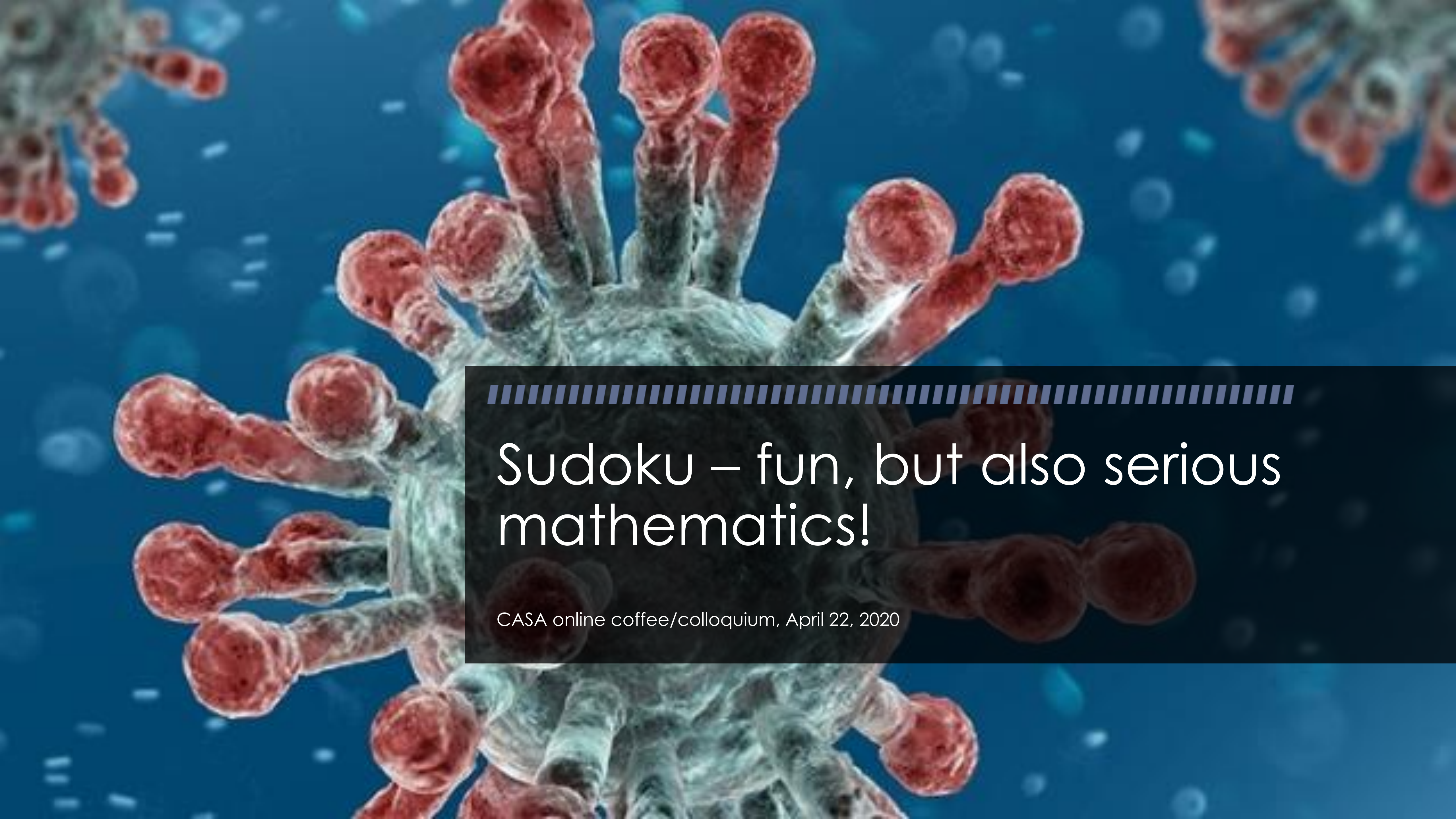


Sudoku – fun, but also serious mathematics!

CASA online coffee/colloquium, April 22, 2020

Rules of sudoku are clear, I hope…..

Every row, column and small square must contain the numbers 1-9 precisely once

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

Rules of sudoku are clear, I hope…..

The solution of a puzzle should be **provably unique**; multiple solutions are not allowed

Hans Zantema of our CS department knows a lot about constructing sudoku puzzles and reducing the number of solutions from many to 1 by changing the numbers and structure

Hans Zantema, "De achterkant van Sudoku"

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

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:

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

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:
  - In the 5^{th} column, 1,2,6 are missing

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

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:
  - In the 5th column, 1,2,6 are missing
  - 1 cannot go into (R4,C5) and (R6,C5), because there is already 1 in (R6,C4)

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

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:
  - In the 5th column, 1,2,6 are missing
  - 1 cannot go into (R4,C5) and (R6,C5), because there is already 1 in (R6,C4)
  - Hence, 1 must go into (R2,C5)

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

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:
    - In the 5^{th} column, 1,2,6 are missing
    - 1 cannot go into (R4,C5) and (R6,C5), because there is already 1 in (R6,C4)
    - Hence, 1 must go into (R2,C5)
    - But then 6 must be in (R4,C5) and 2 in the remaining cell (R6,C5)

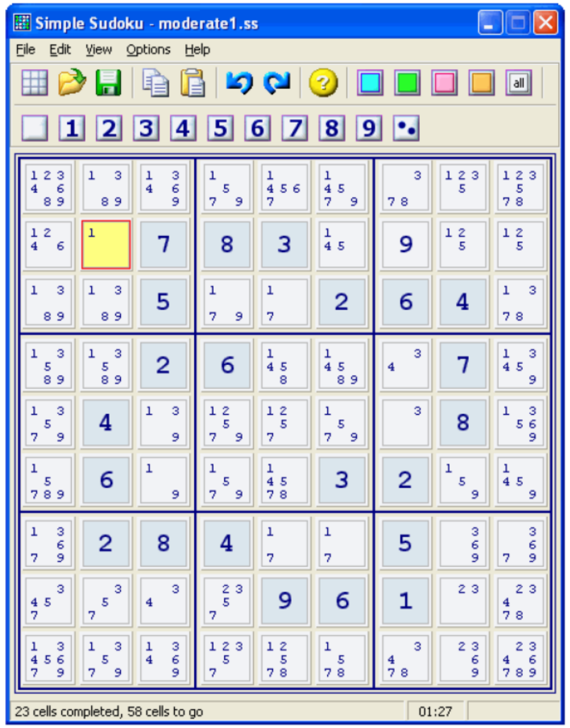
2	5	1	4	8			7	
9			5	1			2	
4		3	2	7	9			5
3			7	6	8	5	1	4
5	1		3	9	4		6	2
6			1	2	5		3	
8	4	5	6	3	1	2	9	7
7	3	6	9	5	2			
1	2	9	8	4	7	3	5	6

Simple logical steps

- Many sudokus can be started off by logical steps only, and the simpler categories can be solved entirely using such steps
- Consider this example:
  - In the 5th column, 1,2,6 are missing
  - 1 cannot go into (R4,C5) and (R6,C5), because there is already 1 in (R6,C4)
  - Hence, 1 must go into (R2,C5)
  - But then 6 must be in (R4,C5) and 2 in the remaining cell (R6,C5)
  - In this way, the sudoku can be completed with logical steps only

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

A mechanistic procedure



- Many books advocate writing down all remaining possibilities for each cell, like in the example on the left
- It is a lot of work, and often no strategies are provided on how to proceed with this information

Number pairs

- I advocate using only number pairs, like displayed in the puzzle on the right (grey cells were filled in with logical steps)
- Often, when progressing, you find number pairs that "correspond"
  - "46" in (R6,C2) & (R6,C3)
  - "59", "89" and "58" in row 6 indicate that 5,8,9 are in these cells
  - But then "15" in (R4,C4) should be reduced to a single digit "1"

	1		6					3
3				7				
5		9	4	3		2		
9	3	8	15 1	26				7
1	2	5	7			9		
7	46	46	59	89	58	3	2	1
4	57	3	2	1	9	8	57	6
			3	5	6			
6			8	4	7		3	

X-wing

- The puzzle I sent around for you to solve contains a well-known pattern termed the **X-wing**
- After progressing with logical steps, one ends up with the situation on the right, and one seems to be stuck

2	9	3	8	4	7	6	1	5
6		7	1	25	9	48	3	
48	1	45	3	25	6	9		7
9	6	28	4	7	5	1	28	3
1	7	48	6	3	2	48	5	9
3	45		9	8	1	7	6	24
7	48	1	2	6	3	5	9	48
45	2	6	57	9	8	3	47	1
58	3	9	57	1	4	2	78	6

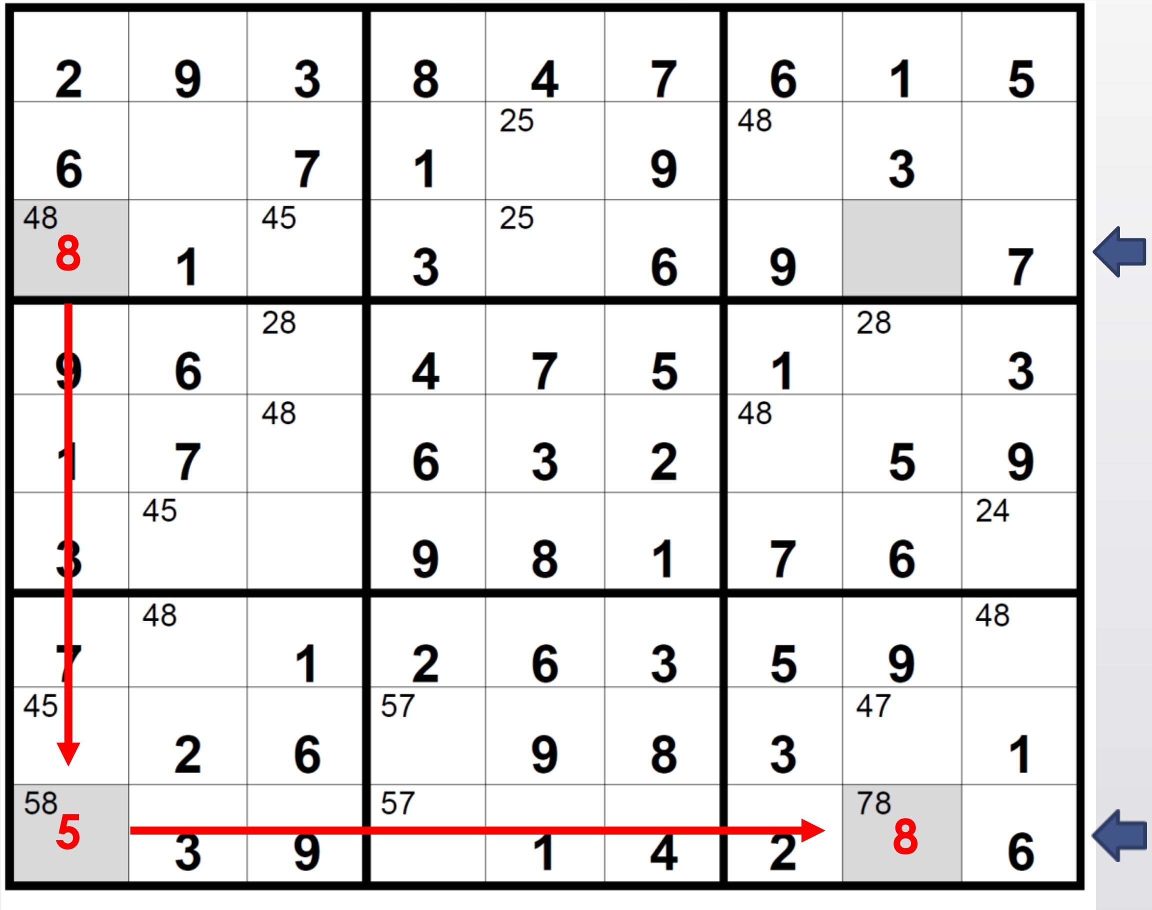
X-wing

- Look at the 3rd and 9th row:
  - The number 8 can only go in two positions: 1st and 8th column

