

Dpll performance

We will try several examples from the <https://www.cs.ubc.ca/~hoos/SATLIB/benchm.html> from the Uniform Random-3-SAT category, we will try both SAT (and possibly UNSAT) examples we will start at 20 variables, 91 clauses and end 75 variables, 325 clauses. For each category we will try 5 SAT and 5 UNSAT examples. Sadly bigger example took to long to solve.

20 variables

For the SAT version we have solved the first 5 formulas in: 13.9244 ms, 11.1115 ms, 14.221 ms, 13.5392 ms, 13.2869 ms. This gives average of 13.2166 ms.

50 variables

For the SAT version we have solved the first 5 formulas in: 106.1836 ms, 491.2626 ms, 11.1711 ms, 357.6828 ms, 19.6191 ms. This gives average of 197.1838 ms. For the UNSAT version we have solved the first 5 formulas in: 1243.034 ms, 1230.173 ms, 1600.7059 ms, 1694.7185 ms, 2754.1332 ms. This gives average of 1704.5529 ms.

75 variables

For the SAT version we have solved the first 5 formulas in: 2905.1712 ms, 18040.8449 ms, 6233.7036 ms, 24925.6289 ms, 76908.4313 ms. This gives average of 25802.7559 ms. For the UNSAT version we have solved the first 5 formulas in: 141533.6872 ms, 597852.2636 ms, 114975.938 ms, 116615.7823 ms, 114112.5811 ms. This gives average of 217018.0504 ms.

Conclusion

Let's put our averages into a table to demonstrate how quickly the running time grows. SAT means whether the instances had model (true) or not (false).

Variables	Clauses	Running time (ms)	SAT
20	91	13.2	true
50	218	197.2	true
50	218	1704.55	false

Variables	Clauses	Running time (ms)	SAT
75	325	25802.76	true
75	325	217018.05	false

As we compare running times we can notice solving solvable instance is about 10 times faster than solving unsolvable instance on CNF of same size. Another observation is how quickly grows the running time when we add 25 variables.