

Sudoku Puzzle



Comparing Different Solution
Algorithms

Luz Therese C. Duñgo

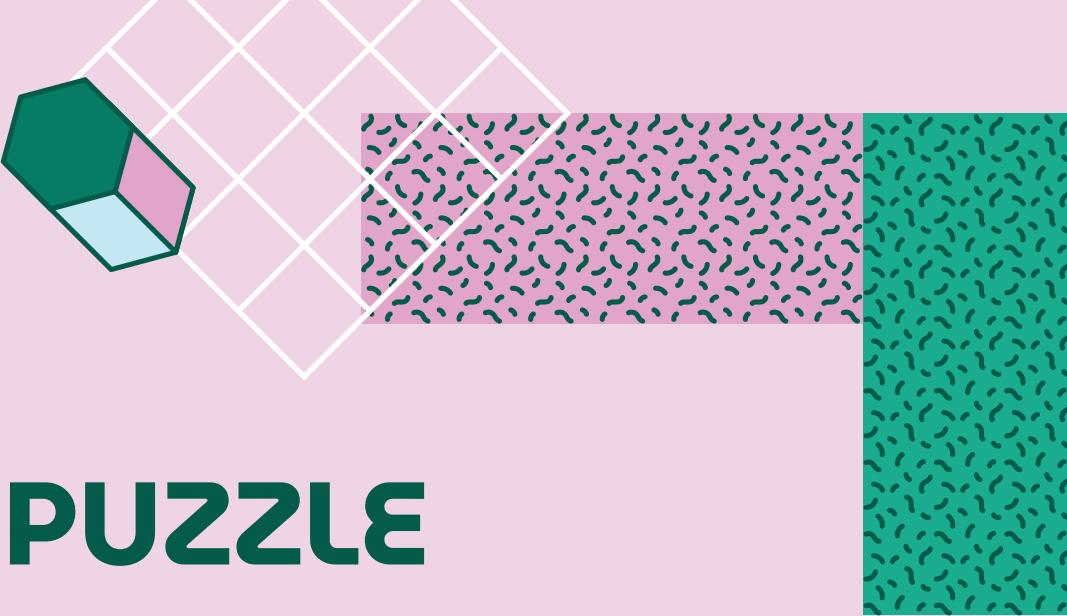




Background | What is a Sudoku Puzzle?
How does it work?

SUDOKU PUZZLE

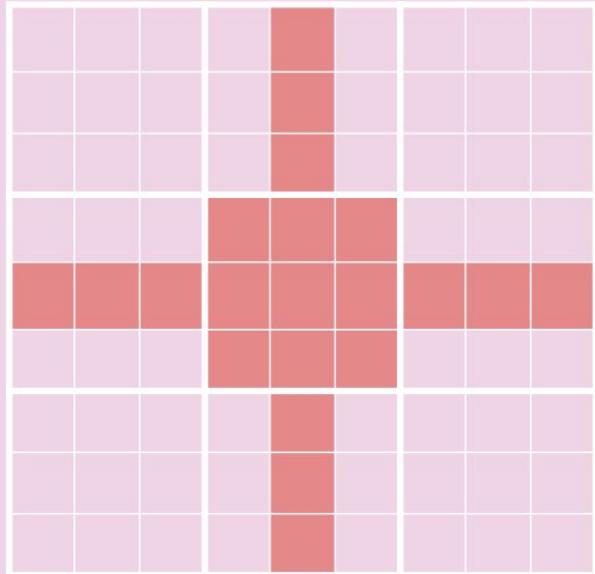
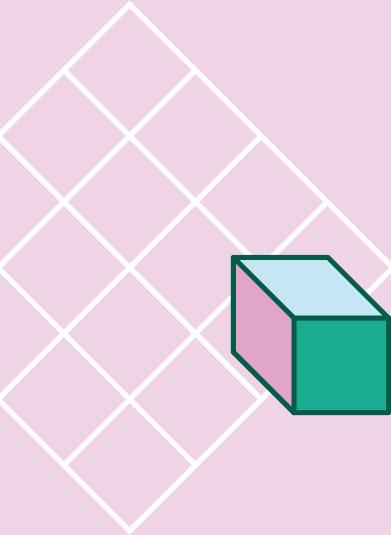
Given a **partially filled in 9x9 grid**, assign digits from 1 to 9 to the empty cells such that every row, column, and 3x3 sub-grid contains exactly one instance of the digits.

