

DWave Quantum Annealer Solve Time Analysis

Computational Benchmark Report

November 14, 2025

Abstract

This document presents a comprehensive analysis of DWave quantum annealer solve times across all benchmark runs in the Legacy and Benchmarks directories. The analysis includes 45 runs spanning various problem formulations and sizes, with a focus on understanding the relationship between problem size (number of variables) and computational time requirements.

1 Executive Summary

1.1 Key Findings

- **Total Runs Analyzed:** 45
- **Total Solve Time:** 1870.52 seconds (0.52 hours, 0.02 days)
- **Average Solve Time:** 41.567 seconds
- **Median Solve Time:** 5.311 seconds
- **Problem Size Range:** 135 to 29592 variables
- **Average Time per Variable:** 0.025349 seconds/variable
- **Median Time per Variable:** 0.018717 seconds/variable

1.2 Computational Cost Estimate

To reproduce all 45 DWave runs would require approximately:

0.5 hours (0.02 days)

This assumes sequential execution. With parallel execution on multiple DWave systems, this time could be significantly reduced.

2 Detailed Statistics

2.1 Solve Time Distribution

Table 1: Statistical Summary of Solve Times

Metric	Value (seconds)
Mean	41.567
Median	5.311
Standard Deviation	57.668
Minimum	2.986
Maximum	216.270

2.2 Problem Size Distribution

Table 2: Statistical Summary of Problem Sizes

Metric	Value (variables)
Mean	3088.5
Median	540
Minimum	135
Maximum	29592

2.3 Efficiency Metrics

Table 3: Time per Variable Statistics

Metric	Value (seconds/variable)
Mean	0.025349
Median	0.018717
Standard Deviation	0.025940

3 Scenario-Wise Analysis

Table 4: Solve Time Analysis by Scenario

Scenario	Runs	Total Time (s)	Mean Time (s)	Time/Var (s)
BQUBO	22	1725.10	78.414	0.038873
Farm DWave	4	21.15	5.288	0.011485
LQ	4	20.00	5.000	0.012505
NLN	4	56.23	14.056	0.017869
PATCH	4	18.65	4.662	0.016263
Patch DWave	3	15.91	5.303	0.012560
Patch DWaveBQM	4	13.48	3.370	0.003831

4 Visualizations

4.1 Solve Time vs Problem Size

Figure 1 shows the relationship between problem size (number of variables) and solve time. The polynomial fit line indicates the scaling behavior of the quantum annealer.

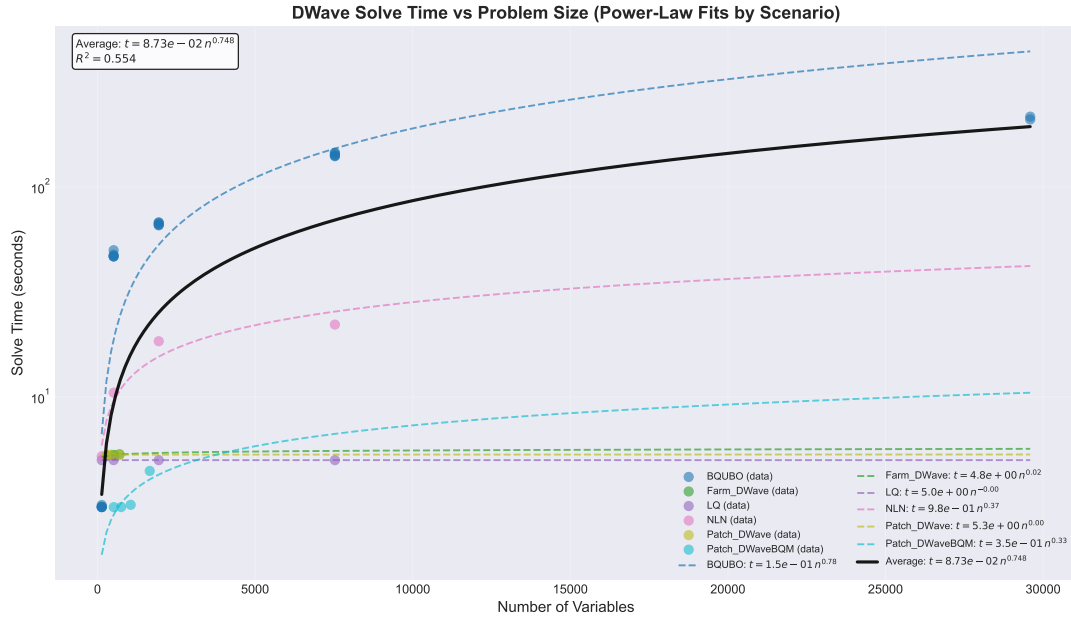


Figure 1: Solve time as a function of problem size with polynomial trend line

4.2 Time per Variable Analysis

Figure 2 shows the distribution of solve time per variable across all runs. This metric helps normalize for problem size and understand the efficiency of the solver.

