

YOUR CHALLENGE: Insert a number from 1 to 9 in each cell without repeating any of those digits in the same row, column or subgrid (3 × 3 box). The solutions to the moderately difficult Sudoku in the center and to others in this article appear at www.sciam.com, along with additional puzzles.

The Science behind SUDOKU

Solving a Sudoku puzzle requires no math, not even arithmetic. Even so, the game poses a number of intriguing mathematical problems

BY JEAN-PAUL DELAHAYE

One might expect a game of logic to appeal to very few people—mathematicians, maybe, computer geeks, compulsive gamblers. Yet in a very short time, Sudoku has become extraordinarily popular, bringing to mind the Rubik's cube craze of the early 1980s.

Unlike the three-dimensional Rubik's cube, a Sudoku puzzle is a flat, square grid. Typically it contains 81 cells (nine rows and nine columns) and is divided into nine smaller squares containing nine cells each; call them subgrids. The game begins with numbers already printed in some cells. The player must fill in the empty cells with the numbers 1 to 9 in such a way that no digit appears twice in the same row, column or subgrid. Each puzzle has one unique solution.

Ironically, despite being a game of numbers, Sudoku demands not an iota of mathematics of its solvers. In fact, no operation—including addition or multiplication—helps in completing a grid, which in theory could be filled with any set of nine different symbols (letters, colors, icons and so on). Nevertheless, Sudoku presents mathematicians and computer scientists with a host of challenging issues.

Family Tree

ONE THING is not unresolved, however: the game's roots. The ancestor of Sudoku is not, as is commonly presumed, the magic square—an array in which the integers in all the rows, columns and

diagonals add up to the same sum. Indeed, aside from the numbers and the grid, Sudoku has almost nothing to do with the magic square—but everything to do with the Latin square [see box on next page].

A Latin square of order n is a matrix of n^2 cells (n cells on a side), filled with n symbols such that the same symbol never appears twice in the same row or column (each of the n symbols is thus used precisely n times). The origin of those grids dates back to the Middle Ages; later, mathematician Leonhard Euler (1707–1783) named them Latin squares and studied them.

A standard Sudoku is like an order-9 Latin square, differing only in its added requirement that each subgrid contain the numbers 1 through 9. The first such puzzle appeared in the May 1979 edition of *Dell Pencil Puzzles and Word Games* and, according to research done by Will Shortz, the crossword editor of the *New York Times*, was apparently created by a retired architect named Howard Garns. Garns died in Indianapolis in 1989 (or 1981; accounts vary), too early to witness the global success of his invention.

The game, published by Dell as “Number Place,” jumped to a magazine in Japan in 1984, which ultimately named it “Sudoku,” loosely translated as “single numbers.” The magazine trademarked that moniker, and so copycats in Japan used the “Number Place” name. In yet another Sudoku-related irony, then, the Japanese call the puzzle by its English name, and English speakers call it by its Japanese name.

Sudoku owes its subsequent success to Wayne Gould, a peri-

SUDOKU'S PREDECESSORS

A Sudoku grid is a special kind of Latin square. Latin squares, which were so named by the 18th-century mathematician Leonhard Euler, are $n \times n$ matrices that are filled with n symbols in such a way that the same symbol never appears twice in the same row or column. Two examples are shown. The standard completed Sudoku grid (also known as a solution grid) is a 9×9 Latin square that meets the additional constraint of having each of its nine subgrids contain the digits 1 to 9.

1	2	3	4
2	3	4	1
3	4	1	2
4	1	2	3

Small Latin square
($n=4$)

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

Latin square that is also a completed
Sudoku grid ($n=9$)



Leonhard Euler

patetic retired judge living in Hong Kong, who came across it while visiting Japan in 1997 and wrote a computer program that automatically generates Sudoku grids. At the end of 2004 the London *Times* accepted his proposal to publish the puzzles, and in January 2005 the *Daily Telegraph* followed suit. Since then, several dozen daily papers in countries all over the world have taken to printing the game, some even putting it on the cover page as a promotional come-on. Specialty magazines and entire books devoted to this diversion have sprung up, as have tournaments, Web sites and blogs.