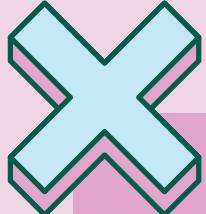


SUDOKU PUZZLE RULES



- Use a number **only once** in a **row**
- Use a number **only once** in a **column**
- Use a number **only once** in a **3x3 square**

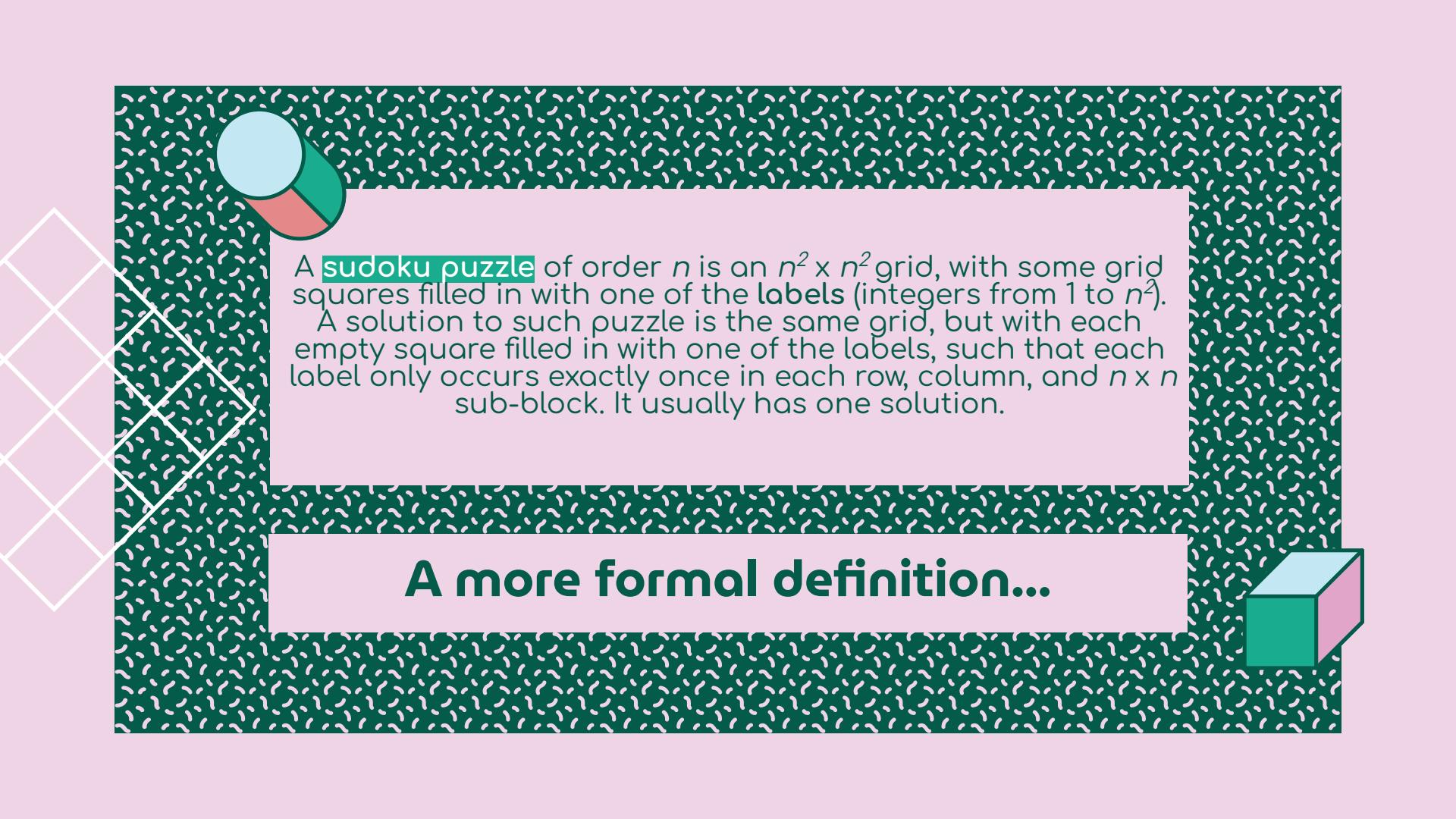




8								
		3	6					
	7			9		2		
	5				7			
				4	5	7		
		1					3	
	1						6	
	8	5					1	
9					4			



