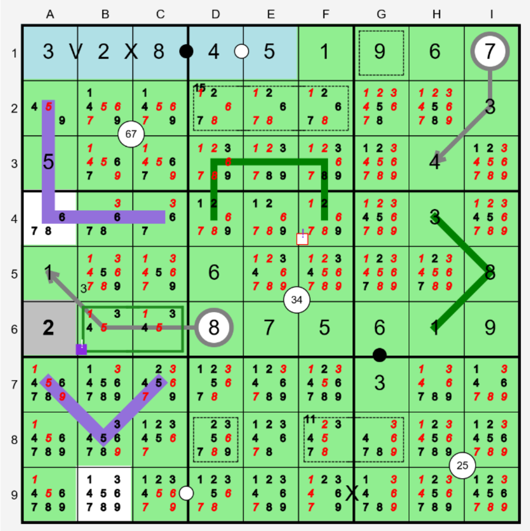
alldifferent$(< X_1, X_2, ..., X_n >)$ This allows to add additional, redundant `alldifferent` constraints. This can dramatically improve the propagation performed.

Adding Manual Constraint: C47 or C48 must be 3

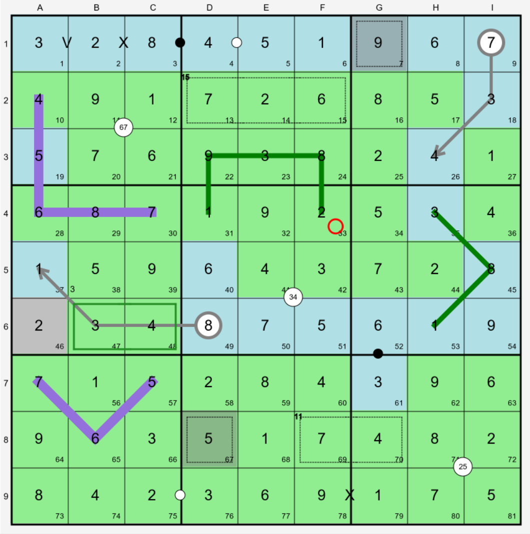


Insight

After Enforcing Manual Constraint



Solved Grid after Enforcing $C33 \neq 9$



Outline

The Problem

Modelling

Manual Constraints

Results

Summary

Four Benchmark Sets

- Nikoli regular Sudoku puzzle books
  - Can we solve popular, regular Sudokus with our system by propagation alone?
- Genuinely Approachable Sudoku (GAS) Puzzles, daily problems
  - Do we cover the variety of variants encountered?
  - Is our morphology comprehensive enough?
- WPF Sudoku Grand-Prix 2025
  - How do we compare to the best human speed-puzzle solvers?
- "Cracking the Cryptic" YouTube channel
  - Extremely popular, but often complex puzzles
  - Do we have the right manual constraints?

World Puzzle Federation: Sudoku Grand-Prix 2025

- Purely a pen and paper competition for humans
- No computers allowed
- 8 rounds, 90 minutes each for around 15 puzzles
- Points for each problem solved, additional time-points when finishing early

Grand-Prix Results for Rounds 1-8

Round	Puzzles	Points Available	Solved	Points Achieved	Time Points
R1	13	660	13	660	899
R2	13	720	13	720	899
R3	18	600	17	555	0
R4	12	780	11	674	0
R5	15	750	15	750	899
R6	13	750	13	750	899
R7	14	660	14	660	899
R8	14	690	13	631	0
Total	112	5,610	109	5,400	4,495

Grand-Prix Evaluation

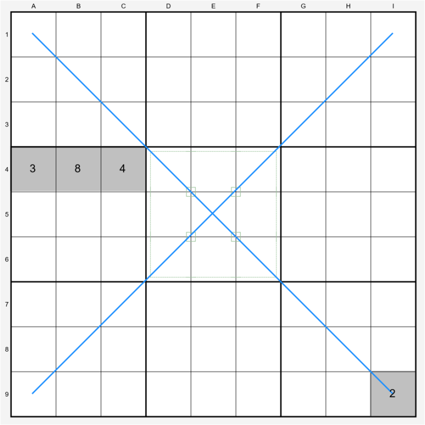
- All but three solved (109 of 112)
- Ranked 5th without including our time-points
  - Compared to human results including their time-points
- Ranked 1st including time-points
  - 9,895 points vs. 6,160.2 for best human

Cracking the Cryptic Examples

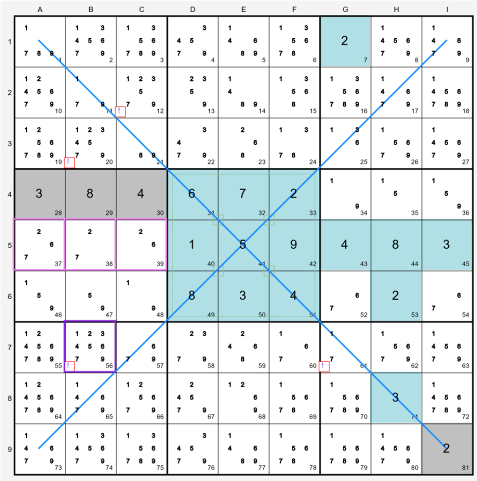
		Cells Filled		Manual Constraints Needed								
Nr	Instance	Givens	Propagated	=	≠	In	MC	MNC	SV	SP	AE	AD
52	Magic Square Sudoku	4	15	-	4	-	-	-	-	-	-	-
94	Non Consecutive Killer	1	0	4	1	-	-	-	2	-	3	-
30	The Miracle	2	12	-	15	-	-	-	-	-	-	-
33	Antiknight Sudoku	13	5	-	3	-	1	-	-	-	-	-
51	10 Million Views	4	13	-	-	-	-	3	-	-	-	-
56	Centipede	1	0	3	-	-	3	-	-	-	-	-
60	Spring Has Sprung	0	11	1	1	-	-	-	-	-	-	-
62	Game Of Arrows	3	2	2	1	-	1	-	-	-	-	-
68	The Climb	0	0	2	-	-	4	1	-	-	-	-
69	Simply Beautiful Sudoku	29	0	-	-	-	-	1	-	-	-	-
92	<= 5 Sudoku	11	7	-	2	-	-	-	-	-	-	-
93	Superking	6	3	1	-	-	-	-	-	-	-	-

Magic Square by Aad van de Wetering (10 million views on YouTube)

- Normal sudoku rules apply.
- In addition, both marked diagonals must also contain the digits 1 to 9.
- Cells that are a knight's move apart (in chess) cannot contain the same digit.
- The central 3x3 box must form a magic square.
- Only four hints are given



After Initial Propagation (Presenting Hint: $C56 \neq 2$)



Outline

The Problem

Modelling

Manual Constraints

Results

Summary

Summary

- Presented an interactive tool to solve Sudoku variant puzzles
- Over 200 constraints
- Wide, but still incomplete coverage of constraints found in the field
- Using off-the-shelf constraint solver
- Good results on variety of benchmark problems
- Aim: help human solve puzzles
- Potential for outreach

Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 12/RC/2289-P2, co-funded under the European Regional Development Fund. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.