Simulated Annealing for 3-SAT Problems

30509 Computer Programming Assignment 2024
Alessandro Ferraiolo 3238219
alessandro.ferraiolo@studbocconi.it

Introduction

This report examines the impact of key parameters on the performance of a Simulated Annealing (SA) algorithm for solving random 3-SAT problems. Using a fixed seed (seed=42), parameters were optimized to balance computational efficiency and problem-solving capability. The analysis includes:

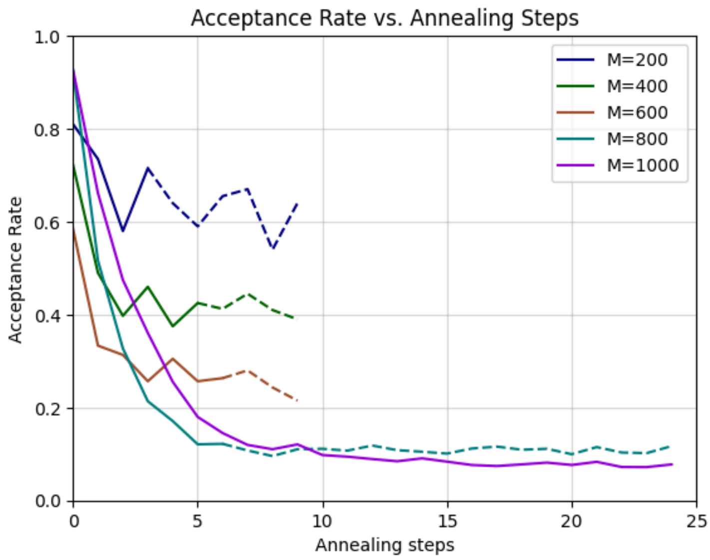
1. **Parameter Efficiency**: Analyzes how Monte Carlo steps (mcmc_steps), annealing steps, and the inverse temperature range ($\beta 1-\beta 0$) influence the solver's ability to find solutions while keeping runtime reasonable.
2. **Acceptance Rate Behavior**: Explores acceptance rate trends and their dependency on the number of clauses (M).
3. **Algorithmic Threshold**: Evaluate the link between the number of variables (N) and the algorithmic threshold (M(alg)), where the solver achieves a 50% success rate.
4. **Solver Power and Accuracy**: Compares two annealing schedules (S1 and S2), highlighting how a stronger solver increases M(alg), nearing the conjectured satisfiability threshold of 4.27.

Parameter fine-tuning

The following parameter values were determined using a fixed seed (seed=42) to achieve efficient problem-solving within a reasonable time frame. Notably, for M=1000, no solution is found:

Final Cost	Ann_Steps to solve	N	M	mcmc steps	Anneal Steps	Beta0	Beta1	Time to Solve (s)
0	3	200	200	200	10	1	10	0.18
0	5	200	400	400	10	1	20	0.36
0	6	200	600	600	10	1	20	0.55
0	6	200	800	3000	25	0.1	20	3.78
7	N/A	200	1000	8000	25	0.1	10	9.36

Analyzing the acceptance rate curves reveals distinct behaviours based on the problem size M. For smaller M values (e.g., M=200 and M=400), the problem is relatively simple, as the larger solution space allows the algorithm to find a global minimum (cost = 0) within a few annealing steps. Once the global minimum is reached, the algorithm continues accepting equivalent global minima or near-zero cost solutions, resulting in the dotted sections of the curves, which appear flat at approximately 60% for M=200 and 40% for M=400. These sections represent states after the solution has been found and are characteristic of the algorithm's behaviour in less constrained systems.



However, if we disregard these dotted sections (indicating the post-solution behaviour), the curves exhibit a consistent downward trend, reflecting the decreasing acceptance rate as the annealing process progresses. This downward slope highlights the growing difficulty in accepting moves as the system cools and the algorithm focuses on refining the solution.

For larger M values (e.g., M=600,800,1000), the problem becomes increasingly constrained, reducing the likelihood of finding a satisfying solution. This results in more pronounced initial drops in the acceptance rate and steeper overall declines. The higher constraints make the search space smaller and the optimization process more challenging, as evident in the rapid reduction in acceptance rates across annealing steps.