As Many Grids as Humans

IT DID NOT TAKE LONG for mathematicians to begin playing “how many” games with Sudoku. For instance, they soon asked how many unique filled-in, or “solution,” grids can be constructed. Clearly, the answer has to be smaller than the number of Latin squares because of the added constraints imposed by the subgrids.

There are only 12 Latin squares of order 3 and 576 of order 4, but 5,524,751,496,156,892,842,531,225,600 of order 9. Group theory, however, says that a grid that can be derived from another is equivalent to the original. For instance, if I systematically replaced each number with some other (say, 1 became 2, and 2 became 7, and so on), or if I swapped two rows or columns, the final results would all be essentially the same. If one counts only the reduced forms, then the number of Latin squares of order 9 is 377,597,570,964,258,816 (a result reported in *Discrete Mathematics* in 1975 by Stanley E. Bammel and Jerome Rothstein, then at Ohio State University).

Exactly how many Sudoku grids can exist has been rather difficult to establish. Today only the use of logic (to simplify the problem) and computers (to examine possibilities systematically) makes it possible to estimate the number of valid Sudoku solution grids: 6,670,903,752,021,072,936,960. That number includes all those derived from any

particular grid by elementary operations. This result, from Bertram Felgenhauer of the Technical University of Dresden in Germany and Frazer Jarvis of the University of Sheffield in England, has now been verified several times. (Verification matters for results that are obtained this way.)

If we count only once those grids that can be reduced to equivalent configurations, then the number shrinks to 5,472,730,538—slightly lower than the population of humans on the earth. Despite this reduction, Sudoku devotees need not fear any shortage of puzzles.

Note that a complete Sudoku solution grid may be arrived at in more than one way from any starting, or clue, grid (that is, an incomplete grid whose solution is a given complete version). Nobody has yet succeeded in determining how many different starting grids there are. Moreover, a Sudoku starting grid is really only interesting to a mathematician if it is minimal—that is, if removing a single number will mean the solution is no longer unique. No one has figured out the number of possible minimal grids, which would amount to the ultimate count of distinct Sudoku puzzles. It is a challenge that is sure to be taken up in the near future.

Another problem of minimality also remains unsolved—to wit, what is the smallest number of digits a puzzle maker can place in a starting grid and still guarantee a unique solution? The answer seems to be 17. Gordon Royle of

Overview/Scientific Sudoku

- Sudoku is more than just an entertaining logic game for players; it also raises a host of deeper issues for mathematicians.
- Such problems include: How many Sudoku grids can be constructed? What is the minimal number of starting clues that will yield one unique solution? Does Sudoku belong to the class of hard problems known as NP-complete?
- Puzzle mavens have come up with an array of approaches to attacking Sudokus and with entertaining variations on the game.

the University of Western Australia has collected more than 38,000 examples that fit this criterion and cannot be translated into one another by performing elementary operations.

Gary McGuire of the National University of Ireland, Maynooth, is conducting a search for a 16-clue puzzle with a unique solution but has so far come up empty-handed. It begins to look as if none exists. On the other hand, Royle and others working independently have managed to find one 16-clue puzzle that has just two solutions. Searchers have not yet uncovered any additional examples.

Is anyone near to proving that no valid Sudoku puzzle can have only 16 clues? McGuire says no. If we could analyze one grid per second, looking for a valid 16-clue puzzle within it, he notes, “we could search all the grids in 173 years. Unfortunately, we cannot yet do this, even on a fast computer.” Soon, he says, searching a grid might be doable in one minute on a powerful computer, but at that rate the endeavor would take 10,380 years. “Even distributed on 10,000 computers, it would take about one year,” he adds. “We really need a breakthrough in our understanding to make it feasible to search all the grids. We either need to reduce the search space or find a much better algorithm for searching.”

