ARTIFICIAL INTELLIGENCE

SOLVING SUDOKU PUZZLE

ARAM ADAMYAN, HOVHANNES HOVHANNISYAN, ARARAT KAZARIAN

Sudoku is a logic-based, combinatorial number-placement puzzle. In classic Sudoku, the objective is to fill a 9 × 9 grid with digits so that each column, each row, and each of the nine 3 × 3 subgrids that compose the grid (also called "boxes", "blocks", or "regions") contain all of the digits from 1 to 9. The puzzle setter provides a partially completed grid, which for a well-posed puzzle has a single solution.

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

SUDOKU VARIATIONS

	6						8	11			15	14			16
15	11				16	14				12			6		
13		9	12					3	16	14		15	11	10	
2		16		11		15	10	1							
	15	11	10			16	2	13	8	9	12				
12	13			4	1	5	6	2	3					11	10
5		6	1	12		9		15	11	10	7	16			3
	2				10		11	6		5			13		9
10	7	15	11	16				12	13						6
9						1			2		16	10			11
1		4	6	9	13			7		11		3	16		
16	14			7		10	15	4	6	1				13	8
11	10		15				16	9	12	13			1	5	4
		12		1	4	6		16				11	10		
		5		8	12	13		10			11	2			14
3	16			10			7			6				12	

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

			3
	4		
		3	2

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

• BACKTRACKING SEARCH

A VARIANT OF DEPTH-FIRST SEARCH CALLED BACKTRACKING SEARCH USES EVEN LESS MEMORY. IN BACKTRACKING, ONLY ONE SUCCESSOR IS GENERATED AT A TIME RATHER THAN ALL SUCCESSORS; EACH PARTIALLY EXPANDED NODE REMEMBERS WHICH SUCCESSOR TO GENERATE NEXT. IN ADDITION, SUCCESSORS ARE GENERATED BY MODIFYING THE CURRENT STATE DESCRIPTION DIRECTLY RATHER THAN ALLOCATING MEMORY FOR A BRAND-NEW STATE. THIS REDUCES THE MEMORY REQUIREMENTS TO JUST ONE STATE DESCRIPTION AND A PATH OF O(M) ACTIONS; A SIGNIFICANT SAVINGS OVER O(BM) STATES FOR DEPTH-FIRST SEARCH. WITH BACKTRACKING WE ALSO HAVE THE OPTION OF MAINTAINING AN EFFICIENT SET DATA STRUCTURE FOR THE STATES ON THE CURRENT PATH, ALLOWING US TO CHECK FOR A CYCLIC PATH IN O(1) TIME RATHER THAN O(M). FOR BACKTRACKING TO WORK, WE MUST BE ABLE TO UNDO EACH ACTION WHEN WE BACKTRACK. BACKTRACKING IS CRITICAL TO THE SUCCESS OF MANY PROBLEMS WITH LARGE STATE DESCRIPTIONS, SUCH AS ROBOTIC ASSEMBLY.

• SIMULATED ANNEALING

A HILL-CLIMBING ALGORITHM THAT NEVER MAKES “DOWNHILL” MOVES TOWARD STATES WITH LOWER VALUE (OR HIGHER COST) IS ALWAYS VULNERABLE TO GETTING STUCK IN A LOCAL MAXIMUM. IN CONTRAST, A PURELY RANDOM WALK THAT MOVES TO A SUCCESSOR STATE WITHOUT CONCERN FOR THE VALUE WILL EVENTUALLY STUMBLE UPON THE GLOBAL MAXIMUM, BUT WILL BE EXTREMELY INEFFICIENT. THEREFORE, IT SEEMS REASONABLE TO TRY TO COMBINE HILL CLIMBING WITH A RANDOM WALK IN A WAY THAT YIELDS BOTH EFFICIENCY AND COMPLETENESS. SIMULATED ANNEALING SIMULATED ANNEALING IS SUCH AN ALGORITHM. IN METALLURGY, ANNEALING IS THE PROCESS USED TO TEMPER OR HARDEN METALS AND GLASS BY HEATING THEM TO A HIGH TEMPERATURE AND THEN GRADUALLY COOLING THEM, THUS ALLOWING THE MATERIAL TO REACH A LOW-ENERGY CRYSTALLINE STATE.