SUDOKU PUZZLE EXAMPLE



A **sudoku puzzle** of order n is an $n^2 \times n^2$ grid, with some grid squares filled in with one of the labels (integers from 1 to n^2).

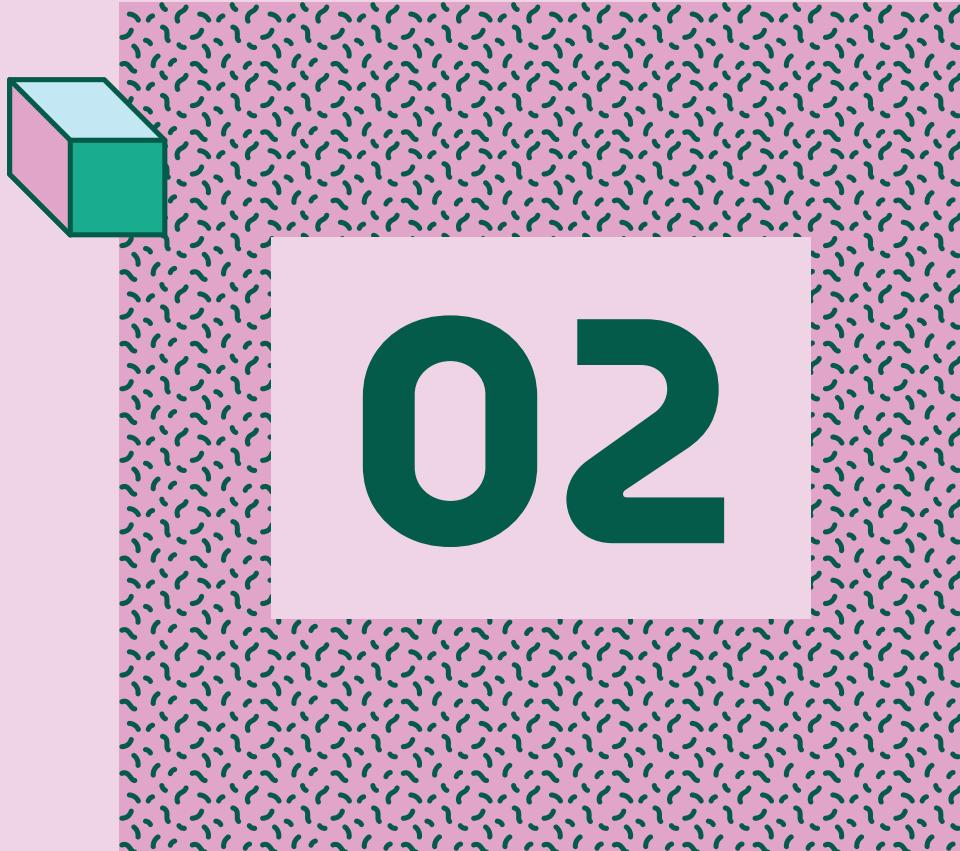
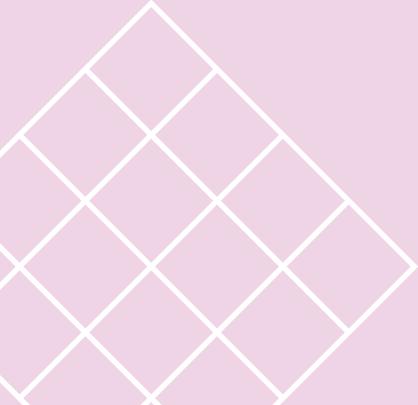
A solution to such puzzle is the same grid, but with each empty square filled in with one of the labels, such that each label only occurs exactly once in each row, column, and $n \times n$ sub-block. It usually has one solution.