Mathematicians do know the solution to the opposite of the minimum number of clues problem: What is the maximum number of givens that do not guarantee a unique solution? The answer is 77. It is very easy to see that with 80, 79 or 78 givens, if there is a solution, it is unique. But the same cannot be guaranteed for 77 givens [see bottom box on page 86].

Computer Solvers

BEYOND THE “HOW MANY” questions, mathematicians and computer scientists enjoy pondering what computers can and cannot do when it comes to solving and generating Sudoku puzzles. For standard Sudokus (9 × 9), it is relatively easy to write computer programs that solve all valid starting grids.

Another problem: What is the smallest number of digits that can be put in a starting grid and still guarantee a unique solution?

The solution programs can employ several methods, but the most common is backtracking, a systematic form of trial and error in which partial solutions are proposed and then modified slightly as soon as they are proved wrong.

The basic backtracking algorithm works like this: The program places the number 1 in the first empty cell. If the choice is compatible with the existing clues, it continues to the second empty cell, where it places a 1. When it en-

counters a conflict (which can happen very quickly), it erases the 1 just placed and inserts 2 or, if that is invalid, 3 or the next legal number. After placing the first legal number possible, it moves to the next cell and starts again with a 1.

If the number that has to be changed is a 9 (which cannot be raised by one in a standard Sudoku grid), the program backtracks and increases the number in the previous cell (the next-to-last number placed) by one. Then it moves for-

HOW LOW CAN YOU GO?

The minimum number of clues that a 9 × 9 Sudoku puzzle can start with and still yield a unique solution seems to be 17; an example is shown. One particular filled-in grid, known to Sudoku aficionados as “Strangely Familiar,” or SF, hides 29 inequivalent 17-clue starting boards—an unusually high number. SF was once considered the grid most likely to harbor a 16-clue puzzle with a unique solution, but an exhaustive search has dashed that hope. The only known 16-clue Sudoku having just two solutions appears at bottom; the final grids interchange the 8’s and 9’s.

One 17-Clue Puzzle

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

“Strangely Familiar” Grid

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

16-Clue Puzzle ...

5	2				4			
		7	1					3
				4	6			
7	2							
1								
6				2				
			3			1		
4								

... with Two Solutions

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

SOLUTION METHODS

Here are a few ways to try solving a Sudoku puzzle. Methods 1 and 2 are the simplest and are usually used in tandem (a bit of one, a bit of the other). Unfortunately, they do not always get one very far, so a player can add in method 3 and, if that proves insufficient, method 4—which works every time but not necessarily easily. You can also invent methods of your own and try the many approaches described on the Web.

a

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

b

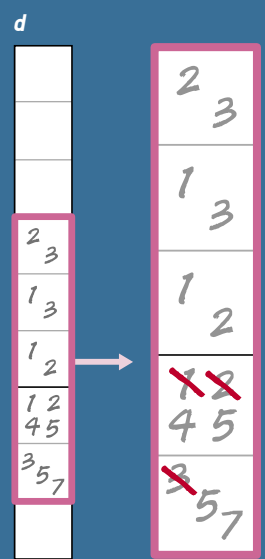
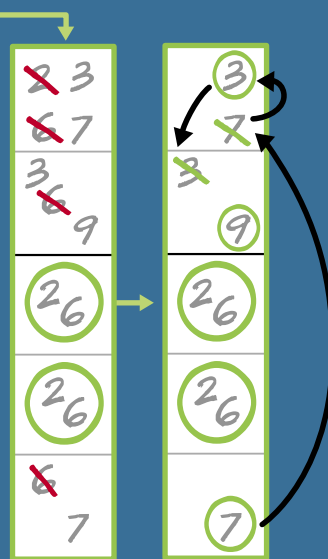
5		1					9	6
				9			5	
				7	5	2		7
4	9		1				7	
	5				7			
1	3	7						2
3	1	4	6	5	9	7		2
9	2	8		7	1			4
7	6	5	8	2				