• GENETIC ALGORITHM

Chapter 4 Search in Complex Environments

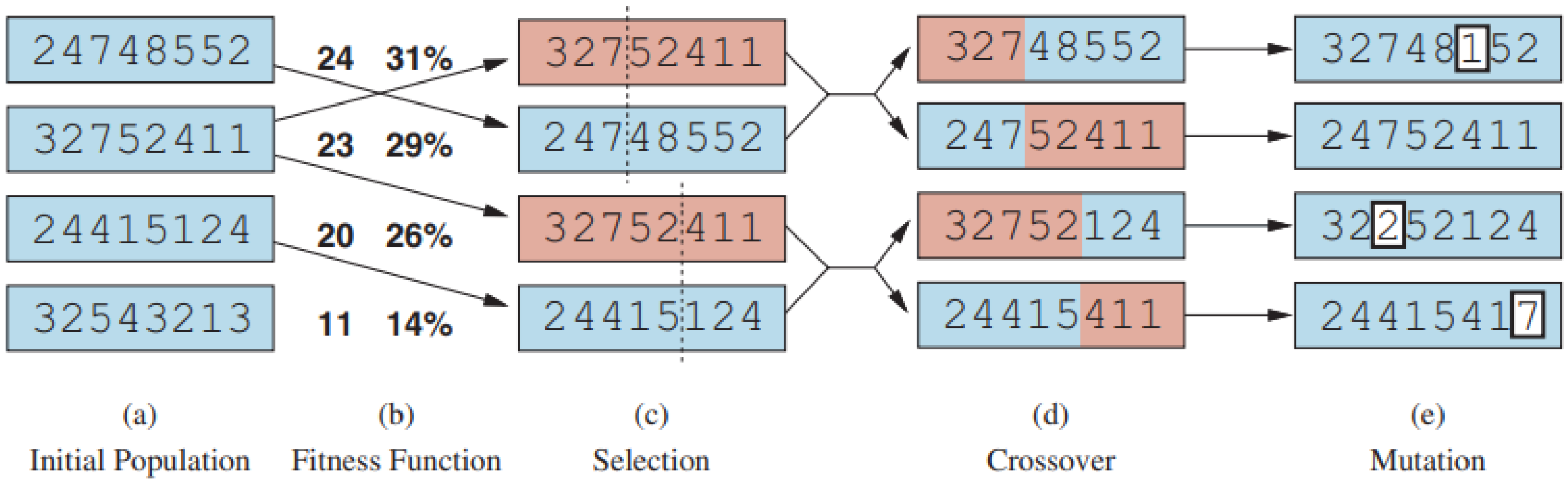