2	9	3	8	4	7	6	1	5
6		7	1	25	9	48	3	
48	1	45	3	25	6	9		7
9	6	28	4	7	5	1	28	3
1	7	48	6	3	2	48	5	9
3	45		9	8	1	7	6	24
7	48	1	2	6	3	5	9	48
45	2	6	57	9	8	3	47	1
58	3	9	57	1	4	2	78	6

X-wing

- Look at the 3rd and 9th row:
  - The number 8 can only go in two positions: 1st and 8th column
  - 8 in (R3,C1) implies 5 in (R9,C1) and hence 8 in (R8,C8)



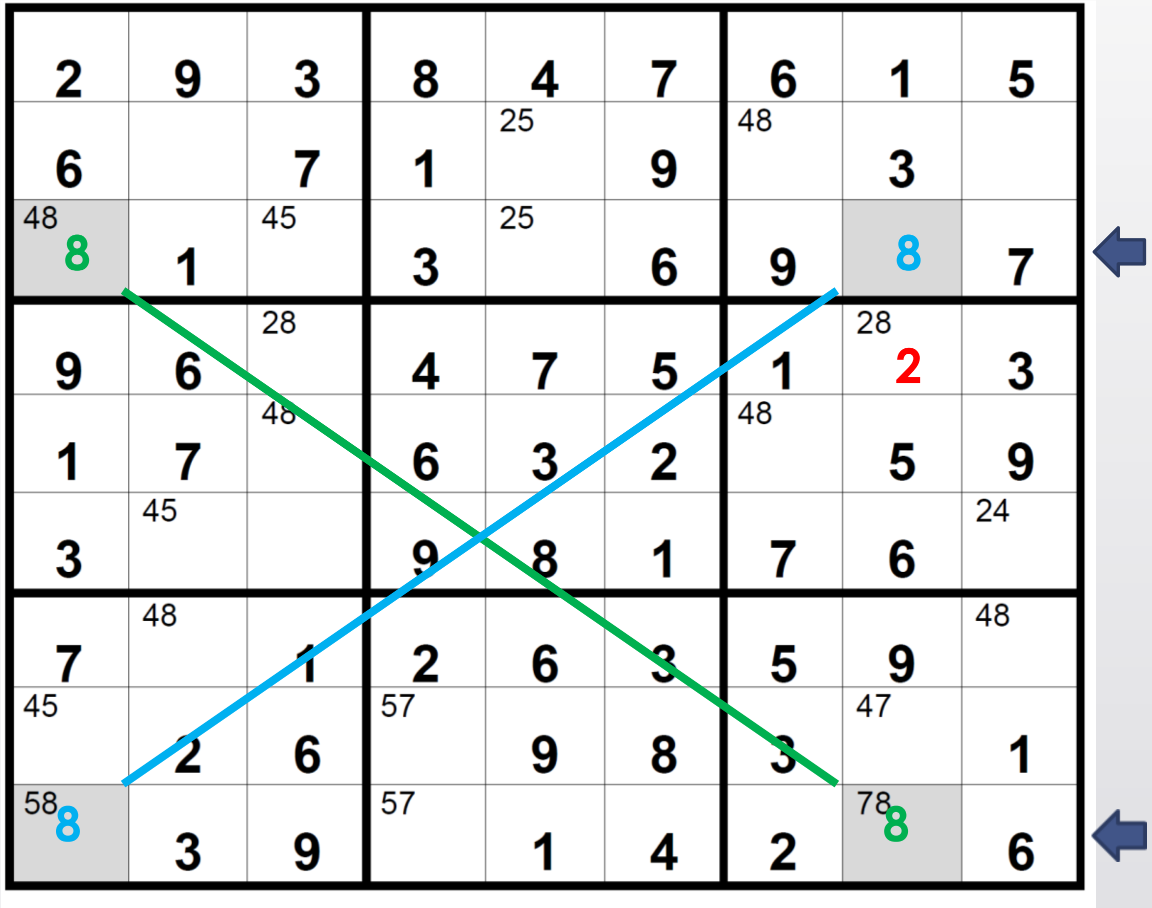
X-wing

- Look at the 3rd and 9th row:
  - The number 8 can only go in two positions: 1st and 8th column
  - 8 in (R3,C1) implies 5 in (R9,C1) and hence 8 in (R8,C8)
  - 8 in (R3,C8) implies 7 in (R9,C8) and hence 8 in (R9,C1)

2	9	3	8	4	7	6	1	5
6		7	1	25	9	48	3	
48	1	45	3	25	6	9	8	7
9	6	28	4	7	5	1	28	3
1	7	48	6	3	2	48	5	9
3	45		9	8	1	7	6	24
7	48	1	2	6	3	5	9	48
45	2	6	57	9	8	3	47	1
58 8	3	9	57	1	4	2	78 7	6

X-wing

- Look at the 3rd and 9th row:
  - The number 8 can only go in two positions: 1st and 8th column
  - 8 in (R3,C1) implies 5 in (R9,C1) and hence 8 in (R8,C8)
  - 8 in (R3,C8) implies 7 in (R9,C8) and hence 8 in (R9,C1)
  - Conclusion:
    - the 8's are positioned in a cross-like fashion
    - In columns 1 and 8 there is always an 8 in one of the two grey cells



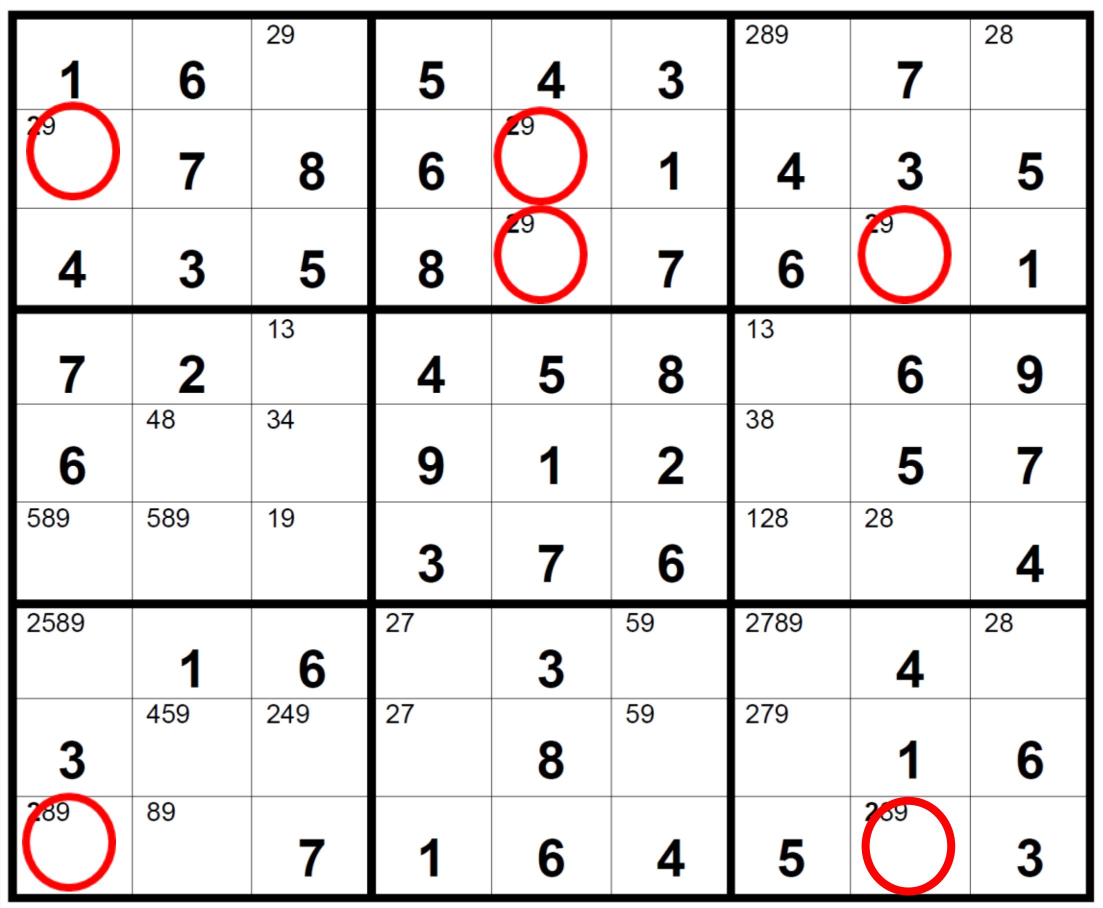
X-wing

- Look at the 3rd and 9th row:
  - The number 8 can only go in two positions: 1st and 8th column
  - 8 in (R3,C1) implies 5 in (R9,C1) and hence 8 in (R8,C8)
  - 8 in (R3,C8) implies 7 in (R9,C8) and hence 8 in (R9,C1)
  - Conclusion:
    - the 8's are positioned in a cross-like fashion
    - In columns 1 and 8 there is always an 8 in one of the two grey cells
  - The latter means that "8" in (R4,C8) can be omitted as option; hence, there must be a 2

2	9	3	8	4	7	6	1	5
6		7	1	(25)	9	(48)	3	
(48)	1	(45)	3	(25)	6	9		7
9	6	(28)	4	7	5	1	(28) 2	3
1	7	(48)	6	3	2	(48)	5	9
3	(45)		9	8	1	7	6	(24)
7	(48)	1	2	6	3	5	9	(48)
(45)	2	6	(57)	9	8	3	(47)	1
(58)	3	9	(57)	1	4	2	(78)	6

Many more patterns

- People have identified many different patterns that can lead to new numbers to be found in sudokus
- On the right we see the "swordfish", which is an extension of the X-wing with 3 rows and 3 columns



Many more patterns

- People have identified many different patterns that can lead to new numbers to be found in sudokus
- On the right we see the “swordfish”, which is an extension of the X-wing with 3 rows and 3 columns
  - In this case, it leads to the elimination of the 2 in (R6,C8), hence 8 remains
- Other patterns: Y-wing, W-wing, jellyfish, aligned pair exclusion, Exocet, Sue-de-coq, ……