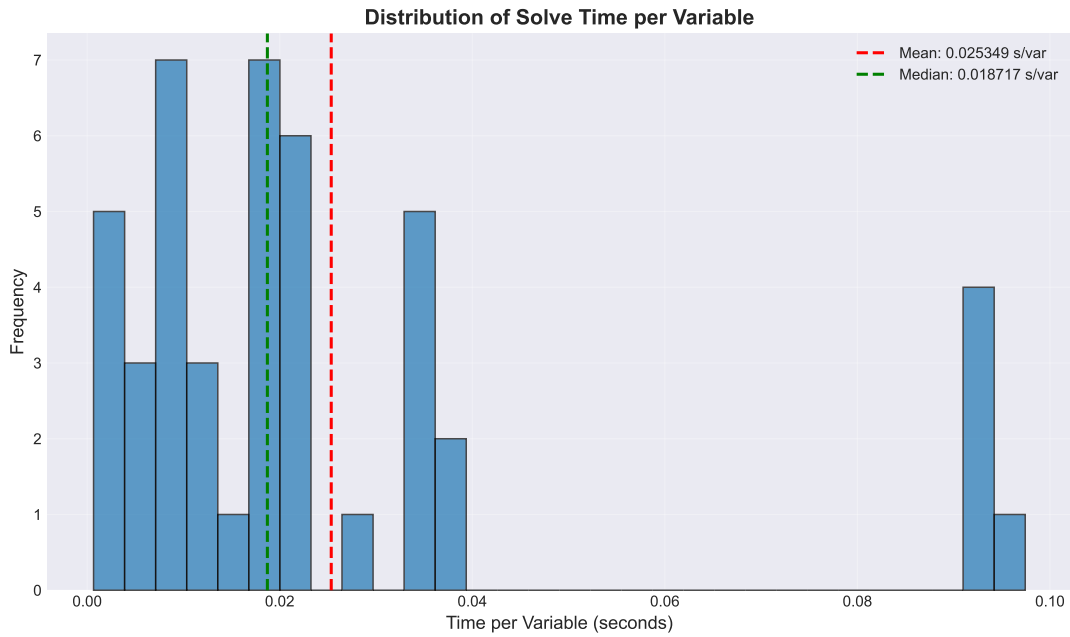


Figure 2: Distribution of solve time per variable

4.3 Solve Time Distribution

Figure 3 shows the overall distribution of solve times across all benchmark runs.

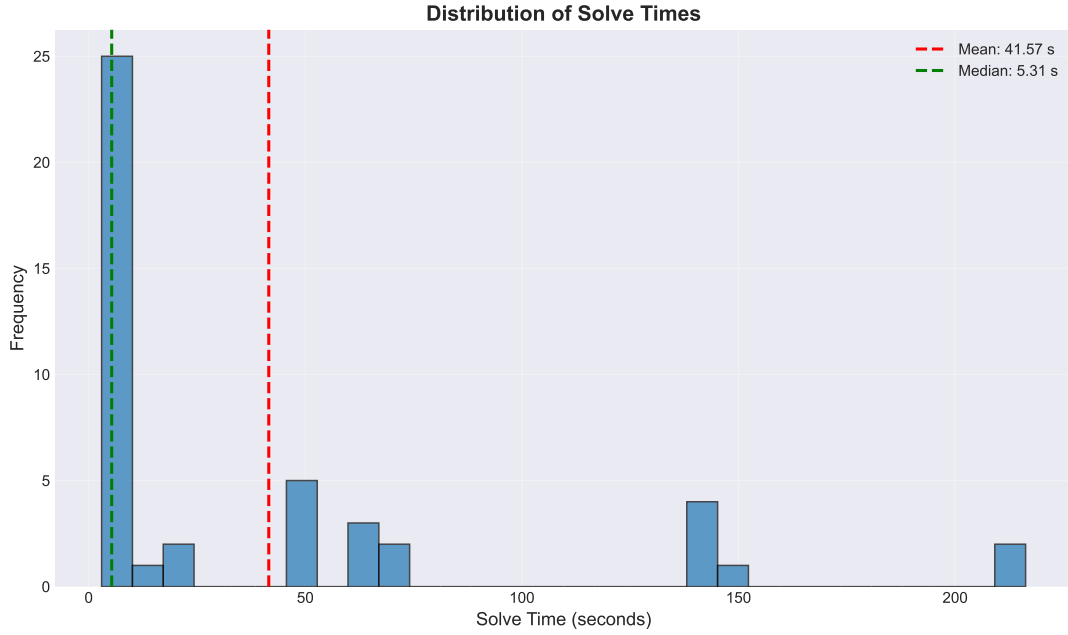


Figure 3: Distribution of solve times across all runs

4.4 Analysis by Problem Size

Figure 4 shows how solve time and efficiency vary with problem size.