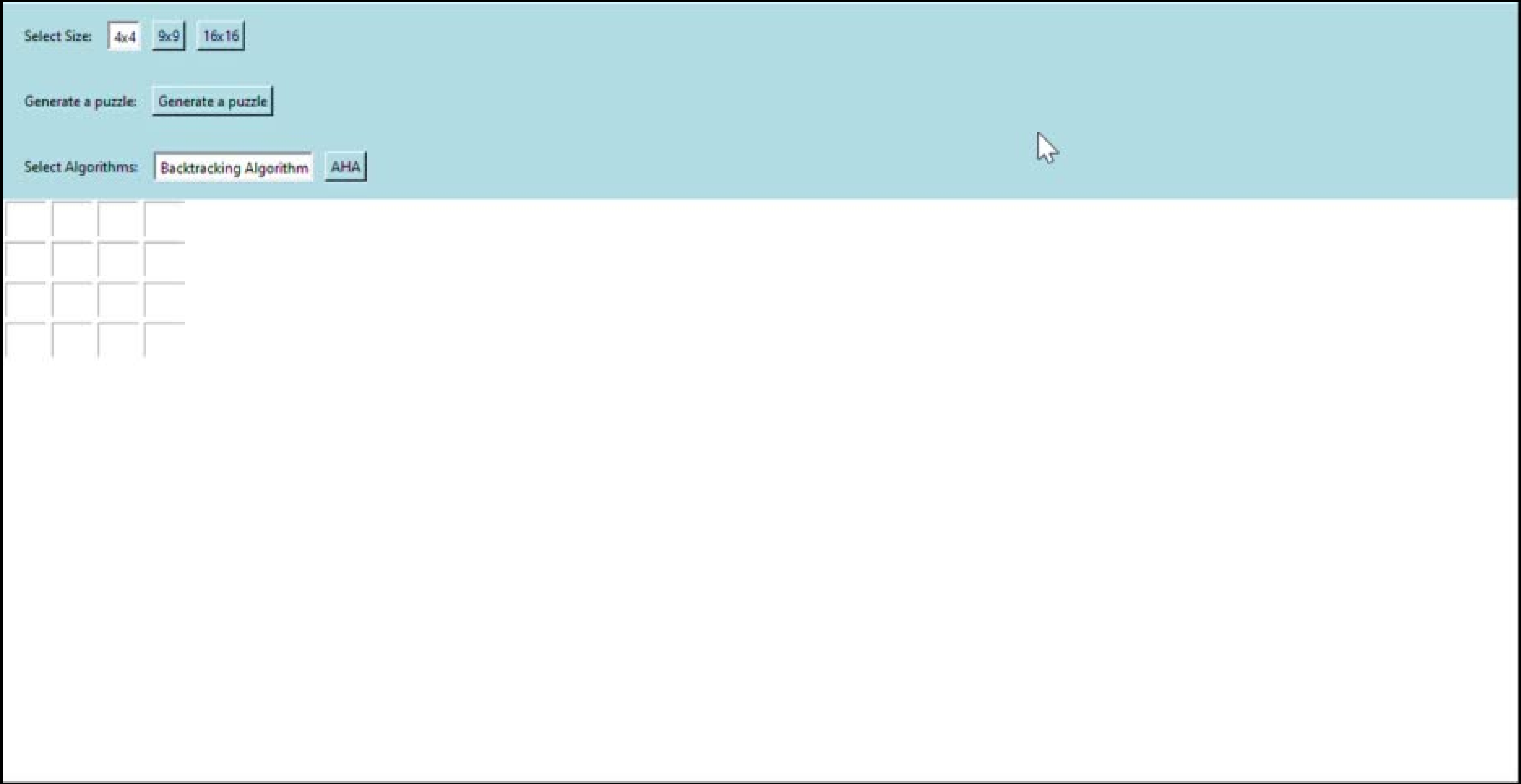