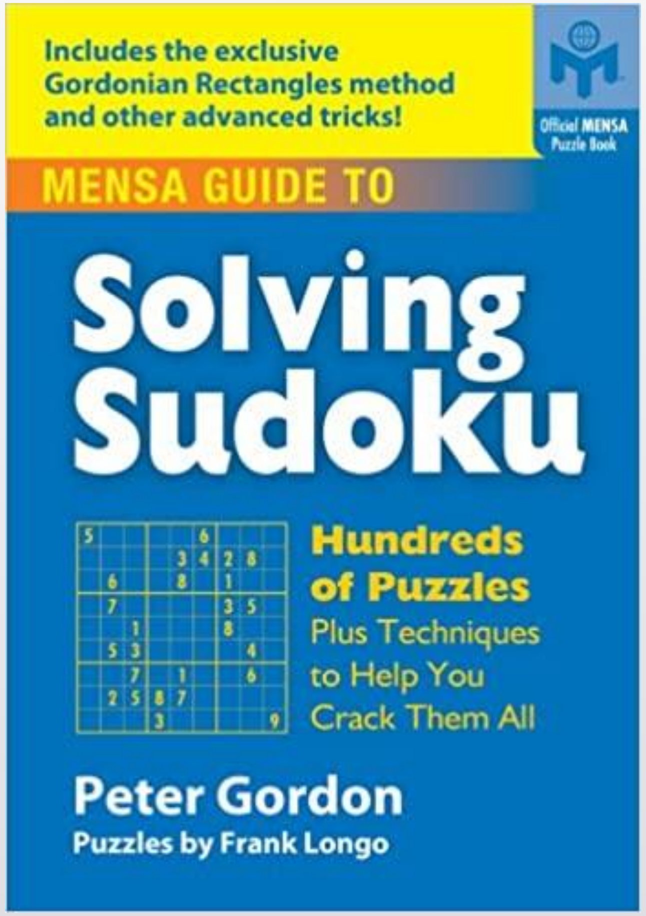
1	6	29	5	4	3	289	7	28
29	7	8	6	29	1	4	3	5
4	3	5	8	29	7	6	29	1
7	2	13	4	5	8	13	6	9
6	48	34	9	1	2	38	5	7
589	589	19	3	7	6	128	28 8	4
2589	1	6	27	3	59	2789	4	28
3	459	249	27	8	59	279	1	6
289	89	7	1	6	4	5	289	3

Read more about sudoku patterns

The following website can solve sudoku puzzles and has a lot of information about patterns:

https://www.sudokuwiki.org

I am not a fan of these patterns, as they can only be detected by computers



A last resort – intelligent guessing

- For the more difficult sudokus, the logical steps are often exhausted at some point, so we are **stuck**
- Then we could choose a number pair, and explore both paths
  - In the example on the right, we choose the cell (R2,C1) and explore the paths starting with a 2 and a 6 in that cell
- Convenient notation:
  - use a pencil, write the double digit number "26" in the cell
  - Now proceed, putting double digit numbers in other cells, the left digit corresponding to the choice 2 in (R2,C1) and the right digit corresponding to the 6

5	8	24		12	7	49	6	3
26 26	9	7	3	4	26	8	1	5
1	36		5	68		49	7	2
	5		12		3	7	9	4
4	7	26	8	9	26	3	5	1
3	1	9	7	5	4	6	2	8
7	4						38	6
	36		4				38	79
89	2	38	6		5	1	4	79

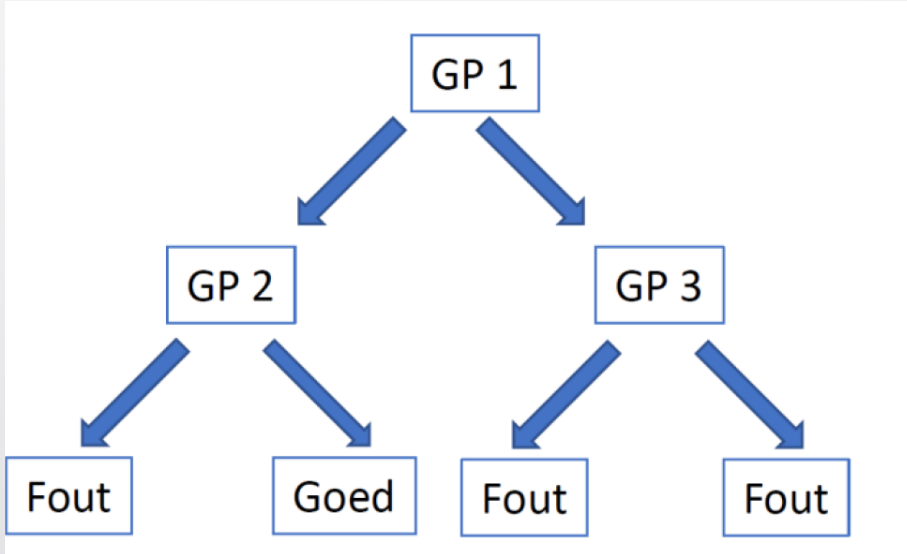
A last resort – intelligent guessing

- When doing this, we end of with the result displayed here
- We observe that several cells contain the same two digits; this means that this number is a certainty in that cell
- Hence, we can rub out all double digit numbers, and put a “4” in (R1,C3), a “9” in (R1,C7) and a “4” in (R3,C7)
- We can now proceed with simple logical steps and solve the sudoku without any further problems

5	8	24 44		12 X1	7	49 99	6	3
26 26	9	7	3	4	26 62	8	1	5
1	36 X3		5	68 8X		49 44	7	2
	5		12 1X		3	7	9	4
4	7	26 62	8	9	26 26	3	5	1
3	1	9	7	5	4	6	2	8
7	4						38	6
	36 X6		4				38	79
89	2	38	6		5	1	4	79

A last resort – intelligent guessing

- All sudokus can be solved with this methodology
  - Sometimes, we will encounter the **same two digits** in a cell, hence we can erase all pencil entries and put the number by pen in that cell
  - Sometimes, we will not encounter any cells with the same two digits; in that case, one of the options leads to a **contradiction**, and we can proceed with the other option
  - For very difficult sudokus, one may need to repeat this procedure, so that we can go several levels deep; only 1 path is correct in the end



World’s most difficult sudoku (2012)

- Was designed by Finnish mathematician Arto Inkala
- No digit can be found with logical steps, so stuck immediately
- There is only 1 cell with two options, namely (R8,C7) with the number pair “39”
- Sudoku turns out to be 5 levels deep when solving with the last resort method

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

World’s most difficult sudoku (2019)

- Even more difficult is this sudoku, designed by Veit Elser from Cornell University
- It does not even have any number pairs, only triplets (and more)
- Last resort can start with a triplet, all three options lead to number pairs, hence we can continue with levels of 2 options:
  - 3 x 2 x 2 x ……. paths

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

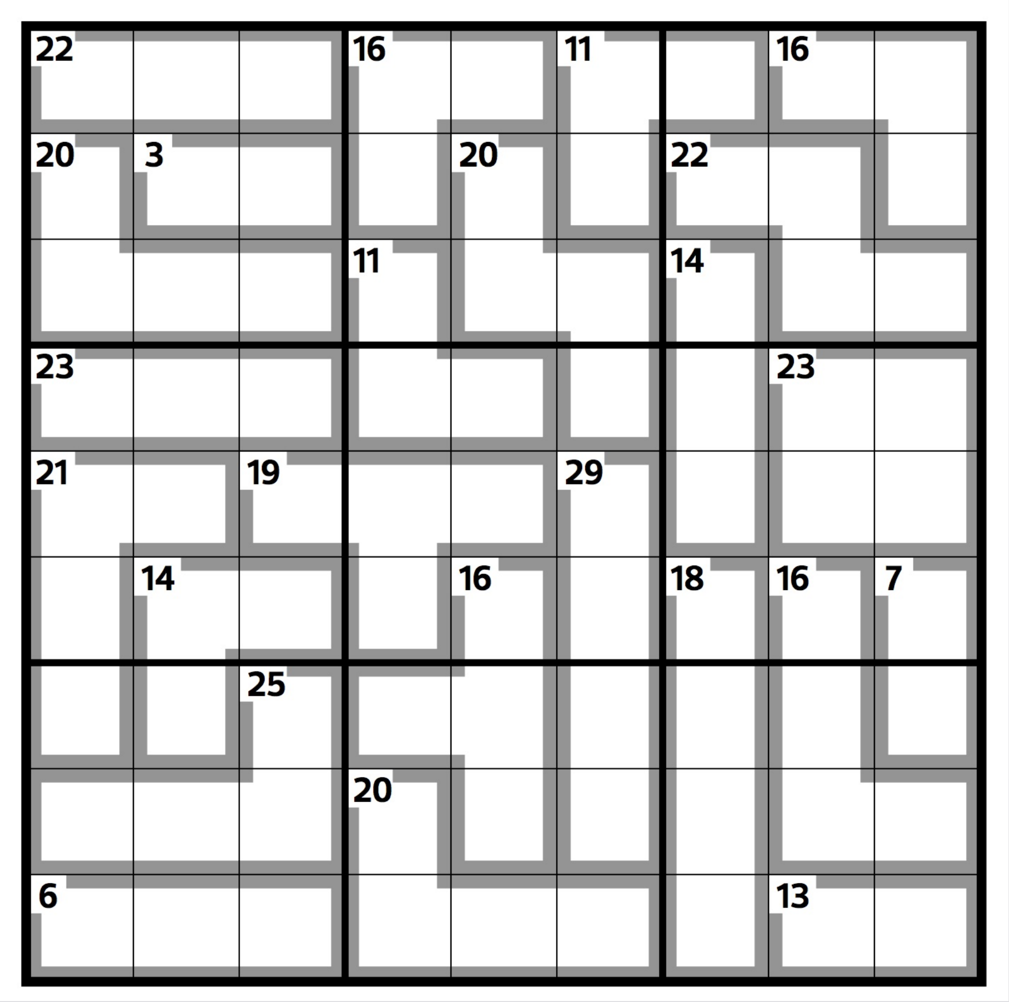
Alternative sudokus

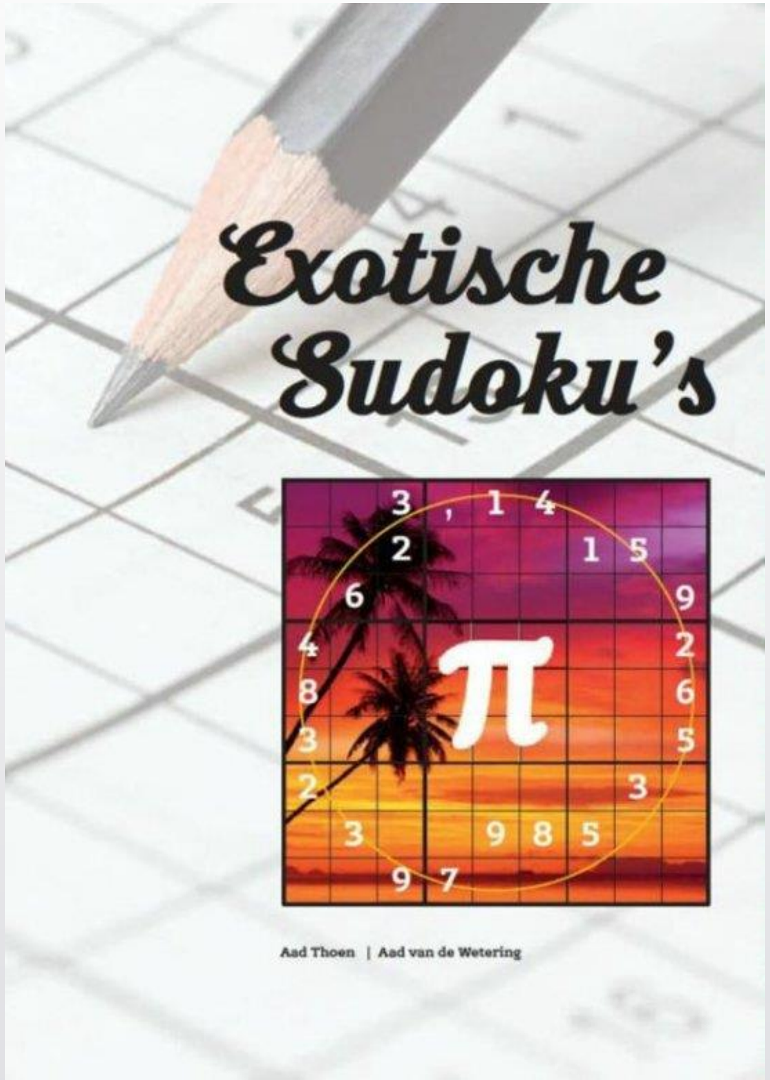
3	9	5	4		7	8		
6	8	1			3		7	
7	4	2	5					6
							8	
	3		8		5			
		6	3	7	2		1	4
		8				1	5	9
4					6	2	3	8
5		3		2		6	4	7