Finding Solutions and Computational Constraints

The Simulated Annealing algorithm can consistently find solutions by increasing the number of Monte Carlo steps (mcmc_steps), annealing steps, and increasing the inverse temperature range, $\Delta\beta=\beta1-\beta0$. However, this increases runtime significantly. Interestingly, for N=200 and M=1000, no solution is found regardless of the solver's annealing schedule. This suggests the existence of a threshold between M=800 and M=1000 where the 3-SAT problem with N=200 transitions from solvable to unsolvable.

Empirical Probability P(N,M) and Solver Accuracy

The probability of solving a problem (P(N,M)) depends on the solver's accuracy. A more robust solver can solve more instances, increasing the estimated probability and, consequently, the algorithmic threshold M(alg). To examine the impact of the solver's accuracy on the next results, two annealing schedules were used for testing:

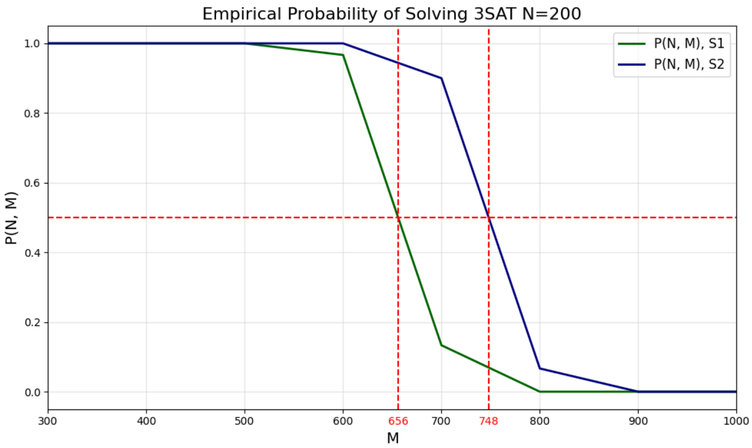
- **"S1" Fast but less accurate:** mcmc_steps=400, annealing steps = 20, beta0 = 0.5, beta1 = 10 (runtime ~10 minutes)
- **"S2" More accurate but time-intensive:** mcmc_steps=3000, annealing steps = 20, beta0 = 0.5, beta1 = 10 (runtime ~1 hour)

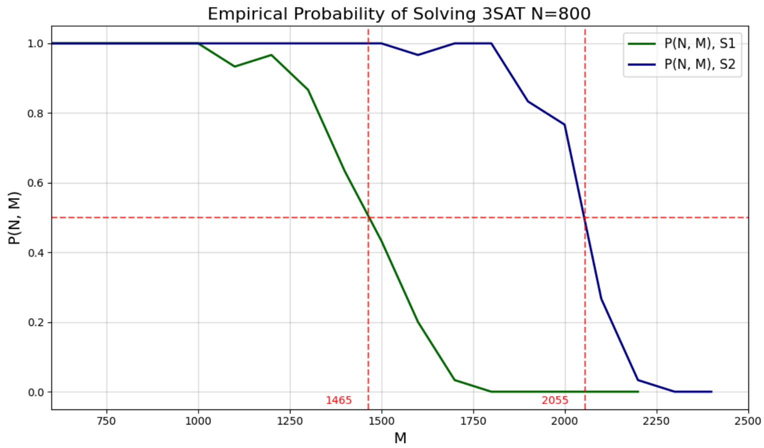
The parameter selection process focused on the influence of mcmc_steps, identified as the key factor driving solver performance in the earlier analysis. Other parameters were held constant while mcmc_steps was adjusted. The runtime represents the duration required by the empirical probability function to evaluate the specified N values across the analyzed M range on the test machine.

The following graph clearly demonstrates the effect of solver performance on P(N,M). Notably:

- Using Solver S1: M(alg)=656
- Using Solver S2: M(alg)=748

As M and N increase, the gap between the two solvers increases. This divergence shows how problem complexity highlights differences in solver accuracy, with Solver S2 showing superior performance for more complex scenarios due to its better handling of challenging configurations.





Probability Curve and Algorithmic Threshold

The S-shaped P(N,M) curve shows a sharp drop from 100% to 0% as M approaches and exceeds M(alg). Beyond this threshold, either is the solver algorithm failing to solve the problems or the problems become fundamentally unsolvable.

The M/N ratio, which represents the number of clauses relative to the number of variables, determines how constrained the problem is:

- **Higher M/N Ratio (More Clauses, Fewer Variables):** Each variable is shared among more clauses, increasing interdependencies and reducing the flexibility of assignments. This significantly shrinks the solution space, making the problem more constrained and harder to solve.
- **Lower M/N Ratio (Fewer Clauses, More Variables):** With fewer constraints relative to the number of variables, assignments have more flexibility, expanding the solution space and making the problem easier to solve.

