

Metaheuristics Lab1

Ricardo Rosa

A62461

Problem 4:

Run your algorithm on the hoos.cnf instance, and list all solutions that satisfy the problem instance.

The hoos.cnf problem instance consists of:

$$F = (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_1) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee x_2) \wedge (\neg x_4 \vee x_3) \wedge (\neg x_5 \vee x_3)$$

And running the instance through the algorithm gives only one

possible solution:

x_1	x_2	x_3	x_4	x_5
True	True	False	False	False

Problem 5:

Run your algorithm on the uf20-01.cnf instance. How many solutions satisfy the problem instance?

The uf20-01.cnf instance consists of an instance with 20 variables and 91 clauses.

After running the instance through the algorithm, it calculated 8 solutions that satisfy the problem, and these were:

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	x_{20}
False	True	True	True	False	False	False	True	True	True	True	False	False	True	True	False	True	True	True	True
True	False	False	False	False	True	False	False	False	False	False	False	True	True	True	False	True	False	False	True
True	False	False	False	False	True	False	False	True	False	False	False	False	True	True	False	True	False	False	True
True	False	False	False	False	True	False	False	True	False	False	False	True	True	True	False	True	False	False	True
True	False	False	True	False	False	False	False	False	True	False	False	True	True	True	False	True	False	False	True
True	False	False	True	False	False	False	True	False	True	False	False	True	True	True	False	True	False	False	True
True	False	False	True	False	True	False	False	False	False	False	False	True	True	True	False	True	False	False	True
True	False	False	True	False	True	False	False	False	True	False	False	True	True	True	False	True	False	False	True

Problem 6:

It is impractical to do the same for a larger instance, say one contained in uf100-430. How could you estimate the time needed by your computer program to complete the task on such an instance? Justify your answer.

With a larger instance such as uf100-430, which has 100 variables and 430 clauses, it would be impractical, as the time complexity for this problem would be:

$$O(2^n)$$

Where n represents the number of variables

For this problem, to estimate the runtime of the program, we're going to assume T as the total number of operations divided by the computer's speed in operations per second.

Assuming the computer can make one billion operations per second (which would correspond to 1GHz), then the total of operations would be:

$$T = \frac{430 \times 2^{100}}{10^9} 5.46 \times 10^{23} \text{ seconds}$$

This corresponds to roughly $1.73 \cdot 10^{16}$ years, or 1.73 quadrillion years.

As we can observe, brute forcing our way to the possible solutions for this problem is not a suitable option for larger datasets, as the time it takes to run the program is exponential to the number of variables.