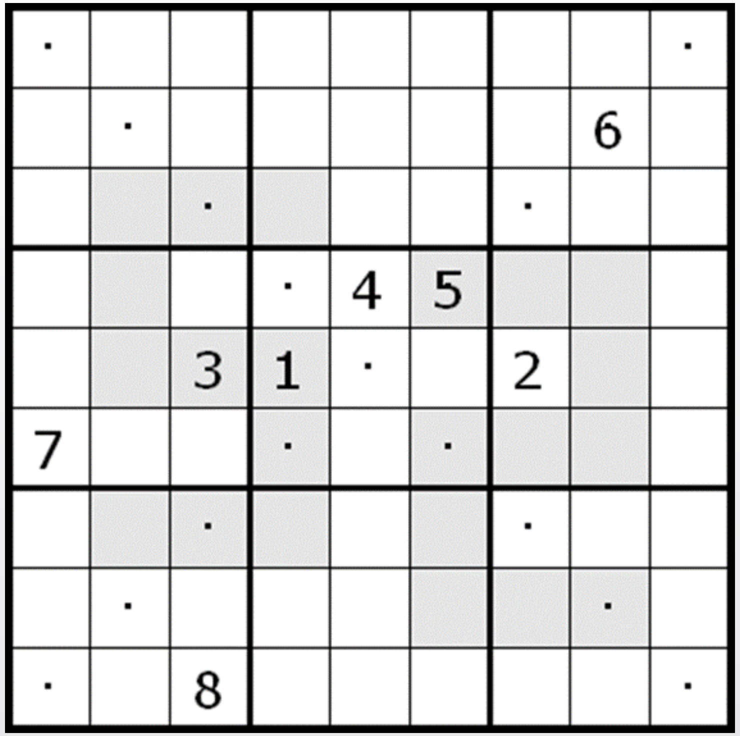
						2		
					4		6	
	4		5				1	
	6			5		3		9
		2						
		7			3			
			9		2	6	5	
				8			4	
			6					

Killer sudoku

- No numbers given
- The sum of numbers in a certain area is provided
    - For example, last row: clearly, the numbers 1,2,3 should be in the first three cells
    - In second row, numbers 1 and 2 in positions 2 and 3 (write down number pair)







Diagonale buren zijn ongelijk.
In de grijze vakjes staan oneven cijfers.

Serious mathematics for sudoku

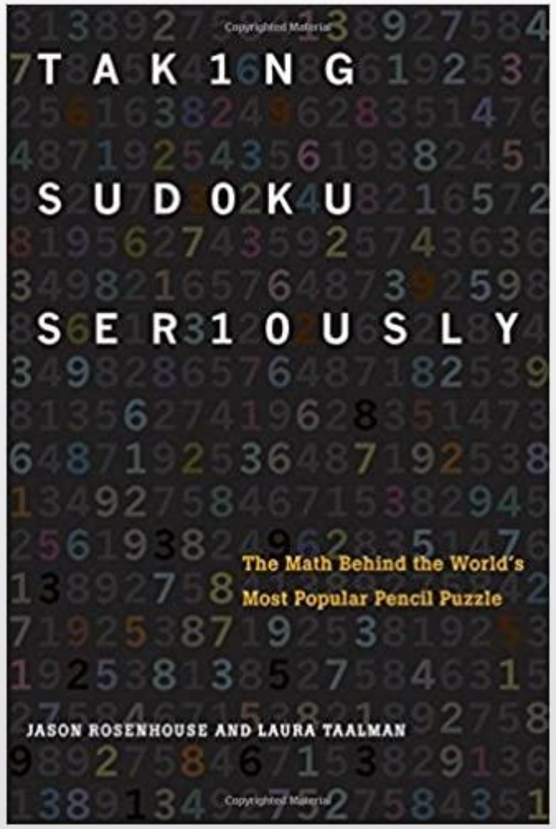
From a mathematical point of view, several questions associated with sudoku

- How many numbers do we need to specify at least in order to produce a correct sudoku (i.e. unique solution)?
- Which patterns of prescribed numbers will lead to correct sudokus?
  - The Japanese design sudokus by hand, and know many patterns that will lead to unique solutions
- What is the total number of distinct sudokus?
- Is there always at least 1 number pair in a uniquely solvable sudoku?
  - No (but the sudoku by Veit Elser is the only exception I know to date)
- In case of the "last resort" method, what is the meaning of the double digit numbers occurring after a while? ("Two paths coming together at some point")

Minimum number of clues in sudoku is 17

- This was proved by Irish mathematician Gary McGuire of UCD in 2012
- No-one ever came up with a 16 digit sudoku, so this strengthened the belief that 17 is the minimum number of clues
- That led to the **conjecture** that 16-clue puzzles with unique solutions simply do not exist.
- A potential way to demonstrate that could be to check all possible completed grids for every 16-clue puzzle, but that would take too much computing time.
- McGuire simplified the problem by designing a 'hitting-set algorithm’:
  - Search for what he calls unavoidable sets, or arrangements of numbers within the completed puzzle that are interchangeable and so could result in multiple solutions.
  - To prevent the unavoidable sets from causing multiple solutions, the clues must overlap, or 'hit', the unavoidable sets.
  - Once the unavoidable sets are found, it is a much smaller—although still non-trivial—computing task to show that no 16-clue puzzle can hit them all.
- Having spent two years testing the algorithm, McGuire and his team used about **7 million CPU hours** at the Irish Centre for High-End Computing in Dublin, searching through possible grids with the hitting-set algorithm.

Comments about the proof



*A consequence of the approach taken is that it will take some time for others to get enough computing time to check the proof, says Laura Taalman, a mathematician also at James Madison University, who co-authored the book **Taking Sudoku Seriously: The Math Behind the World's Most Popular Pencil Puzzle** with Rosenhouse.*

Interesting consequences of the proof

- McGuire says that his approach may pay off in other ways. The hitting-set idea that he developed for the proof has been used in papers on **gene-sequencing analysis** and **cellular networks**, and he looks forward to seeing if his algorithm can be usefully adapted by other researchers.

Nature doi:10.1038/nature.2012.9751

Stefan Heine in Germany publishes magazines and books containing sudoku puzzles with only 17 clues

Number of distinct sudokus

- The first known solution to complete enumeration was posted by Guenter Stertenbrink in 2003,obtaining 6,670,903,752,021,072,936,960 (6.67×10^{21}) distinct solutions
- “Distinct” means that at least 1 number in the puzzle is different; symmetry relations (such as rotations) are not taken into account, they count as different
- In a 2005 study, Felgenhauer and Jarvis calculated the number of distinct sudokus by mathematical means (group theory), ending up with the number 6,670,903,752,021,072,936,960, confirming the value obtained by Stertenbrink
- This number is equal to $9! \times 722 \times 27 \times 27{,}704{,}267{,}971$, the last factor of which is prime
- <u>NOTE:</u> Bertram Felgenhauer is a very talented mathematician, won the IMO silver and gold medal in 1995 resp 1996.

Solving sudoku with Matlab

The MathWorks News&Notes

CLEVE'S CORNER

Solving Sudoku with MATLAB

By Cleve Moler

Human puzzle-solvers and computer programs use very different Sudoku-solving techniques. The fascination with solving Sudoku by hand derives from the discovery and mastery of a myriad of subtle combinations and patterns that provide hints about the final solution. It is not easy to program a computer to duplicate these human pattern-recognition capabilities. For this reason, most Sudoku-solving programs take a very different approach, relying on the computer's almost limitless capacity to carry out brute-force trial and error. That is the approach that I used for the MATLAB® program.

Michiel Hochstenbach created a superfast Matlab program!

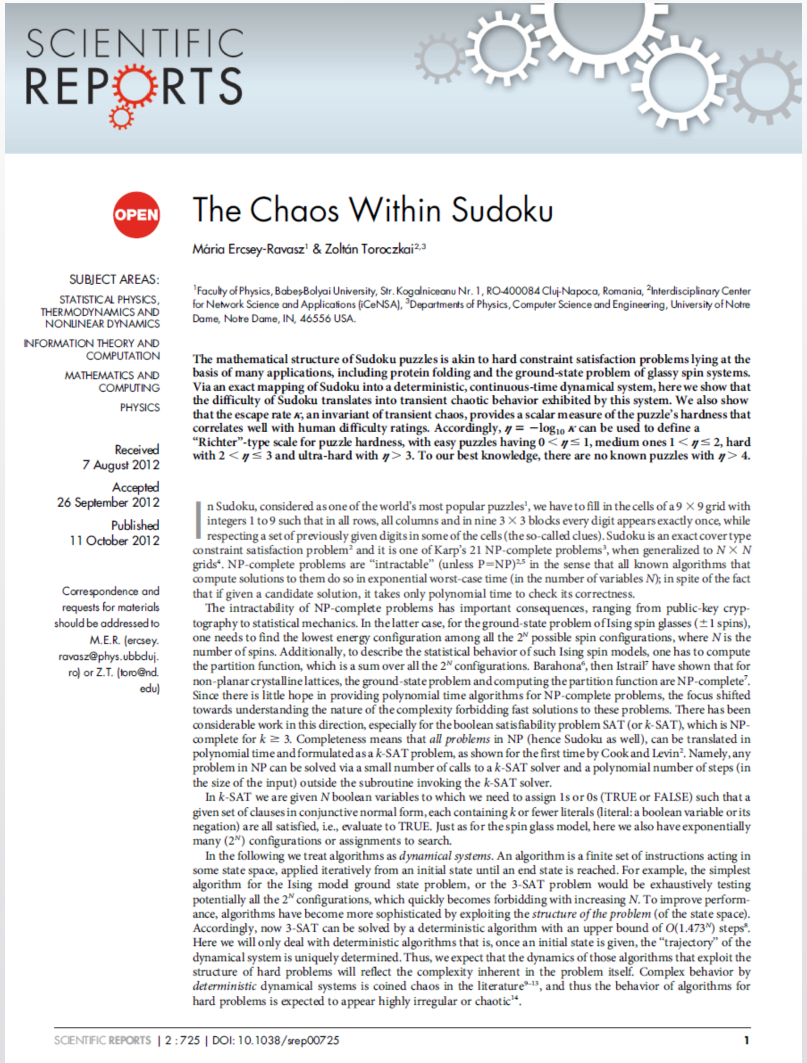
Numerical approach?