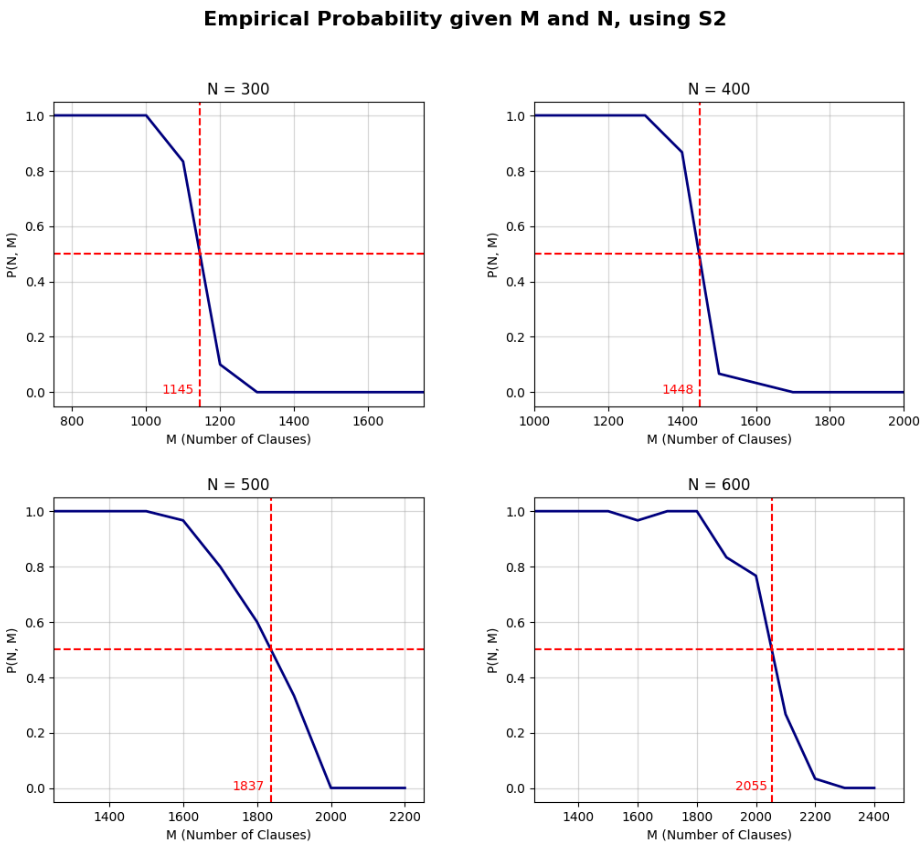
Thus, a higher M/N ratio signals greater problem complexity, while a lower ratio indicates reduced constraints.

As will be explored further, when M approaches M(alg), the ratio M/N converges to approximately 4.27.

Effect of N on M(alg)

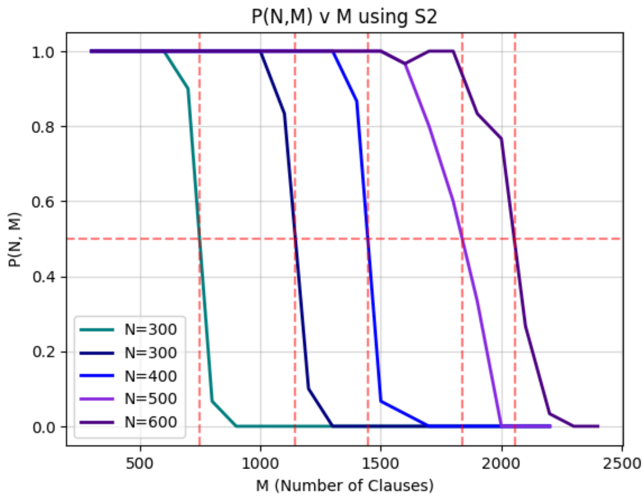
As a consequence of what was explained previously, the algorithmic threshold depends on N. Increasing N provides more variables for constructing clauses, making the problem less constrained and shifting the P(N,M) curve and M(alg) to the right (towards higher values of M). This indicates that with more variables, the problem is solvable even for a higher number of clauses.