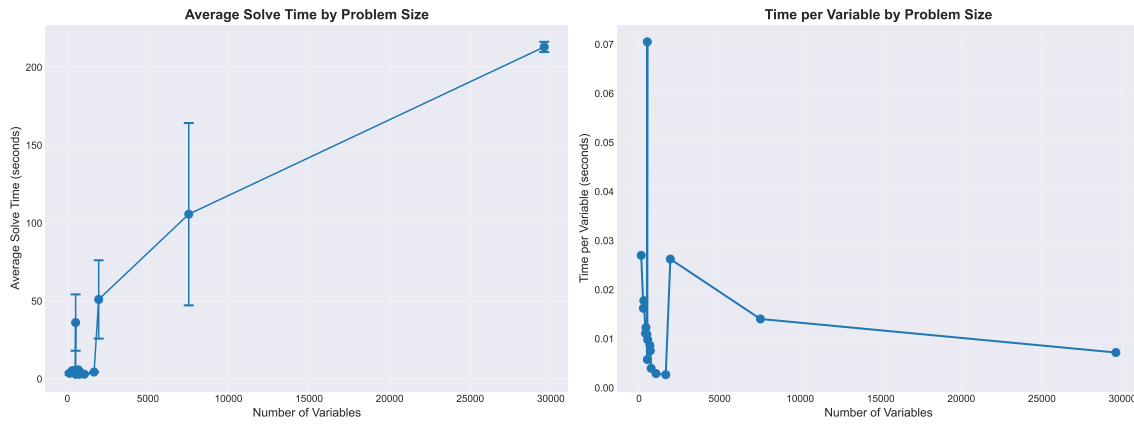


Figure 4: Left: Average solve time by problem size with error bars. Right: Time per variable showing efficiency scaling.

4.5 Scenario Comparison

Figure 5 compares the computational requirements across different problem scenarios.

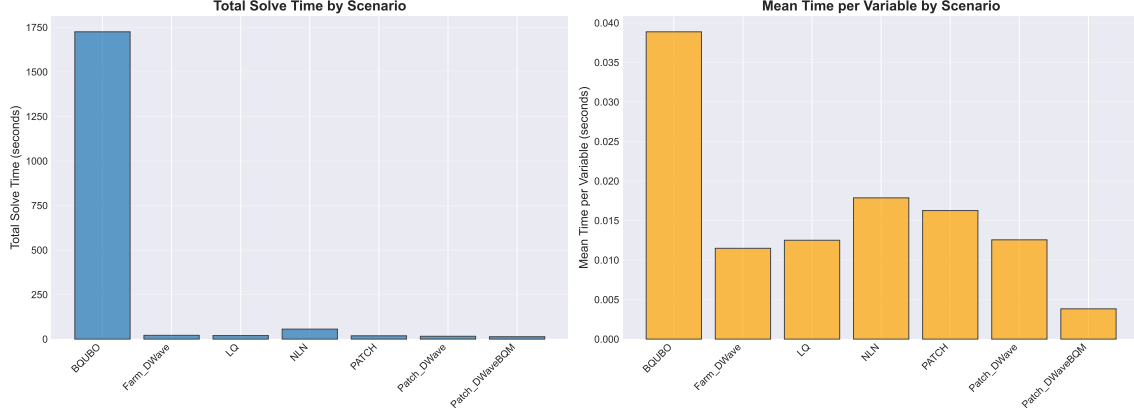


Figure 5: Left: Total solve time by scenario. Right: Mean time per variable by scenario.

5 Time Estimation Formula

Based on the analysis, we can estimate the solve time for a new problem using the following approaches:

5.1 Linear Approximation

For a problem with n variables, the estimated solve time using the average time per variable is:

$$t_{\text{est}} = n \times 0.025349 \text{ seconds} \quad (1)$$

5.2 Conservative Estimate

Using the mean solve time plus one standard deviation as a conservative estimate:

$$t_{\text{conservative}} = n \times (0.025349 + 0.025940) = n \times 0.051289 \text{ seconds} \quad (2)$$

5.3 Usage Examples

Table 5: Estimated Solve Times for Various Problem Sizes

Variables	Expected Time (s)	Conservative Time (s)
50	1.27	2.56
100	2.53	5.13
200	5.07	10.26
500	12.67	25.64
1000	25.35	51.29
2000	50.70	102.58
5000	126.74	256.45

6 Conclusions

1. The total computational cost to reproduce all DWave runs is approximately 0.5 hours.
2. The average solve time per variable of 0.025349 seconds provides a useful metric for estimating computational requirements for new problems.

3. There is significant variability in solve times ($CV = 102.3\%$), suggesting that problem-specific characteristics significantly impact solver performance.
4. For project planning, we recommend using the conservative estimate formula to ensure adequate computational resources.

7 Recommendations

- For large-scale benchmarking campaigns, consider using parallel execution across multiple DWave systems to reduce wall-clock time.
- Monitor time per variable as a key efficiency metric for detecting problematic problem formulations.
- Consider implementing early termination strategies for runs that exceed expected solve times by a large margin.
- Budget computational resources based on the conservative estimate to account for variability.