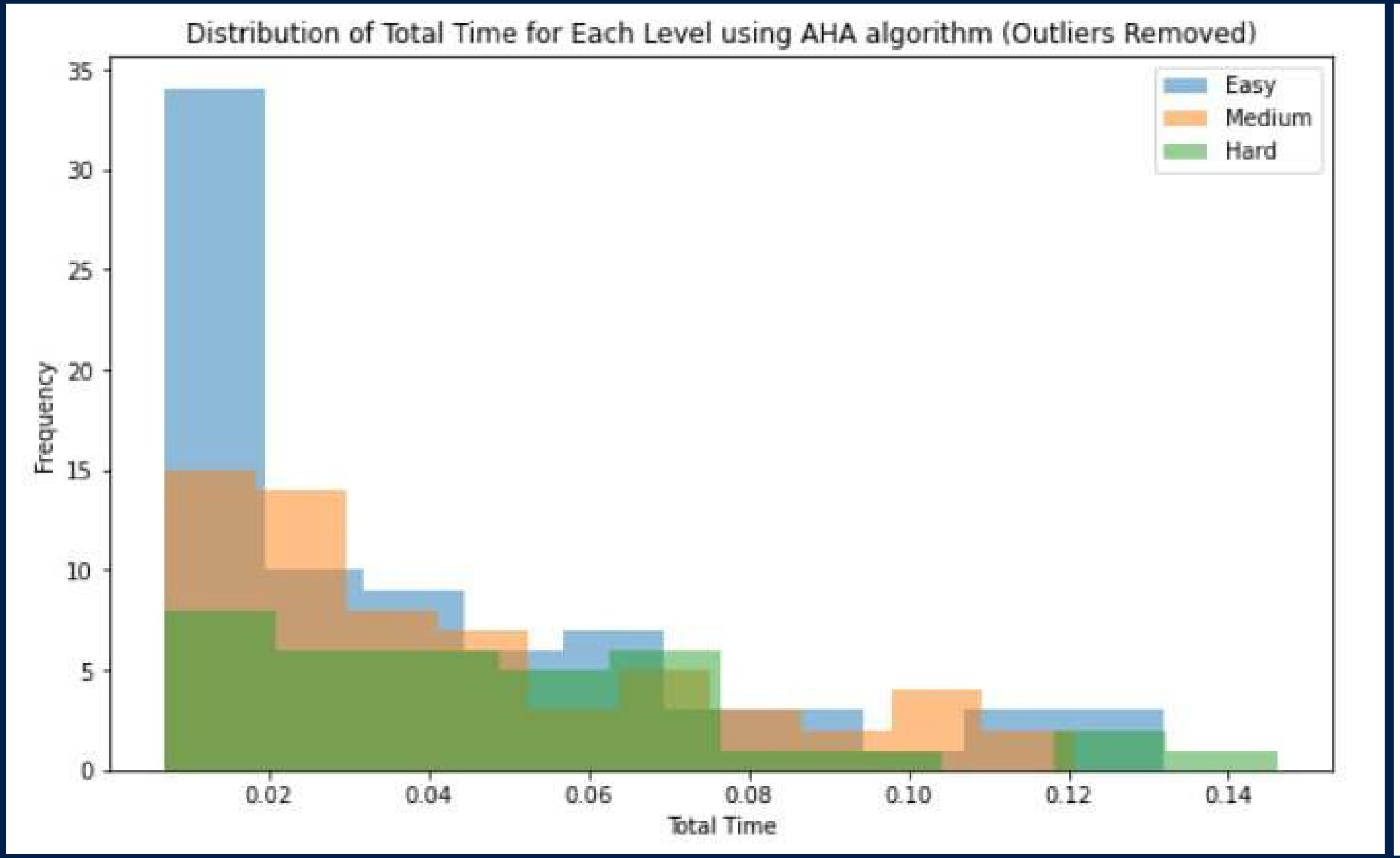
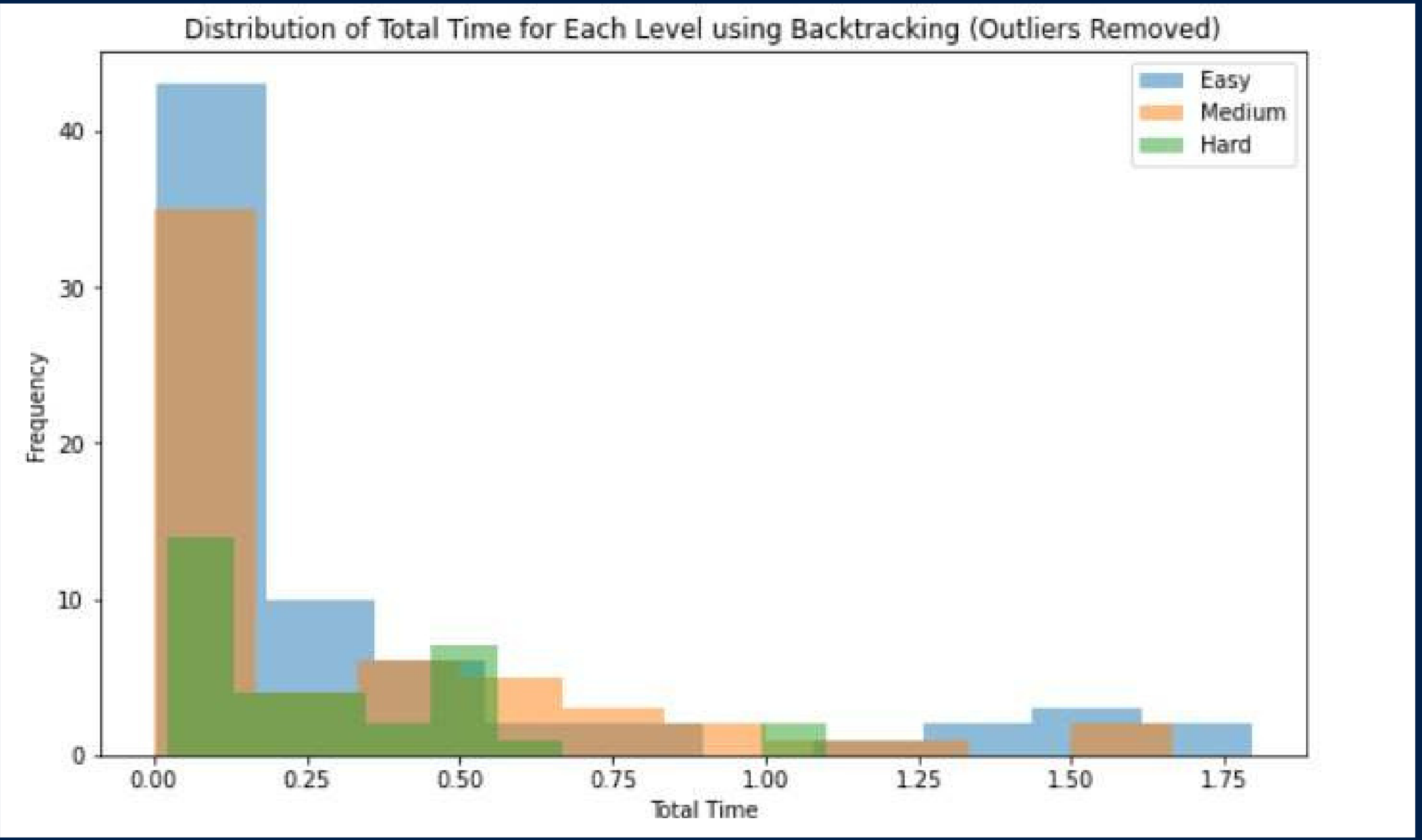