The following two graphs clearly show the described effect on N of P(N,M) and M(alg):



The values of M(alg) found with S2 are:

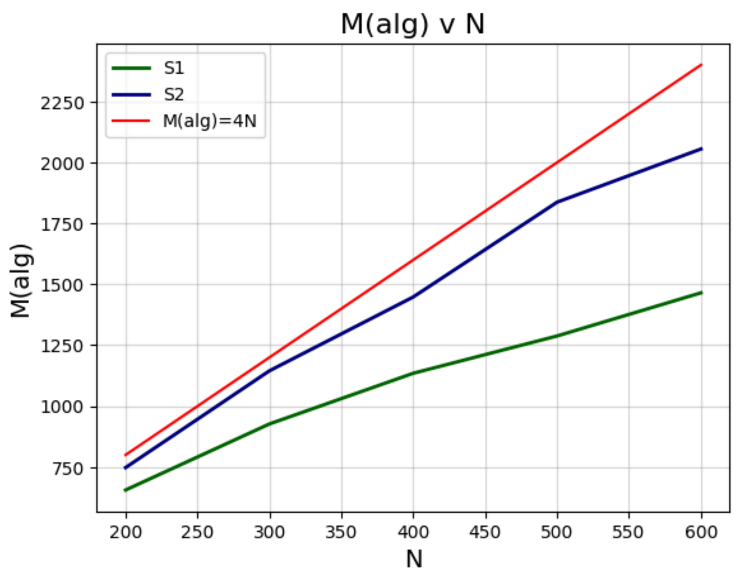
N	200	300	400	500	600
M(alg)	748	1145	1448	1837	2055



Impact of Solver Power

The strength of the solver directly influences M(alg)/N:

- With a weaker solver like S1, M(alg) grows less than proportionally, due to a greater underestimation of the solvable problems.
- With a stronger solver M(alg)/N stabilizes around 4.

