CS:3205 Artificial Intelligence

Practical 1: WalkSAT

Jason Shipp

Description:

The WalkSAT problem as described in the lecture and in the practical description sheet is based on the idea of creating a combined truth statement, in this case signified by negative and positive numbers, which contains the correct truth value at a certain position for at least one place in every clause provided. In our version of the problem negative numbers correlate to a false, positive to a true, and the absolute value of the number to the position in the truth statement.

The problem is solved via a mixture of a heuristic based on making more statements correct than we break, and flipping a random value a certain percentage of the time. This method avoids being stuck in local maximums which the straight heuristic can follow.

Testing

Using a batch test script I tested my solution to the WalkSAT problem on problems which could be satisfied of sizes 20, 75, and 100 variables. Using a static probability of flipping randomly at 0.4, the 20 variable problems are solved sub 1 second, hovering around 0.6-0.7s, the 75 variable problems range from 5 seconds to 25 seconds, and the 100 sized problems take upwards of minutes, and are not optimally solved.

Using the same scripts to experiment with the probability value, denoted p in the program, I found that as I moved it lower the program became significantly faster before levelling out and slowing down or not being able to finish at all. Moving from p = 0.5, to p = 0.2, the time for completion was much more erratic at p=0.5, as opposed to p=0.2, although the fastest average completion time was reported at p=0.3, for all clause sizes.

Evaluation of Performance

The solution was very efficient at solving all puzzles with 20 variables and 91 clauses, solving most on an average hovering around 0.8s to 0.5s depending on the value of p chosen at the command prompt. Moving into more difficult problems the performance became more erratic, with a range being seen around 30 seconds. This gap was further emphasized at the 100 variable problem where the range was 3 minutes. The 75 variable problem could be called feasibly solved problem, especially with optimal p values chosen, however the 100 variable problem's breadth was too large for my implementation to solve quickly.