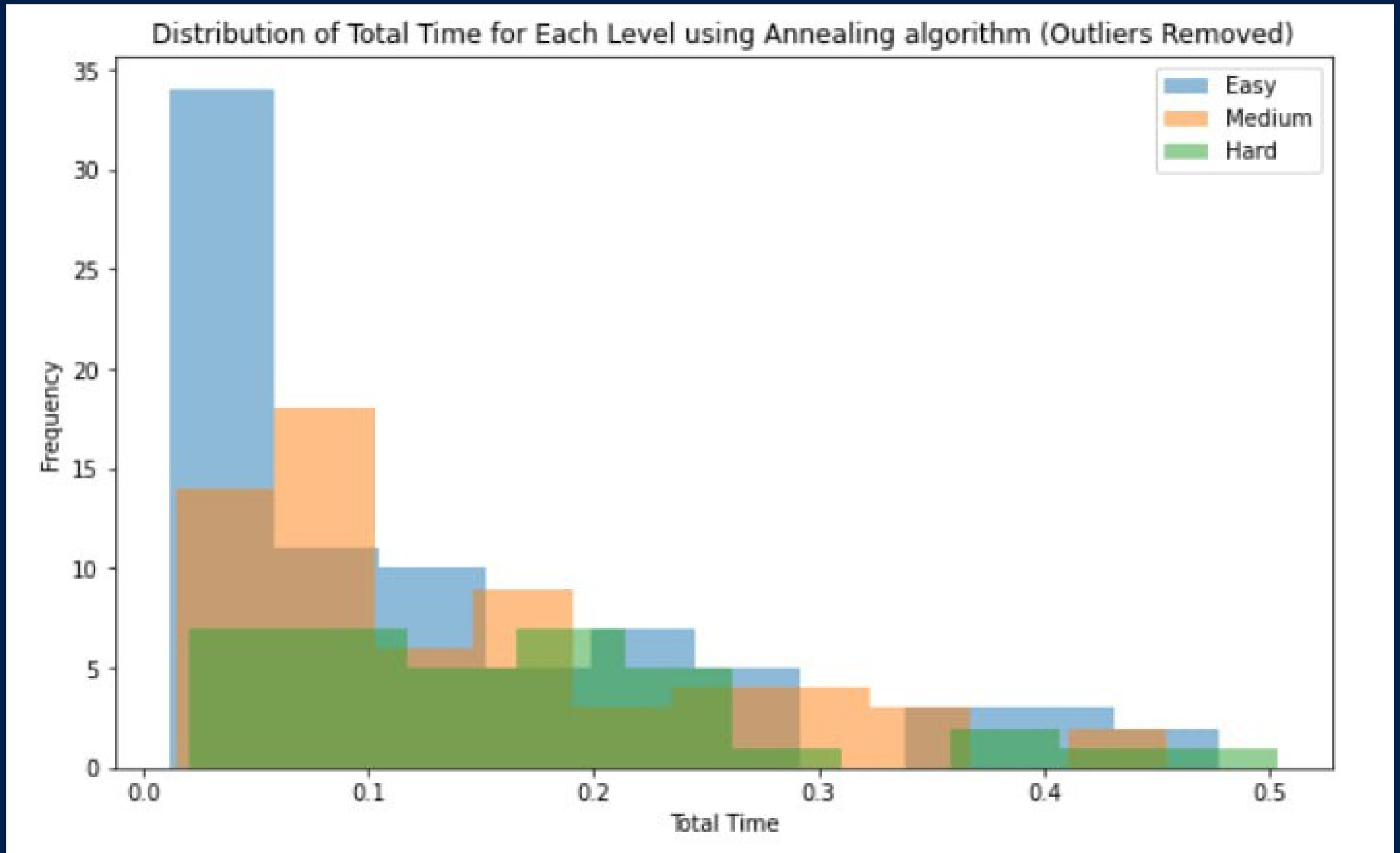