A more formal definition...

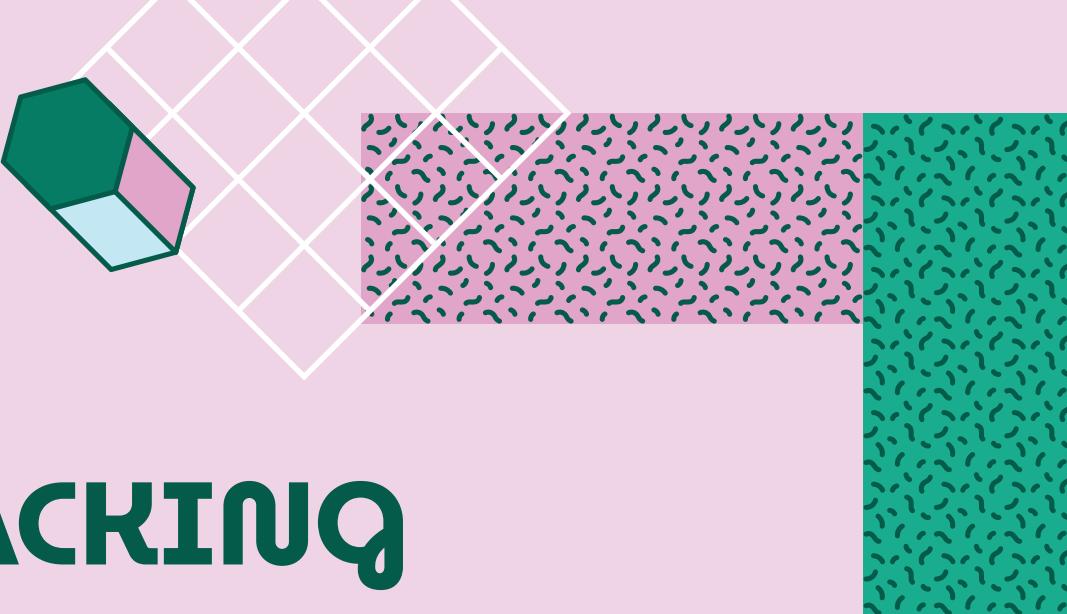
Backtracking

What is this? How is this
used to solve the puzzle?



BACKTRACKING

A problem-solving algorithm that uses brute-force and recursion wherein if the current solution is not suitable, it **goes back to a previous solution** and tries other solutions.



Backtracking in Sudoku



How does it work?

Find first empty square (starting from top left) then choose the first candidate

- (no contradiction) move on to the next square
- (w/ contradiction) choose next candidate
- (no more candidates) go back to previous square and fill in the next candidate

Repeat steps until the bottom right cell is reached

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



Stochastic Search

What is this? How is this
used to solve the puzzle?