- All sudokus can be easily solved by a brute force approach; smart phones can take a picture, and will analyse all possible solutions until the correct one is found
- I was wondering whether it is possible, for a given sudoku, to set up a number of equations, and then solve those by numerical methods
- Clearly, we can start with 27 linear equations (for 9 rows, columns, subsquares), but it turns out that the rank of this system of 27 equations is 21
  - Proof (M. Hochstenbach): taking 3 subsquares together, either in row or column direction, leads to a sum of 3 rows/columns, which we already had. Hence, 3 equations in row direction and 3 in column direction disappear, hence the rank is 27-2x3=21. For k^2 x k^2 sudoku, the rank is 3 k^2 – 2 k
- So non-linear equations will need to be added, and as a consequence Newton's method (or alternatives) will have to be used
- Can lead to multiple solutions, non-convergence; additional problem: solutions need to be integers

Any ideas? WELCOME!

In a similar direction……

- In 2012, on one of my travels (Sevilla), I was called by the Dutch newspaper “Trouw” who had come across the paper on the right (I did not know it yet)



SCIENTIFIC REPORTS

OPEN

The Chaos Within Sudoku

Mária Ercsey-Ravasz[1] & Zoltán Toroczkai[2,3]

[1]Faculty of Physics, Babeş-Bolyai University, Str. Kogalniceanu Nr. 1, RO-400084 Cluj-Napoca, Romania, [2]Interdisciplinary Center for Network Science and Applications (iCeNSA), [3]Departments of Physics, Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, 46556 USA.

SUBJECT AREAS:
STATISTICAL PHYSICS, THERMODYNAMICS AND NONLINEAR DYNAMICS
INFORMATION THEORY AND COMPUTATION
MATHEMATICS AND COMPUTING
PHYSICS

Received
7 August 2012

Accepted
26 September 2012

Published
11 October 2012

Correspondence and requests for materials should be addressed to M.E.R. (ercsey.ravasz@phys.ubbcluj.ro) or Z.T. (toro@nd.edu)

The mathematical structure of Sudoku puzzles is akin to hard constraint satisfaction problems lying at the basis of many applications, including protein folding and the ground-state problem of glassy spin systems. Via an exact mapping of Sudoku into a deterministic, continuous-time dynamical system, here we show that the difficulty of Sudoku translates into transient chaotic behavior exhibited by this system. We also show that the escape rate κ, an invariant of transient chaos, provides a scalar measure of the puzzle's hardness that correlates well with human difficulty ratings. Accordingly, $\eta = -\log_{10} \kappa$ can be used to define a “Richter”-type scale for puzzle hardness, with easy puzzles having $0 < \eta \leq 1$, medium ones $1 < \eta \leq 2$, hard with $2 < \eta \leq 3$ and ultra-hard with $\eta > 3$. To our best knowledge, there are no known puzzles with $\eta > 4$.

In Sudoku, considered as one of the world's most popular puzzles[1], we have to fill in the cells of a 9×9 grid with integers 1 to 9 such that in all rows, all columns and in nine 3×3 blocks every digit appears exactly once, while respecting a set of previously given digits in some of the cells (the so-called clues). Sudoku is an exact cover type constraint satisfaction problem[2] and it is one of Karp's 21 NP-complete problems[3], when generalized to $N \times N$ grids[4]. NP-complete problems are “intractable” (unless P=NP)[2,5] in the sense that all known algorithms that compute solutions to them do so in exponential worst-case time (in the number of variables N); in spite of the fact that if given a candidate solution, it takes only polynomial time to check its correctness.

The intractability of NP-complete problems has important consequences, ranging from public-key cryptography to statistical mechanics. In the latter case, for the ground-state problem of Ising spin glasses (± 1 spins), one needs to find the lowest energy configuration among all the 2^N possible spin configurations, where N is the number of spins. Additionally, to describe the statistical behavior of such Ising spin models, one has to compute the partition function, which is a sum over all the 2^N configurations. Barahona[6], then Istrail[7] have shown that for non-planar crystalline lattices, the ground-state problem and computing the partition function are NP-complete[7]. Since there is little hope in providing polynomial time algorithms for NP-complete problems, the focus shifted towards understanding the nature of the complexity forbidding fast solutions to these problems. There has been considerable work in this direction, especially for the boolean satisfiability problem SAT (or k-SAT), which is NP-complete for $k \geq 3$. Completeness means that *all problems* in NP (hence Sudoku as well), can be translated in polynomial time and formulated as a k-SAT problem, as shown for the first time by Cook and Levin[2]. Namely, any problem in NP can be solved via a small number of calls to a k-SAT solver and a polynomial number of steps (in the size of the input) outside the subroutine invoking the k-SAT solver.

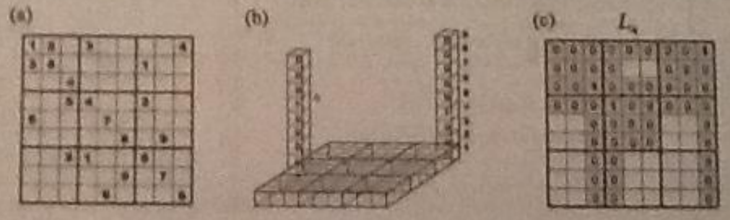
In k-SAT we are given N boolean variables to which we need to assign 1s or 0s (TRUE or FALSE) such that a given set of clauses in conjunctive normal form, each containing k or fewer literals (literal: a boolean variable or its negation) are all satisfied, i.e., evaluate to TRUE. Just as for the spin glass model, here we also have exponentially many (2^N) configurations or assignments to search.

In the following we treat algorithms as *dynamical systems*. An algorithm is a finite set of instructions acting in some state space, applied iteratively from an initial state until an end state is reached. For example, the simplest algorithm for the Ising model ground state problem, or the 3-SAT problem would be exhaustively testing potentially all the 2^N configurations, which quickly becomes forbidding with increasing N. To improve performance, algorithms have become more sophisticated by exploiting the *structure of the problem* (of the state space). Accordingly, now 3-SAT can be solved by a deterministic algorithm with an upper bound of $O(1.473^N)$ steps[8]. Here we will only deal with deterministic algorithms that is, once an initial state is given, the “trajectory” of the dynamical system is uniquely determined. Thus, we expect that the dynamics of those algorithms that exploit the structure of hard problems will reflect the complexity inherent in the problem itself. Complex behavior by *deterministic* dynamical systems is coined chaos in the literature[9–13], and thus the behavior of algorithms for hard problems is expected to appear highly irregular or chaotic[14].

Wiskundigen bedenken gouden Sudokuformule

■ Puzzel 'uitklappen' als flatgebouw ■ 'Nog niet praktisch voor puzzelaars'

Van onze redactie wetenschap



Van links naar rechts: een sudokupuzzel, cijfers uitklappen tot 'flatgebouw' en het beeld van de 'vierde verdieping'.

De meeste mensen hanteren een omslachtige en tijdrovende wijze om sudokupuzzels op te lossen. Dat zeggen computerwetenschappers van de Amerikaanse Notre Dame-universiteit. Ze publiceerden in het wetenschappelijke tijdschrift *Nature Scientific Reports* een artikel over de sudoku, de populaire puzzel waarin men de cijfers 1 tot en met 9 zo moet invullen dat deze in elke rij, kolom en in elk vierkant slechts één keer voorkomen.

De meest grove manier is natuurlijk om gewoon maar een cijfer in te vullen en vervolgens te kijken of het klopt, om bij falen weer opnieuw te beginnen. Deze gokstrategie brengt in een enkel geval succes, maar de kans is groter dat de puzzelaar eindeloos bezig is nieuwe gokjes te wagen. Fanatieke hobbyisten hebben pogingen gedaan om dit proces te versnellen met een computerprogramma, maar ook dat werkt niet altijd.

Daarom hanteren veel mensen een andere strategie: ze vullen in elk leeg vakje de mogelijke getallen in en gaan vervolgens kijken welke combinaties in de verschillende vakken onverenigbaar zijn. Helaas, dit levert niet alleen een beklad puzzelvel op, maar is in feite ook erg omslachtig, zeggen de onderzoekers.

Collega-wiskundigen hebben al geprobeerd om een serie vergelijkingen op te stellen, waarin alle hokjes in een rij optellen tot 45 – de cijfers 1 tot en met 9. Hetzelfde geldt voor de hokjes in iedere kolom en in ieder vierkant.

Bij een eenvoudige sudoku, waar al veel getallen ingevuld staan, kan dan met wiskunde van de middelbare school de waarde voor ieder hokje worden berekend. Bij een moeilijker type is dat onmogelijk.

> 'Het algoritme is te moeilijk om met het blote hoofd op te lossen'

De onderzoekers van Notre Dame beschouwden de sudoku als een soort flatgebouw – op ieder hokje zetten ze precies het aantal 'verdiepingen' van het cijfer dat uiteindelijk in het hokje moet komen te staan. Door de factor tijd te introduceren, komen ze tot een reeks instructies waarmee de sudoku stapsgewijs kan worden opgelost, waarbij het aantal stapjes stijgt naarmate de puzzel moeilijker is.

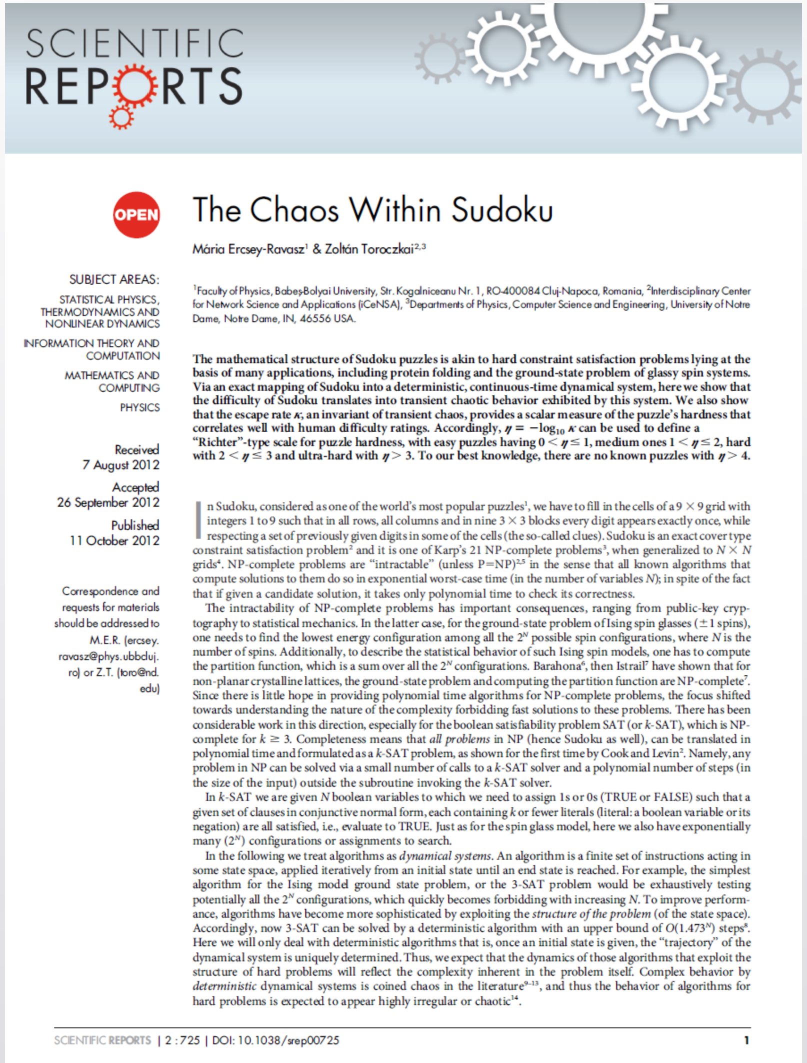
"Het is prachtig dat dit algoritme bedacht is, en de link met serieuze en diepe wiskunde is aangetoond", zegt de Eindhovense hoogleraar wiskunde Wil Schilders. "Helaas zal het ons niet helpen bij het oplossen van een sudoku in de trein of het vliegtuig. Het algoritme is te moeilijk om met het blote hoofd op te lossen, zeker als het om honderden vergelijkingen gaat."