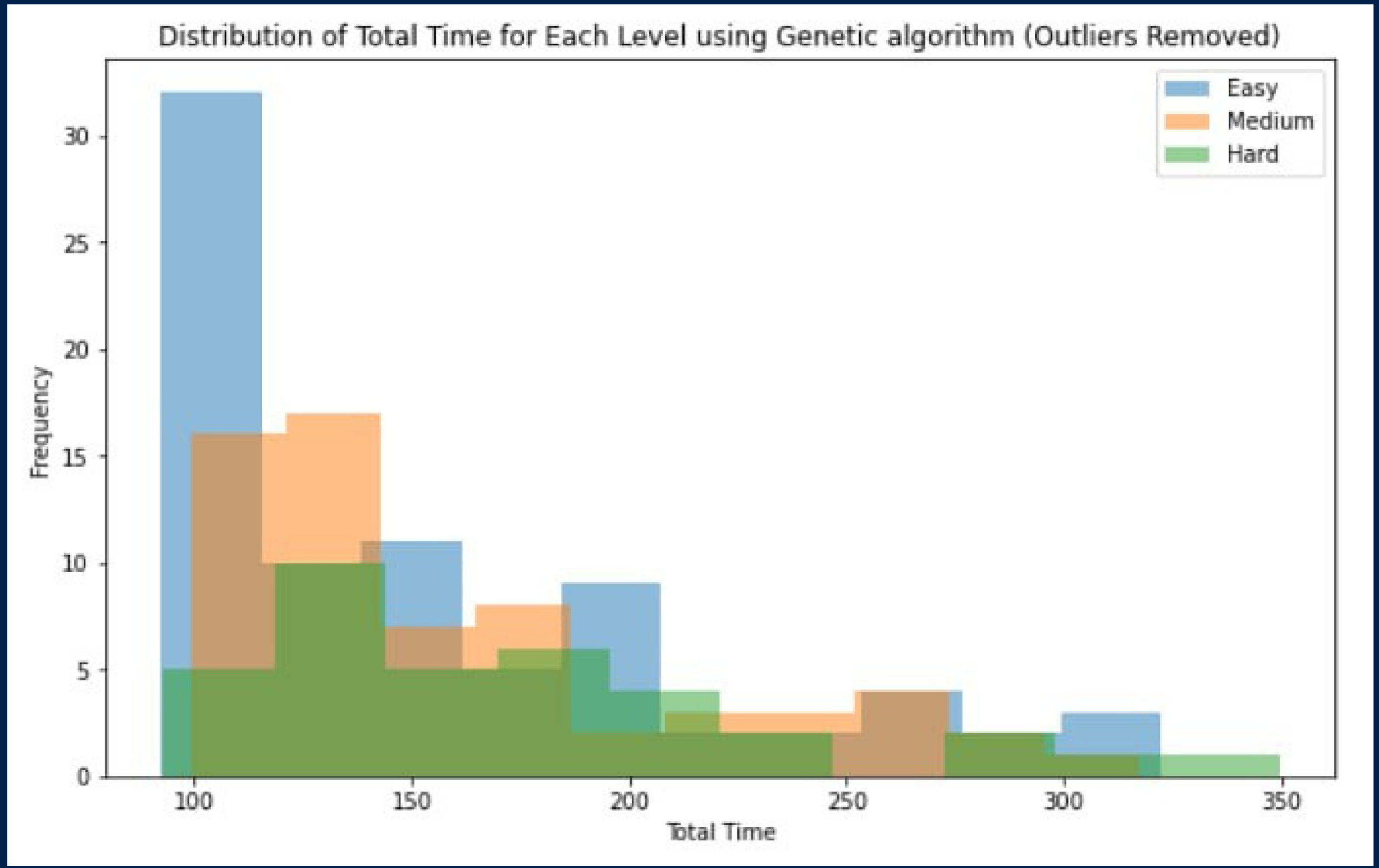
Figure 4.6 A genetic algorithm, illustrated for digit strings representing 8-queens states. The initial population in (a) is ranked by a fitness function in (b) resulting in pairs for mating in (c). They produce offspring in (d), which are subject to mutation in (e).