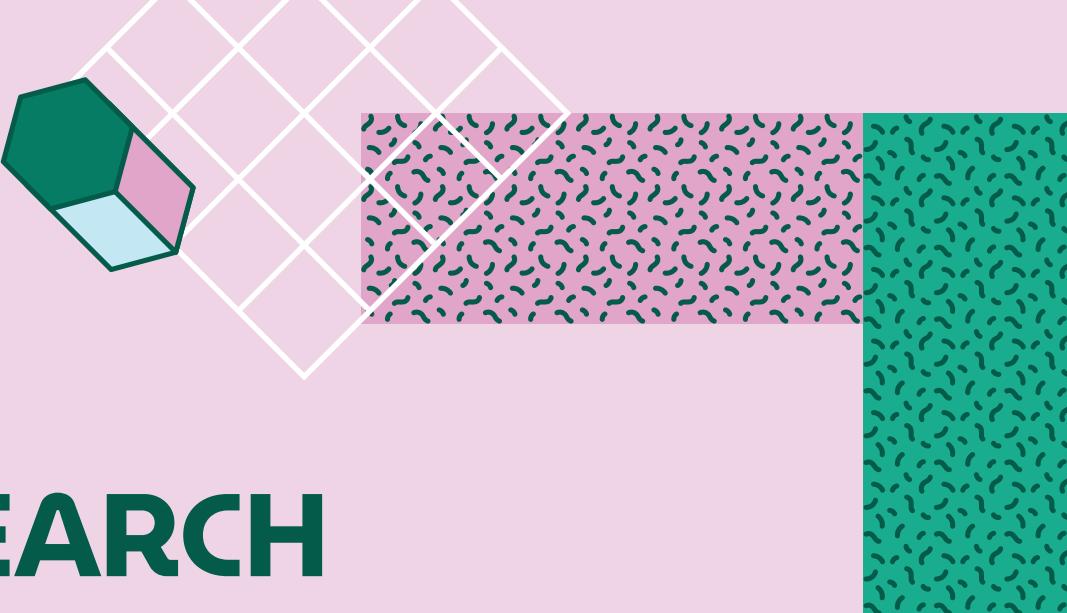
STOCHASTIC SEARCH

An optimization algorithm that **incorporates randomness** in its exploration of the search space.



BEAM SEARCH

A heuristic search algorithm that examines a graph by extending the most promising node at each level. It progresses level by level and moves downwards only from the best nodes at each level.



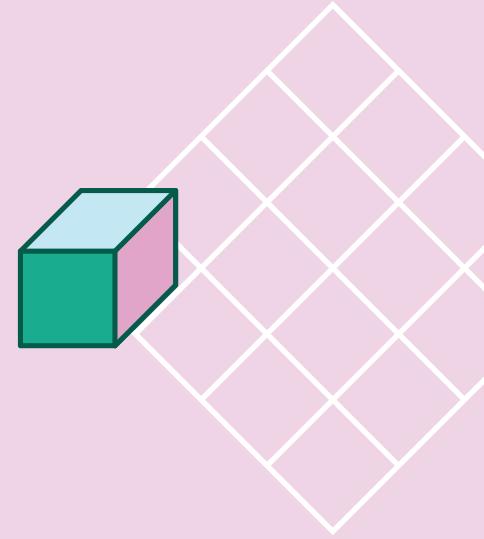
Stochastic Search in Sudoku



How does it work?

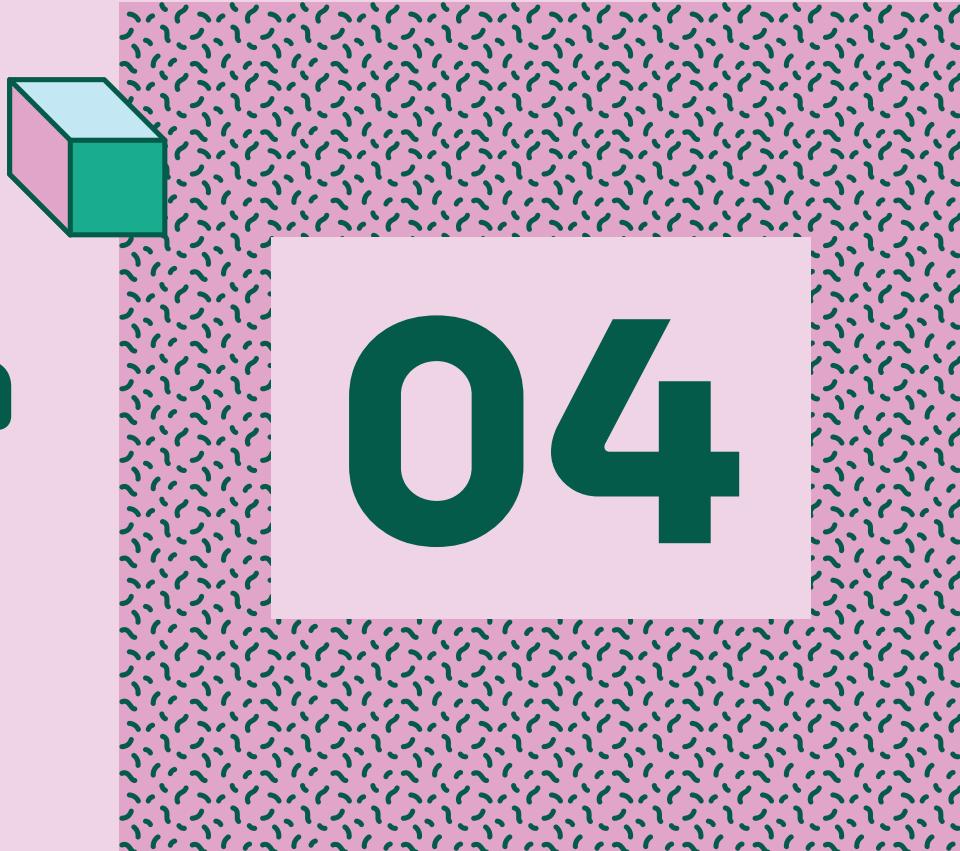
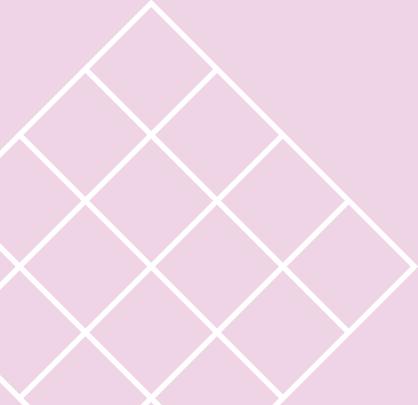
Create a series of 10 candidate boards by randomly filling in the empty squares in the puzzle then check how many mistakes each board has

