

# Joshua Pfefferkorn

## Programming Assignment #5

COSC 76, Fall 2021

### Description

#### GSAT

GSAT is a search algorithm used to solve Boolean satisfiability problems. It operates by generating a random assignment, then choosing variables to flip until a satisfying assignment is found. The variable is either chosen randomly or chosen based on the number of clauses flipping it will satisfy – this variable selection depends on some chosen threshold. I used a set to represent the knowledge base, full of inner frozensets, each representing a clause. Inside clauses were integers that represented specific variables, each one either positive or negative (indicating negation). I used a list to store the assignment, indices representing variable numbers and either a 0 (for false) or 1 (for true) in each position.

#### WalkSAT

WalkSAT is quite similar to GSAT, the difference in that instead of considering all variables for flipping, it considers only the variables from a randomly-chosen unsatisfied clause. This greatly increases computation speed, allowing WalkSAT to iterate more quickly and solve more complex puzzles. My implementation used the same data structures as GSAT.

#### Parameters

I chose a threshold of 0.7 (30% change of random variable pick) and a maximum iteration value of 100,000 for both of my algorithms.

### Evaluation

As far as I can tell, my algorithms work as intended. Each produced a visually-verifiable solution for the puzzles that they completed. My algorithms also found solutions in a reasonable number of iterations, comparable to those provided by Prof. Quattrini Li in the Slack channel. Furthermore, logging elapsed time confirmed that WalkSAT computed more quickly than GSAT, as anticipated. Results for both algorithms are included below.

#### GSAT Results

`one_cell.cnf`

Iterations: 3

Time elapsed: 0.0005171298980712891

```

9 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0

```

```

-----
0 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0

```

```

-----
0 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0
0 0 0 | 0 0 0 | 0 0 0

```

all\_cells.cnf

Iterations: 525

Time elapsed: 249.82505798339844

```

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

```

```

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

```

```

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

```

rows.cnf

Iterations: 621

Time elapsed: 297.6539409160614

```

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

```

```

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

```

```

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

```

## WalkSAT Results

one\_cell.cnf

Iterations: 3

Time elapsed: 0.00029587745666503906

Solution:

4	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0

0	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0

0	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0
0	0	0		0	0	0		0	0	0

all\_cells.cnf

Iterations: 623

Time elapsed: 1.5903599262237549

Solution:

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

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

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

rows.cnf

Iterations: 767

Time elapsed: 2.209458112716675

Solution:

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

```

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

```

```

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

```

rows\_and\_cols.cnf

Iterations: 3303

Time elapsed: 13.783544063568115

Solution:

```

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

```

```

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

```

```

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

```

rules.cnf

Iterations: 7207

Time elapsed: 34.6796338558197

Solution:

```

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

```

```

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

```

```

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

```

puzzle1.cnf

Iterations: 29987

Time elapsed: 119.70454907417297

Solution:

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

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

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

puzzle2.cnf

Iterations: 17673

Time elapsed: 80.26382207870483

Solution:

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

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

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