Hij zou puzzelaars toch eerder aanraden de sudoku met eenvoudiger middelen te lijf te gaan. Zoals, bij de wat moeilijker puzzels, de zoektocht naar een hokje met slechts twee mogelijkheden. Wie vanaf dat hokje begint te gokken, hoeft maximaal maar een keer opnieuw te beginnen.

Schilders, die een boek schreef met tips voor fanatieke sudokupuzzelaars, vermoedt dat bij iedere oplosbare sudoku er minstens één hokje is waar slechts twee getallen mogelijk zijn – maar om dat vermoeden te bewijzen is nieuw onderzoek nodig.

In a similar direction……

- The paper presents a **deterministic approach** for solving sudokus (rather than a brute force, or a guessing approach)
- It also leads to a classification strategy for the difficulty of sudokus
- Paper is by Zoltan Toroczkai en Maria Ercsey-Ravasz of Notre Dame Univ (USA) (Maria is also in Cluj, Romania) – theoretical physics department



SCIENTIFIC REPORTS

OPEN

The Chaos Within Sudoku

Mária Ercsey-Ravasz[1] & Zoltán Toroczkai[2,3]

[1]Faculty of Physics, Babeş-Bolyai University, Str. Kogalniceanu Nr. 1, RO-400084 Cluj-Napoca, Romania, [2]Interdisciplinary Center for Network Science and Applications (iCeNSA), [3]Departments of Physics, Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, 46556 USA.

SUBJECT AREAS:
STATISTICAL PHYSICS, THERMODYNAMICS AND NONLINEAR DYNAMICS
INFORMATION THEORY AND COMPUTATION
MATHEMATICS AND COMPUTING
PHYSICS

Received
7 August 2012

Accepted
26 September 2012

Published
11 October 2012

Correspondence and requests for materials should be addressed to M.E.R. (ercsey.ravasz@phys.ubbcluj.ro) or Z.T. (toro@nd.edu)

The mathematical structure of Sudoku puzzles is akin to hard constraint satisfaction problems lying at the basis of many applications, including protein folding and the ground-state problem of glassy spin systems. Via an exact mapping of Sudoku into a deterministic, continuous-time dynamical system, here we show that the difficulty of Sudoku translates into transient chaotic behavior exhibited by this system. We also show that the escape rate κ, an invariant of transient chaos, provides a scalar measure of the puzzle's hardness that correlates well with human difficulty ratings. Accordingly, $\eta = -\log_{10}\kappa$ can be used to define a "Richter"-type scale for puzzle hardness, with easy puzzles having $0 < \eta \leq 1$, medium ones $1 < \eta \leq 2$, hard with $2 < \eta \leq 3$ and ultra-hard with $\eta > 3$. To our best knowledge, there are no known puzzles with $\eta > 4$.

In Sudoku, considered as one of the world's most popular puzzles[1], we have to fill in the cells of a 9×9 grid with integers 1 to 9 such that in all rows, all columns and in nine 3×3 blocks every digit appears exactly once, while respecting a set of previously given digits in some of the cells (the so-called clues). Sudoku is an exact cover type constraint satisfaction problem[2] and it is one of Karp's 21 NP-complete problems[3], when generalized to $N \times N$ grids[4]. NP-complete problems are "intractable" (unless P=NP)[2,5] in the sense that all known algorithms that compute solutions to them do so in exponential worst-case time (in the number of variables N); in spite of the fact that if given a candidate solution, it takes only polynomial time to check its correctness.

The intractability of NP-complete problems has important consequences, ranging from public-key cryptography to statistical mechanics. In the latter case, for the ground-state problem of Ising spin glasses (± 1 spins), one needs to find the lowest energy configuration among all the 2^N possible spin configurations, where N is the number of spins. Additionally, to describe the statistical behavior of such Ising spin models, one has to compute the partition function, which is a sum over all the 2^N configurations. Barahona[6], then Istrail[7] have shown that for non-planar crystalline lattices, the ground-state problem and computing the partition function are NP-complete[7]. Since there is little hope in providing polynomial time algorithms for NP-complete problems, the focus shifted towards understanding the nature of the complexity forbidding fast solutions to these problems. There has been considerable work in this direction, especially for the boolean satisfiability problem SAT (or k-SAT), which is NP-complete for $k \geq 3$. Completeness means that *all problems* in NP (hence Sudoku as well), can be translated in polynomial time and formulated as a k-SAT problem, as shown for the first time by Cook and Levin[2]. Namely, any problem in NP can be solved via a small number of calls to a k-SAT solver and a polynomial number of steps (in the size of the input) outside the subroutine invoking the k-SAT solver.

In k-SAT we are given N boolean variables to which we need to assign 1s or 0s (TRUE or FALSE) such that a given set of clauses in conjunctive normal form, each containing k or fewer literals (literal: a boolean variable or its negation) are all satisfied, i.e., evaluate to TRUE. Just as for the spin glass model, here we also have exponentially many (2^N) configurations or assignments to search.

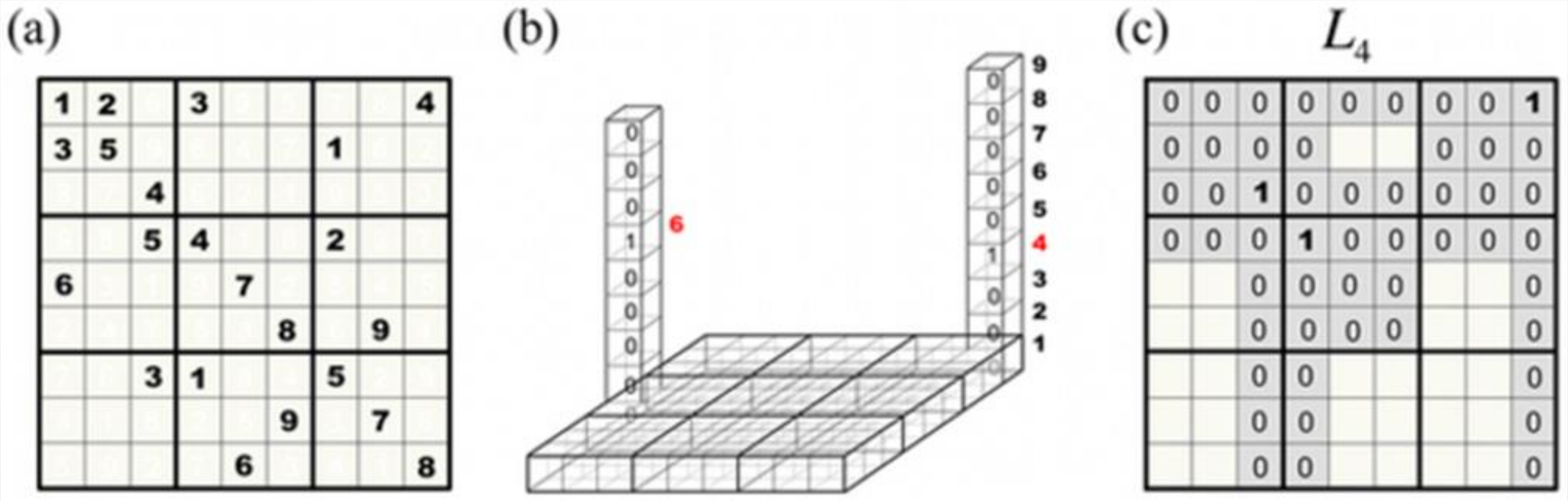
In the following we treat algorithms as *dynamical systems*. An algorithm is a finite set of instructions acting in some state space, applied iteratively from an initial state until an end state is reached. For example, the simplest algorithm for the Ising model ground state problem, or the 3-SAT problem would be exhaustively testing potentially all the 2^N configurations, which quickly becomes forbidding with increasing N. To improve performance, algorithms have become more sophisticated by exploiting the *structure of the problem* (of the state space). Accordingly, now 3-SAT can be solved by a deterministic algorithm with an upper bound of $O(1.473^N)$ steps[8]. Here we will only deal with deterministic algorithms that is, once an initial state is given, the "trajectory" of the dynamical system is uniquely determined. Thus, we expect that the dynamics of those algorithms that exploit the structure of hard problems will reflect the complexity inherent in the problem itself. Complex behavior by *deterministic* dynamical systems is coined chaos in the literature[9–13], and thus the behavior of algorithms for hard problems is expected to appear highly irregular or chaotic[14].

About the paper

- They start with a discussion on the so-called Boolean satisfiability problem, also termed SAT
  - Determining if there exists an interpretation that satisfies a given Boolean formula.
  - In other words, it asks whether the variables of a given Boolean formula can be consistently replaced by the values TRUE or FALSE in such a way that the formula evaluates to TRUE.
- SAT was the first problem that was proved to be **NP-complete** (Cool-Levin)
- The sudoku problem is then transformed into a so-called **k-SAT problem**, which is a special kind of SAT (and: k-SAT is NP-complete for k >= 3)

How it works

- Transform the sudoku into a 3-d structure where, in the n-th layer, a **1** is put in the position where the number n was given in the sudoku; **0** otherwise