- (no mistakes) that board is the solution
- (w/ mistakes) a set of 4 successors is generated from it by swapping labels of two cells in the same row
- Successors are added to candidate boards which is sorted by number of mistakes (least mistakes taken)



Constraint Programming

What is this? How is this
used to solve the puzzle?





CONSTRAINT PROGRAMMING

A paradigm that **identifies feasible solutions** out of a set of candidates where the problem can be modeled in terms of arbitrary constraints.

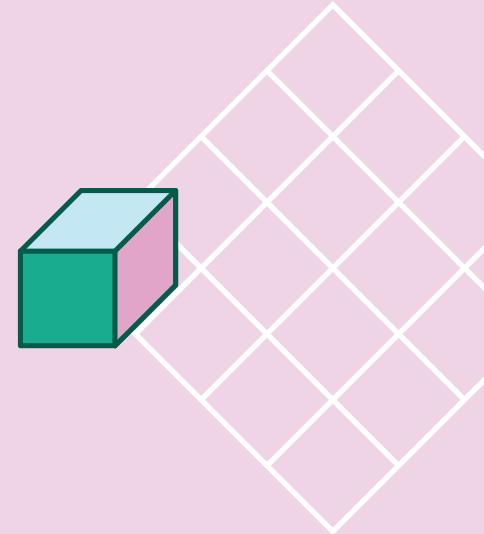
Constraint Programming in Sudoku

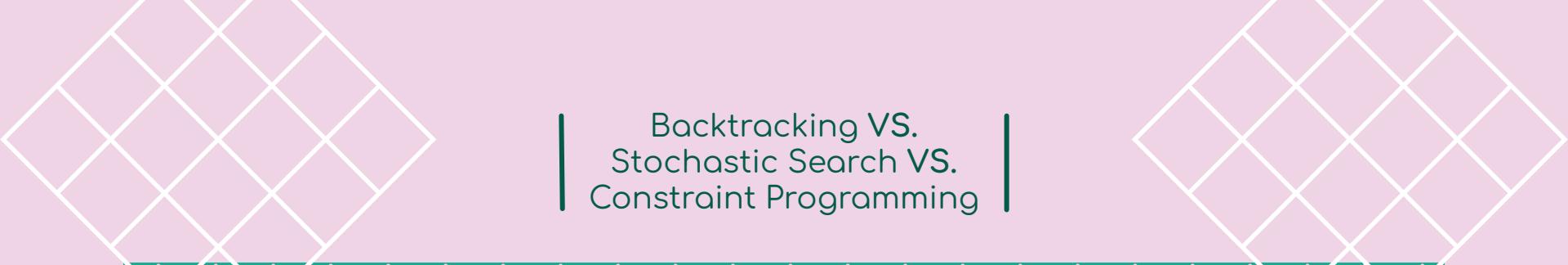


How does it work?