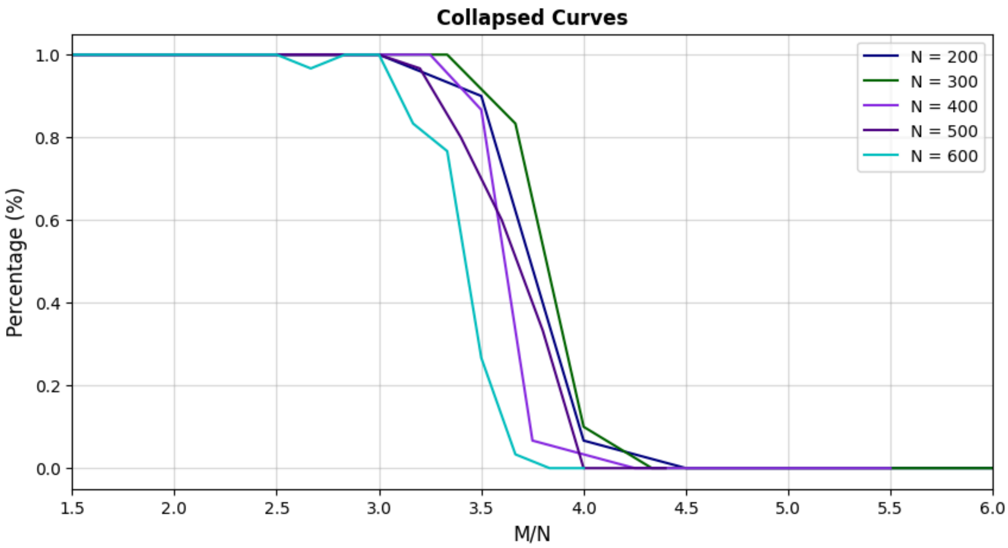


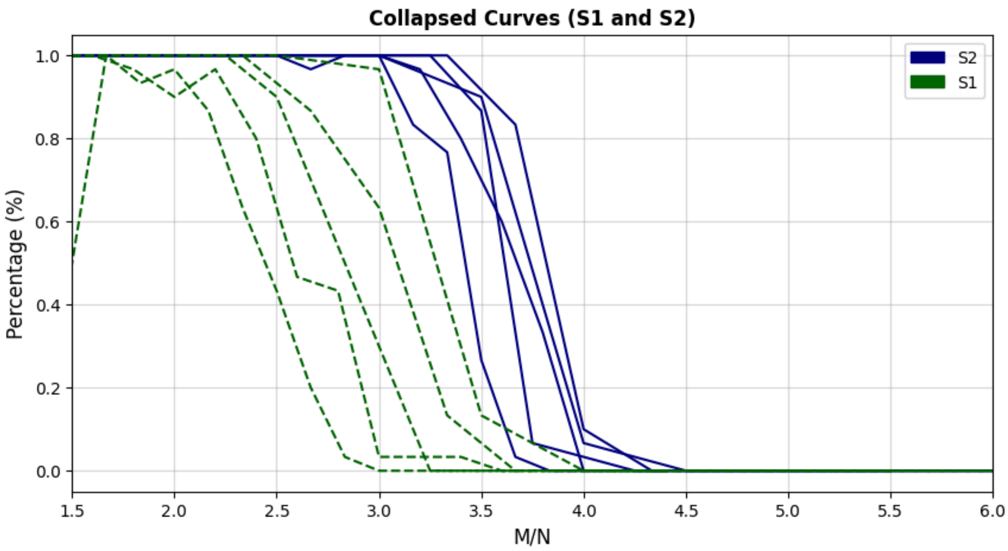
However, as M and N increase, problem complexity rises, causing some solvable instances to remain unsolved, further lowering the estimated P(N,M) and M(alg).

This graph illustrates that as the solver becomes more powerful (leaving fewer solvable problems unsolved), the algorithmic threshold approaches 4.
Notably, the currently conjectured value for the structural threshold is approximately 4.27 [1].

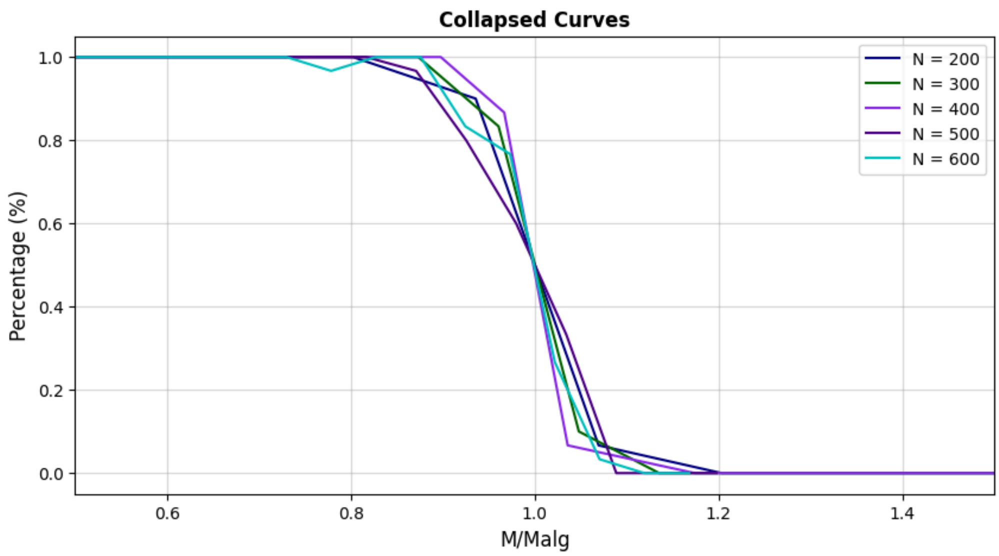
Collapsing curves

Therefore, by scaling M/N, the curves collapse toward a value of 4.27 as the solver's accuracy improves, indicating a structural threshold beyond which the problem becomes unsolvable.
The following two graphs clearly illustrate the critical role of an effective solver in identifying the structural threshold of 3SAT problems and achieving curve collapse.





(Trivially, scaling M/M(alg) collapses all curves around M/M(alg)=1, as, by definition, for any given N when M=M(alg), P(N,M)=0.5.)



Conclusion

The results emphasize the influence of the M/N ratio on problem complexity and solution space constraints. As the solver's power increases, fewer solvable instances are left unsolved, pushing M(alg) closer to the theoretical threshold. Scaling M/N further reveals structural trends in solution behaviour, with the threshold aligning around 4.27, consistent with the findings in Mézard, Mertens, and Zecchina (2003) [1].

References

[1] Mézard, Marc, Stephan Mertens, and Riccardo Zecchina. "Threshold Values of Random K-SAT from the Cavity Method." *arXiv preprint arXiv:cs/0309020*, 2003.