- This makes it easier to check the rules of a sudoku, as every layer should satisfy the same rules; in addition, rules for interaction between layers are needed

k-SAT and dynamical systems

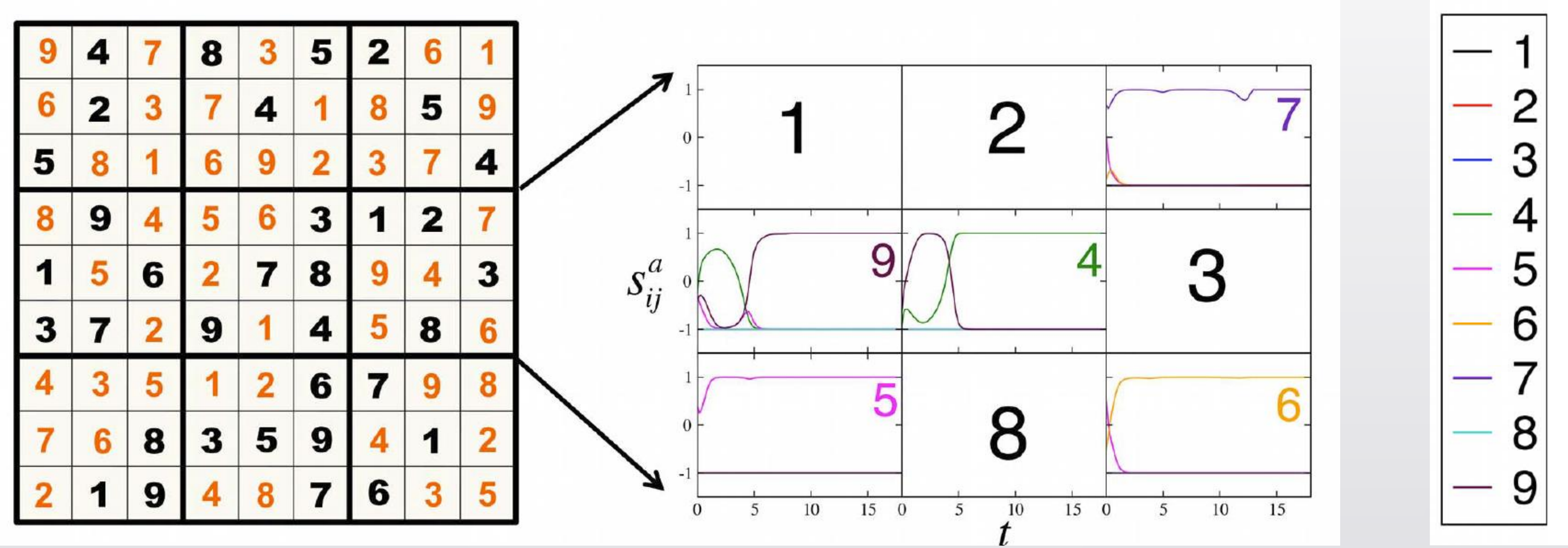
- In 2011, the authors provided a deterministic continuous-time solver for the Boolean k-SAT problem using **coupled ordinary differential equations** with a one-to-one correspondence between the k-SAT solution clusters and the attractors of the corresponding system of ODEs.

CASA heart

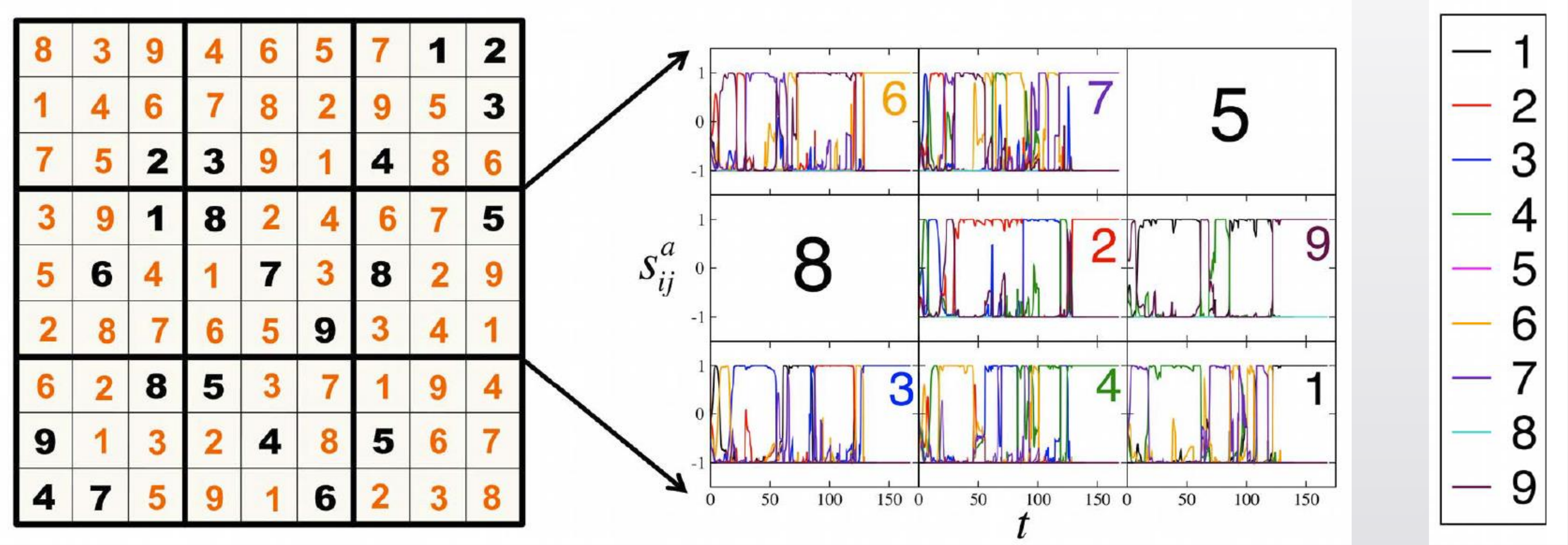
k-SAT and dynamical systems

- In 2011, the authors provided a deterministic continuous-time solver for the Boolean k-SAT problem using **coupled ordinary differential equations** with a one-to-one correspondence between the k-SAT solution clusters and the attractors of the corresponding system of ODEs.
- This continuous-time dynamical system is in a form naturally suited for chaos theory methods, and thus it allows studying the relationship between optimization hardness and chaotic behavior.
- In the paper the focus is only on solvable (satisfiable) instances, and thus the observed chaotic behavior will necessarily be transient; hence, the sudoku will be solved with a time-dependent solver!
- Sudoku hardness is then defined according to the complexity of solving the system of ODEs
- I will skip further details; if you are interested, you can get a copy of the paper

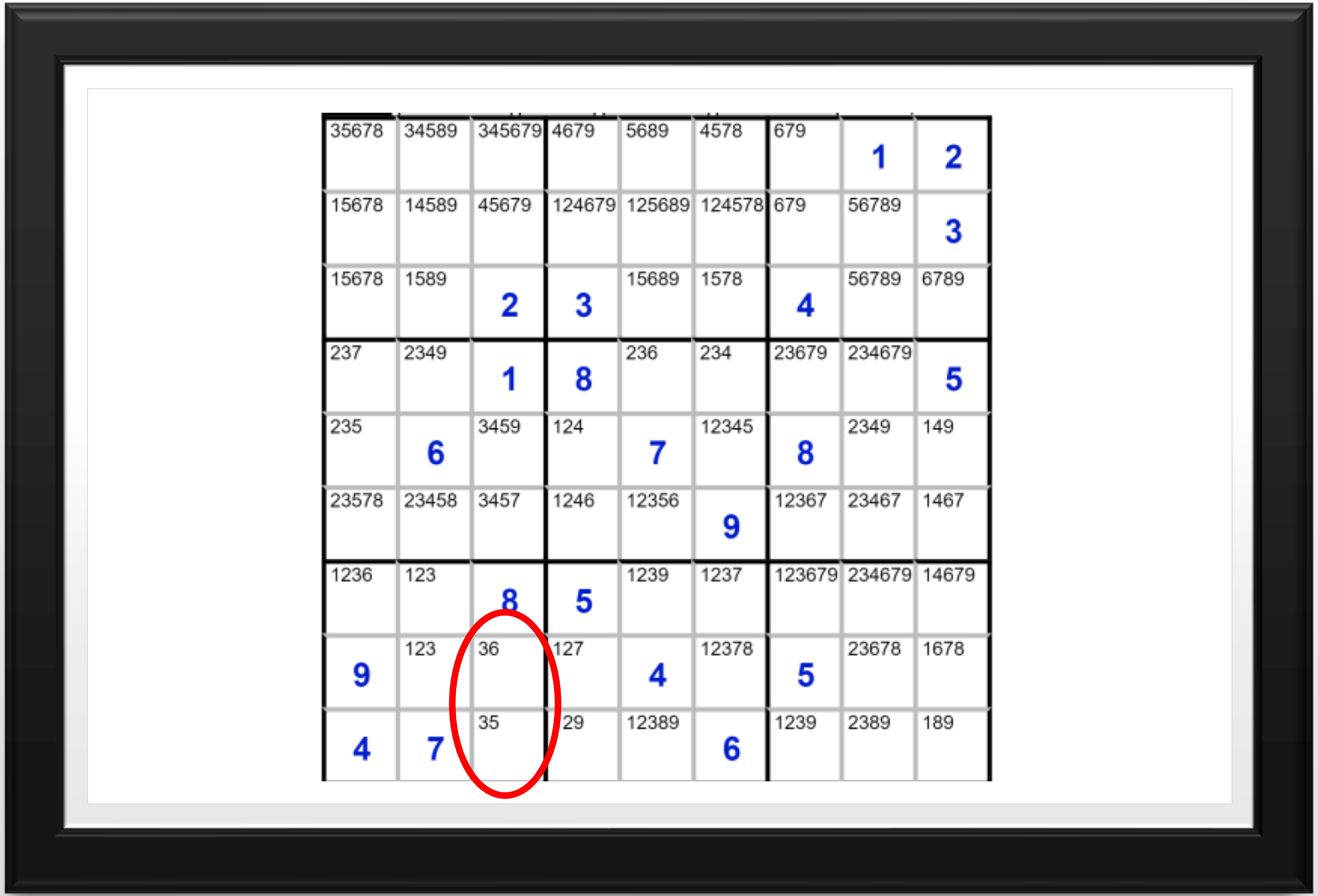
The solution of a simple sudoku



The solution of a difficult sudoku



Indeed it is a difficult one



35678	34589	345679	4679	5689	4578	679	1	2
15678	14589	45679	124679	125689	124578	679	56789	3
15678	1589	2	3	15689	1578	4	56789	6789
237	2349	1	8	236	234	23679	234679	5
235	6	3459	124	7	12345	8	2349	149
23578	23458	3457	1246	12356	9	12367	23467	1467
1236	123	8	5	1239	1237	123679	234679	14679
9	123	36	127	4	12378	5	23678	1678
4	7	35	29	12389	6	1239	2389	189

Graph coloring problems and sudoku

- Let *G(V,E)* be a graph with vertices (*V*) connected by edges (*E*).
- A **proper m-coloring** of G is a mapping c: V → K={1,…,m}, assigning one of the m possible colors to each vertex, such that no two adjacent vertices share the same color, i.e. c(i) =/ c(j) for all i,j, in *E*
- The graph coloring problem consists in determining whether it is possible to find a proper m-coloring of the graph G
- The graph coloring problem is **NP-complete**, hence approximate methods are needed to solve it
- Solution: Karger, D., Motwani, R., Sudan, M.: Approximate graph coloring by semidefinite programming. J. ACM (JACM) 45(2), 246–265 (1998) ***(KMS method)***

Douglas-Rachford splitting algorithm

- The Douglas–Rachford algorithm is a classical optimization method (originally a numerical method using finite differences) that has found many applications
  - J. Douglas, H.H. Rachford, "On the numerical solution of heat conduction problems in two and three space variables", Trans. Amer. Math. Soc., 82 (1956), pp. 421-439
- Veit Elser (Cornell) suggested (2012) to use the DR algorithm to iteratively find solutions to the semidefinite programming task representing the graph colouring problem
- Francisco J. Aragón Artacho, Rubén Campoy and Veit Elser (2019) proposed several alternative versions of the DR method, and used these to attack sudokus
- Again no details, people interested can ask me for the paper(s)