Select Size:
4x4
9x9
16x16
Generate a puzzle:
Generate a puzzle
Select Algorithms:
Backtracking Algorithm
AHA

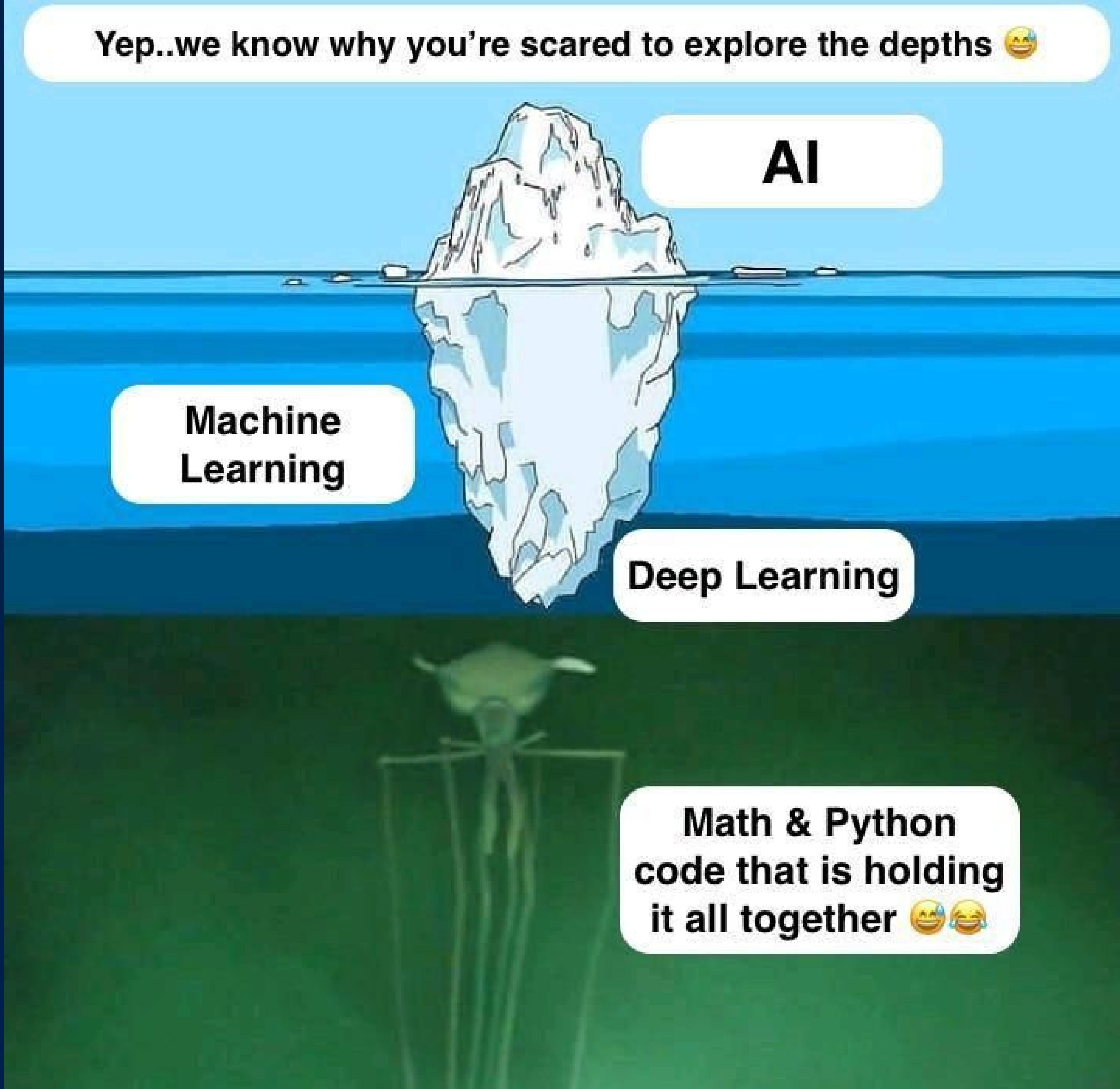

Distribution of Total Time for Each Level using AHA algorithm (Outliers Removed)
Easy
Medium
Hard
Frequency
Total Time
0
5
10
15
20
25
30
35
0.02
0.04
0.06
0.08
0.10
0.12
0.14


Distribution of Total Time for Each Level using Backtracking (Outliers Removed)
Easy
Medium
Hard
Frequency
Total Time
0
10
20
30
40
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75


Distribution of Total Time for Each Level using Annealing algorithm (Outliers Removed)
Easy
Medium
Hard
Frequency
Total Time
0.0
0.1
0.2
0.3
0.4
0.5
0
5
10
15
20
25
30
35


Distribution of Total Time for Each Level using Genetic algorithm (Outliers Removed)
Easy
Medium
Hard
Frequency
Total Time
100
150
200
250
300
350
0
5
10
15
20
25
30

FUTURE WORK


Yep..we know why you're scared to explore the depths 😅
AI
Machine Learning
Deep Learning
Math & Python code that is holding it all together 😅😂

THANK YOU