"Only" cells (blue)

"Forced" cells (orange)

c

5	4 ₇₈	1	2 ₃	3 ₄₈	2 ₃	3 ₄₈	9	6
2 ₆₈	4 ₇₈	2 ₃	2 ₃	2 ₃	9	2 ₃₄	1 ₃	5
6 ₈₉	4 ₈	3 ₆₉	3 ₄₆	1 ₄₃	6 ₈	5	2	1 ₃₈
4	9	2 ₆	1	3 ₆₈	2 ₃	3 ₅₈	7	3 ₅₈
2 ₆₈	5 ₈	2 ₆	2 ₃₄	3 ₄	7	1 ₃₄₅	1 ₃	1 ₃₄
1	3	6 ₇	4 ₅	4 ₆₈	4 ₆₈	4 ₅₆	2	4 ₅
3	1	4	6	5	9	1 ₆	1 ₆₈	1 ₂₈
9	2	8	3 ₆	7	1	3 ₅	4	3 ₅₉
7	6	5	8	2	3 ₄	1 ₃₉	1 ₃	1 ₃₉



ward until it hits a conflict. (The program sometimes backtracks several times before advancing.) In a well-written program, this method exhaustively explores all possible hypotheses and thus finishes by finding the solution, if one exists. And if multiple solutions exist, as would be the case with a flawed puzzle, the program finds all of them.

Of course, refinements are possible, and they speed up the discovery of the unique solution. One favorite is called

constraint propagation: after each new number is placed, the program generates a table of the remaining possible numbers in each empty cell and considers only numbers from this table.

Backtracking techniques can be encoded by fairly short solution programs. Indeed, concise programs have been written for Sudoku in Prolog, a computer language incorporating a backtracking algorithm. Alain Colmerauer and Philippe Rousset of the University of Marseilles in

France invented Prolog in the late 1970s.

For human players, the backtracking techniques employed by computer programs would be infeasible; they would require extraordinary patience. So mortals use more varied and smarter rules and generally turn to trial and error only as a last resort. Some programs try to mimic human methods to an extent: they are longer than the others but work just as well. The programs that simulate human reasoning are also useful for assessing the difficulty of starting grids, which are ranked from "easy" (requiring simple tactics) to what many call "devilish" or "diabolical" (because of the need to apply more mind-bending rules of logic).

One way computer scientists think about solving a Sudoku puzzle is to equate the task to a graph-coloring prob-

THE AUTHOR

JEAN-PAUL DELAHAYE is professor of computer science at the University of Sciences and Technologies of Lille in France and a researcher in the Computer Science Laboratory of Lille (LIFL) of the National Center for Scientific Research [CNRS] based at the university. His work focuses on computational game theory (such as the iterated prisoner's dilemma) and complexity theory (such as Kolmogorov complexity) and applications of these theories to genetic analysis and, recently, to economics. This article expands on one Delahaye published in the December 2005 *Pour la Science*, the French edition of *Scientific American*.

JEN CHRISTIANSEN; SOURCE: JEAN-PAUL DELAHAYE

METHOD 1

“FORCED” CELL

This approach considers a cell to be fixed. By eliminating as possibilities any other numbers in the same column, row and subgrid, you see whether only one possibility remains. Such analysis of grid *a* will reveal that the boxes containing orange numbers in grid *b* are “forced” cells.

METHOD 2

“ONLY” CELL

Here a given value is the focus—for example, the number 5. Columns one and three in grid *a* already have 5s, but column two so far has none. Where can that column’s 5 be? Not in the three first cells of column two, because they sit in a subgrid that already has a 5. Not in the seventh cell of the column, because its subgrid, too, has a 5. Thus, the 5 of column two is either in the fourth, fifth or sixth cell of the column. Only the fifth cell is free, so the number goes there. The cells marked with blue numbers in grid *b* are “only” cells.