Experimental results of DR on 'nasty' sudoku

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

Time	Cubic		Binary		Rank	
	Inst.	Cumul.	Inst.	Cumul.	Inst.	Cumul.
0-24	12	12%	15	15%	61	61%
25-49	0	12%	2	17%	36	97%
50-99	0	12%	1	18%	3	100%
100-299	0	12%	1	19%	0	100%
Unsolved	81	88%	81	81%	0	0%

Fig. 12 Number of solved instances (right), among 100 random starting points, to find the solution of the 'nasty' Sudoku (left) by DR with the cubic, binary, and rank formulations. For each interval of time (in s), we show the number of solved instances and the cummulative proportion of solved instances for each formulation. The algorithm was stopped after a maximum of 5 min, in which case the problem was labeled as "Unsolved"

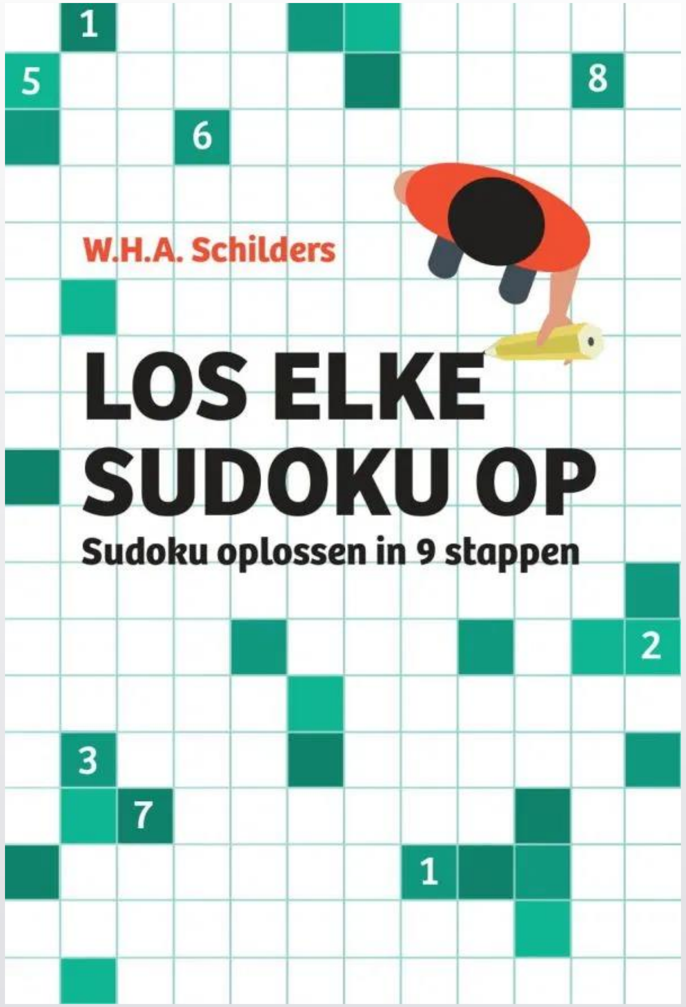
Conclusion

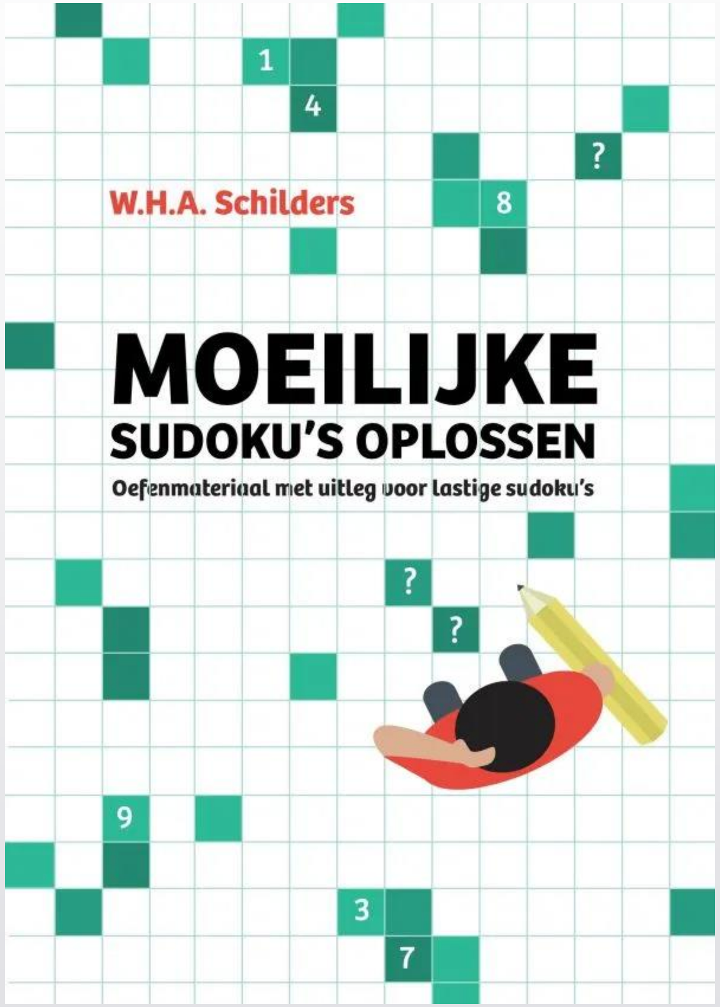
4		6	3	8			2	
5		3	7		4			
			9			8	4	3
2	3			1		9		
	4					5	7	1
	5		6	4	7			
9		1	4		8	3		
	6	4						7
8		5	1		3		9	2

Conclusion

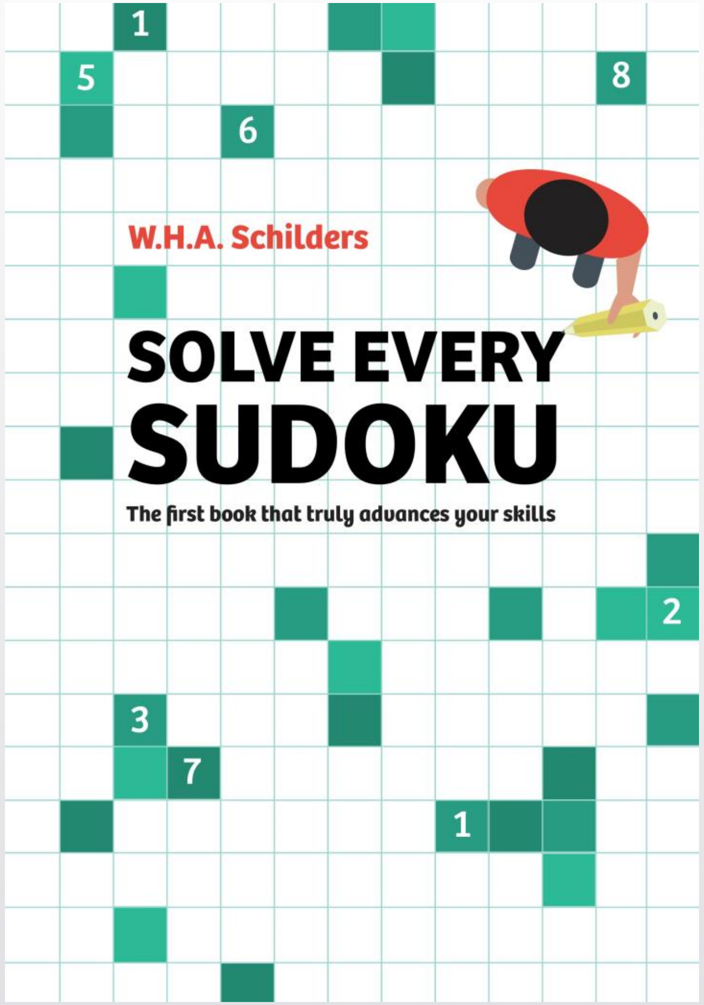
- Sudoku is not only fun, but also the subject of quite a few very interesting mathematical works
- Basic questions, such as the total number of distinct sudokus, or the minimal number of clues to be prescribed, have been solved
- More advanced questions, such as "which structures allow uniquely solvable sudokus" or "what is the meaning of ending up with the same number in a certain cell when following two distinct paths" are yet unanswered
- Deterministic ways of solving sudoku have been developed, by relating sudoku to graph colouring or Boolean satisfiability
- The question still stands whether we can find other deterministic ways, based upon sets of linear and nonlinear equations solved with numerical methods
- Typing "sudoku mathematics" in on google, many more interesting mathematical work on sudoku is found
- <u>(Nearly) Final note:</u> I discussed with Hans Zantema whether we could have an "NWO Klein" on sudoku

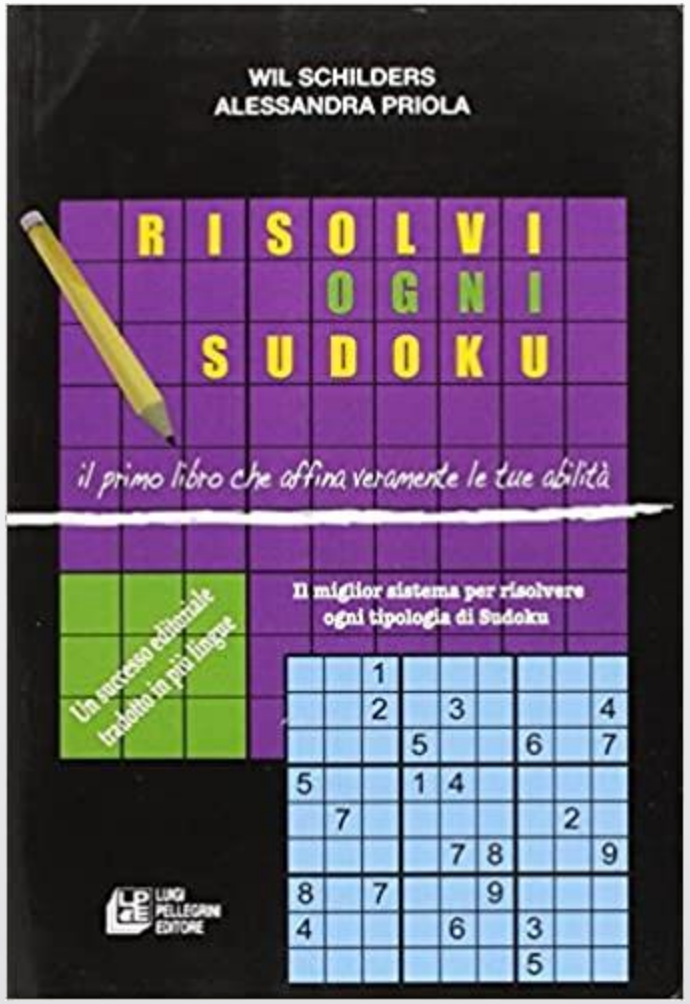
Commercial message





https://www.loselkesudokuop.nl





Translations underway:

- German (May)
- French (June)
- Spanish
- Portuguese

https://www.solveeverysudoku.com http://loesejedessudoku.de http://resoudrelessudokus.fr

Winners CASA sudoku challenge

- Mia Jukic (EN version)
- Han Slot (NL version)
- Karin Veroy-Grepl (EN version)
- Jim Portegies (NL version)

The books will come your way! (send me your home addresses)