Sudoku rules are written as **constraints** (rules a possible solution must follow) and these constraints are combined with the backtracking algorithm

- Check the constraint every time a value in that row changed or if the search space is reduced



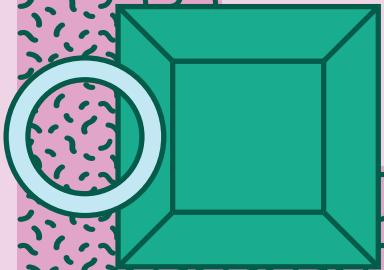


Backtracking VS. Stochastic Search VS. Constraint Programming

05

Demo & Comparison

Demo



Comparing the Algorithms



	Backtracking	Stochastic Search	Constraint Programming
Time Complexity	$O(N^{(n*n)})$	$O(b * (N^{(n*n)}))$	$O(N^{(n*n)})$
Method	Depth-first search	Breadth-first search	Depth-first search
# of solutions (if possible)	1	1 or none	At least 1

Comparing the Algorithms



	Backtracking	Stochastic Search	Constraint Programming
Advantage/s	<ul style="list-style-type: none">• Faster than constraint programming• Would always find a solution if there is one	<ul style="list-style-type: none">• Possibility of finding a solution right away	<ul style="list-style-type: none">• Finds all possible solutions

Comparing the Algorithms



	Backtracking	Stochastic Search	Constraint Programming
Disadvantage/s	<ul style="list-style-type: none">May spend a long time assuming a value which is wrong	<ul style="list-style-type: none">Not optimalUnreliableProblematic if puzzle has more than one solutionMore possible successors, less chances of picking the right one	<ul style="list-style-type: none">Higher run time than backtracking (if multiple solutions)

Thank you!

Does anyone have any
questions/comments/reactions?

Luz Therese C. Duñgo
CMSC 142 | Section G

CREDITS: This presentation template was created
by Slidesgo, including icons by Flaticon, and
infographics & images by Freepik

References



- Computerphile. (2020, February 13). *Python Sudoku Solver - Computerphile* [Video]. YouTube. https://www.youtube.com/watch?v=G_UYXzGuqvM
- Hoexum, E. (2020). Revisiting the proof of the complexity of the sudoku puzzle (Doctoral dissertation). Retrieved from https://fse.studenttheses.ub.rug.nl/22745/1/bMATH_2020_HoexumES.pdf.pdf
- GeeksForGeeks. (2017, July 16). *Sudoku (Explanation) | Backtracking | Set 7* / GeeksforGeeks [Video]. YouTube. <https://www.youtube.com/watch?v=l7f9-GNH1j8>
- Backtracking algorithm. Programiz. (n.d.). Retrieved from <https://www.programiz.com/dsa/backtracking-algorithm>.
- Neumann, F., & Witt, C. (2010). Stochastic Search Algorithms. Natural Computing Series, 21–32. doi:10.1007/978-3-642-16544-3_3

References



- *What is stochastic search.* IGI Global. (n.d.). Retrieved from <https://www.igi-global.com/dictionary/stationary-density-stochastic-search-processes/28313>.
- *Introduction to beam search algorithm.* GeeksforGeeks. (2021, July 18). Retrieved from <https://www.geeksforgeeks.org/introduction-to-beam-search-algorithm/>.
- Google. (n.d.). *Constraint optimization.* Google. Retrieved from <https://developers.google.com/optimization/cp>.
- *Sudoku puzzles, constraint programming and graph theory.* OpenSourc.ES. (n.d.). Retrieved from <https://opensourc.es/blog/sudoku/>.
- *Sudoku: Backtracking-7.* GeeksforGeeks. (2021, July 22). Retrieved from <https://www.geeksforgeeks.org/sudoku-backtracking-7/>.

References



- *Sudoku solver.* AfterAcademy. (n.d.). Retrieved from <https://afteracademy.com/blog/sudoku-solver>.
- *Define Beam Search.* Javatpoint. (n.d.). Retrieved from <https://www.javatpoint.com/define-beam-search>.



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



SUDOKU PUZZLE EXAMPLE