METHOD 3

SIMPLIFYING THE RANGE OF POSSIBILITIES

This technique is extremely powerful but requires a pencil and eraser. In each cell, you write all possible solutions very small or use dots whose positions represent the numbers 1 to 9. Then you apply logic to try to eliminate options.

For example, grid *c* shows how grid *a* would look if it were marked up by rote, without methods 1 and 2 being applied first. In the third column, the array of possibilities for the second, third, fourth, fifth and sixth cells are, respectively, {2, 3, 6, 7}, {3, 6, 9}, {2, 6}, {2, 6} and {6, 7}. The column must contain a 2 and a 6, so these numbers must be in the two cells whose sole possibilities are 2 and 6 (circled in first detail). Consequently, 2 and 6 cannot be anywhere else in this column and can be deleted from the column’s other cells (red). The range of possibilities for the column is simplified to {3, 7}, {3, 9}, {2, 6}, {2, 6}, {7}. But that isn’t all. Stipulating the position of 7 in turn dictates the positions of 3 and of 9 (second detail). The final possibilities are {3}, {9}, {2, 6}, {2, 6}, {7}. A single uncertainty remains: where 2 and 6 should go.

The general rule for simplifying possibilities is the following: if, among a set of possibilities (for a row, column or subgrid), you find *m* cells that contain a subset consisting only of *m* numbers (but not necessarily all of them in each cell), the digits in the subset can be eliminated as possibilities from the other cells in the larger set. For instance, in *d* the arrangement {2, 3}, {1, 3}, {1, 2}, {1, 2, 4, 5}, {3, 5, 7} can be simplified to {2, 3}, {1, 3}, {1, 2}, {4, 5}, {5, 7}, because the cells {2, 3}, {1, 3}, {1, 2} all come from the subset {1, 2, 3} and have no other numbers.

METHOD 4

TRIAL AND ERROR

By applying methods 1 through 3, you can solve many Sudoku grids. But diabolical-level Sudokus often require a phase of trial and error. When uncertainty persists, you make a random choice and apply all your strategies as if that were the correct decision. If you hit an impossibility (such as two identical numbers in the same column), you know you chose incorrectly. For instance, you might try 2 in the fourth cell of the third column in grid *c*. If that fails, you begin again from the same starting point, but this time with 6 in the box.

Unfortunately, sometimes you have to do several rounds of trial and error, and you have to be prepared to backtrack if you guess incorrectly. Indeed, the idea behind the trial-and-error method is the same as that used by backtracking algorithms, which computer programs can easily implement but which can sorely tax human brains. How remarkable it is that the method most effective for a machine is the least effective for a human being.

lem in which two adjacent cells (otherwise known as “two vertices joined by an edge”) cannot take the same color and the available palette has nine colors. The graph contains 81 vertices, some of which are colored at the outset. The coloring problem is actually quite complex because each 9×9 grid has hundreds of edges. Each cell is part of a row including eight other cells, a column including eight other cells, and a subgrid including eight other cells (of which four have already been counted in the column or row). So each of the 81 cells is linked to 20 ($8 + 8 + 4$) other cells, which makes a grand total of 1,620 cells that share one edge with a neighbor—which, in turn, means that the total number of edges is 810 (1,620 divided by 2).

That Sudoku puzzles can be trans-

lated into a coloring problem is meaningful to scientists, because that property links Sudoku to a class of important problems. In particular, Takayuki Yato and Takahiro Seta of the University of Tokyo have recently demonstrated that Sudoku belongs to the category of NP-complete problems. Such problems are ones that probably cannot be solved in a realistic time frame. Well-known examples include the 3-colorability problem, which asks whether it is possible to shade each node in a graph with three colors in such a way that no

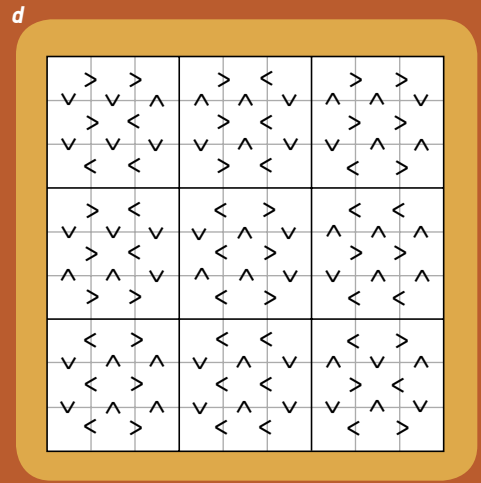
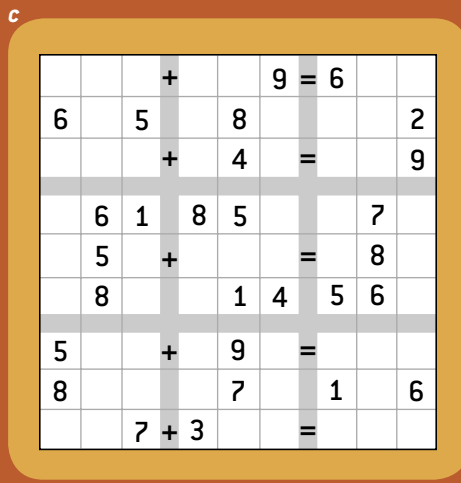
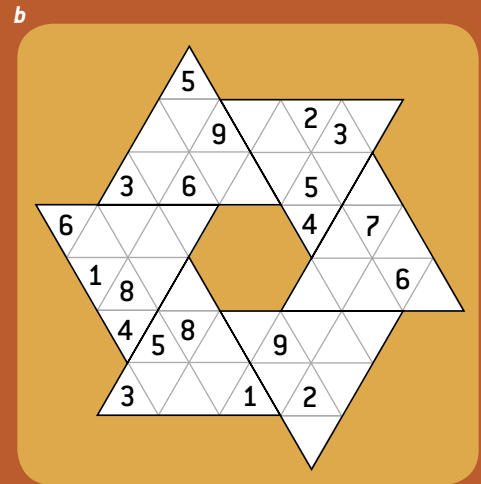
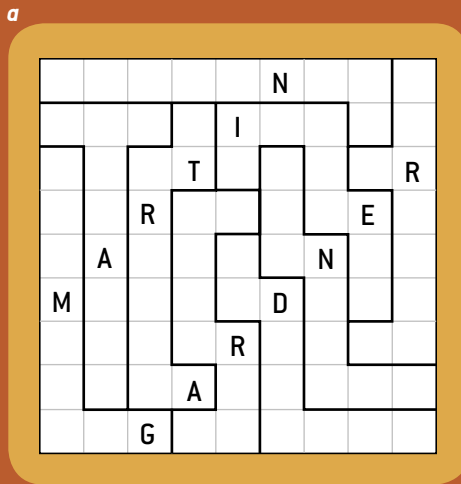
two nodes joined by an edge are assigned the same color. In the case of Sudoku, the apparently impossible challenge is designing an efficient program that would solve Sudokus of all sizes—that is, every grid that takes the form $n^2 \times n^2$, not just the standard $3^2 \times 3^2$ (9×9) versions. No program for solving all puzzles would work efficiently because the time required to find a solution increases dramatically as *n* gets bigger.

If you have an algorithm that solves classic Sudokus, you can use it to obtain an algorithm that designs them. Indeed,

That Sudoku puzzles can be translated into a coloring problem links the game to a class of important mathematical problems.

VARIATIONS ON A THEME

In need of something more than standard diabolical grids? In the puzzles here, the usual rules apply, with some twists. In *a*, the letters in the words GRAND TIME replace numbers, and geometric shapes replace the square subgrids. Its inventor calls it a Du-Sum-Oh puzzle. In *b*, which contains six triangular subgrids, the rows and the slanted columns may be interrupted in the center, and when a row or column has only eight cells, the nearby cell that forms a point of the “star” serves as the ninth cell. In *c*, the three-place numbers formed by the marked rows in the first two subgrids add up to the number in the third subgrid. In *d*, greater-than and less-than signs indicate where the digits belong. In *e*, the dominoes at the bottom need to be placed into the empty spaces. In *f*, three game boards overlap. Visit www.sciam.com for solutions and more games.



although early Sudokus were constructed by hand, today almost all are produced by computer programs based on the following approach or a similar one. Numbers are placed at random on a grid board, and a solution algorithm (for ex-

ample, backtracking) is applied. If the puzzle possesses a unique solution, the program stops. If the partially completed problem has no solution, one number is taken away from the starting arrangement and the program begins again. If

the puzzle has various solutions, one is chosen and the algorithm then adds as many numbers as needed to the starting clues to ensure that the chosen solution is unique.

Human Strategies

FANS WHO ENJOY solving Sudokus manually can choose among many tactics, but two basic approaches offer a decent starting point. First, search for the most constrained empty cells: those that belong to a row, column or subgrid that is already pretty well filled in. Sometimes eliminating impossibilities (the numbers already occupying cells in the same row, column or grid) will lead you to discover the only number that will work in a particular cell; in any case, the method should greatly narrow the options.

TOO FEW CLUES

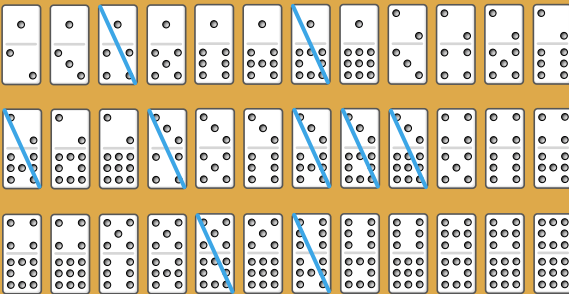
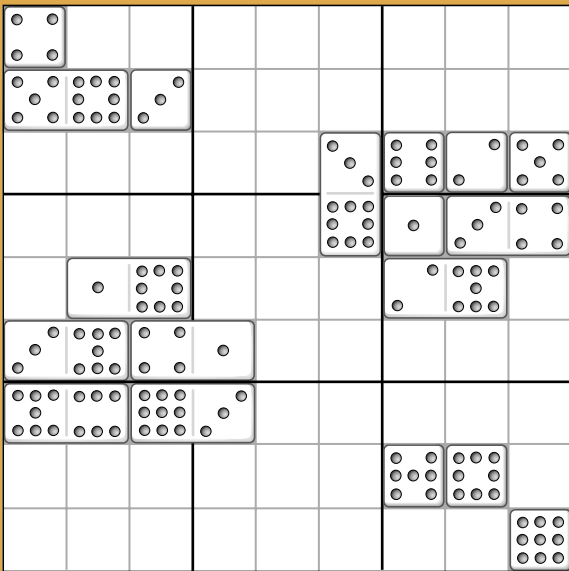
77 clues are not necessarily enough to guarantee a unique solution. Despite having only four empty cells, the grid here has two solutions: in the first two columns the missing 1's and 2's (*inset*) are interchangeable.

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

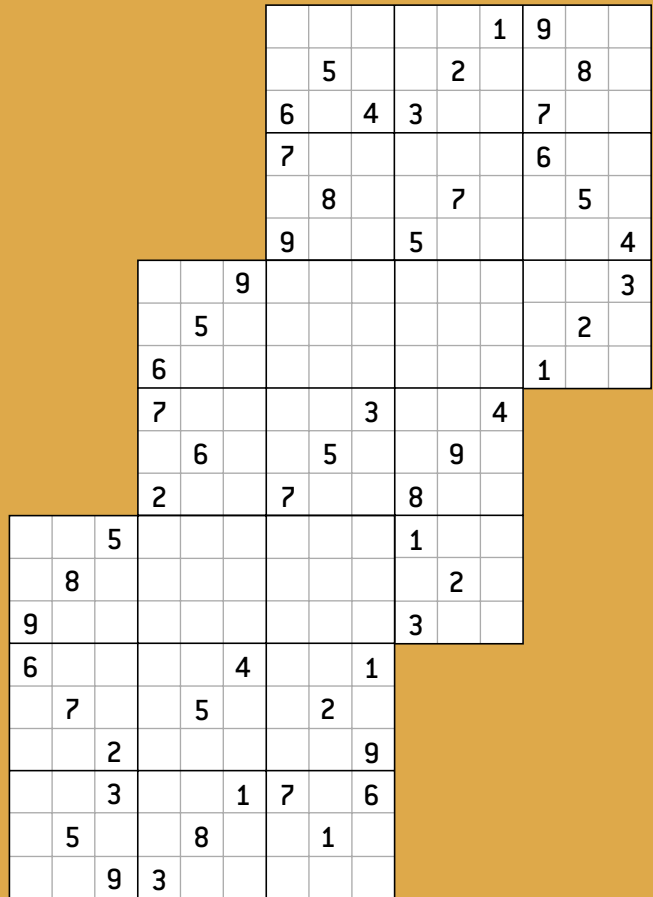
		3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6
		4	3	9	8	5	6	7
8	6	5	2	7	1	3	9	4
9	3	7	6	4	5	8	1	2
3	4	1	8	6	2	9	7	5
5	7	2	9	1	4	6	3	8
6	9	8	5	3	7	2	4	1

JEN CHRISTIANSEN; SOURCES: BOB HARRIS IN THE GRAND TIME SUDOKU AND THE LAW OF LEFTOVERS; A.K. PETERS LTD (IN PRESS) (a); © NONZERO/OLGA LEONTIEVA (b); ED PEGG, JR. (c); © NONZERO/CHAN ALTAY (d)

e



f



Second, search for where a given value might be found in a particular column, row or subgrid (for example, look for the only places where a 3 might fit in row four). Sometimes the query will have only one possible response. Other times just knowing that the 3 can go only in two or three particular spots ends up being helpful. See the box on pages 84 and 85 for more details. Also, visit the Web sites listed in “More to Explore” to find a host of strategies, some of which have such creative names as “swordfish” and “golden chain.”

A number of software programs easily found on the Internet will generate boards of specified difficulty and help you find solutions (without, of course, solving the puzzle for you!). For example, some allow you to put temporary marks in the cells and to erase them,

thus making pencil and eraser unnecessary. Some even enable you to create links between cells. Do not overlook these software programs. In freeing you from such tedious tasks as erasing, they will actually spur you to greater subtlety and virtuosity in this game of logic.

Once you have gotten bored with traditional Sudokus, you can go looking for the innumerable variants: some overlap multiple grids; others replace

square subgrids with other structures; still others introduce additional constraints. These alternatives appeal because they compel you to explore new logical strategies. Moreover, devotees who take only a quarter of an hour to do a traditional puzzle can immerse themselves the entire day in the delights of combining cells and numbers in giant versions of Sudoku. But enough of that. On to the next grid!

MORE TO EXPLORE

1st World Sudoku Championship: www.wsc2006.com/eng/index.php

Math Games. Ed Pegg, Jr.: www.maa.org/editorial/mathgames/mathgames_09_05_05.html

The Mathematics of Su Doku. Sourendu Gupta: <http://theory.tifr.res.in/~sgupta/sudoku/>

Mathematics of Sudoku. Tom Davis: www.geometer.org/mathcircles

SadMan Software Sudoku techniques: www.simes.clara.co.uk/programs/sudokutechniques.htm

Sudoku, an overview: www.sudoku.com/howtosolve.htm

Sudoku, from Wikipedia: <http://en.wikipedia.org/wiki/Sudoku>

A Variety of Sudoku Variants: www.sudoku.com/forums/viewtopic.php?t=995