

Quantifying the Stability–Scalability Trade-off of First-Order and Quasi-Newton Optimization Methods Under Controlled Conditioning

Mohammed Murtaza Tajammul^{1*}, Iyad Mohiuddin², Syed Sulayman²

¹ Department of Computer Science Engineering, Muffakham Jah College of Engineering and Technology, Hyderabad, India.

² Lords Institute of Engineering and Technology, Hyderabad, India.

ABSTRACT

The choice of optimization algorithm affects not only training time and final performance but also the reliability of the training process. While optimizers are commonly compared using accuracy and computational cost, their behavior under changes in problem conditioning and scale is less often considered together. This paper compares Stochastic Gradient Descent (SGD), Momentum SGD, Adam, and L-BFGS using controlled synthetic linear regression problems. The experiments independently varied sample size, feature dimensionality, condition number, and learning rate. Three sample sizes (1,000, 10,000, and 100,000), three feature dimensionalities (10, 100, and 1,000), three condition numbers (10, 100, and 10,000), and three learning rates (0.001, 0.01, and 0.1) were tested, giving a total of 324 experimental runs. Optimizer behavior was evaluated primarily through failure-to-converge rate, with loss oscillation and wall-clock runtime used as additional measures. Within the tested ranges, conditioning had a much stronger effect on convergence stability than sample size. SGD and Momentum SGD had overall divergence rates of 50.6% and 45.7%, respectively, and both diverged in every run at a condition number of 10,000. Adam had an overall divergence rate of 9.9%, increasing to 29.6% under the most severe conditioning. L-BFGS recorded the lowest overall divergence rate at 3.7% and an 11.1% divergence rate at a condition number of 10,000, but required substantially more computation time per run. The results indicate a trade-off between computational cost and convergence reliability. Under the conditions examined here, first-order methods were computationally cheaper, while L-BFGS provided greater stability as conditioning became severe. The findings also suggest that problem conditioning can be a more useful indicator of convergence difficulty than dataset size alone when comparing these optimization methods.

Keywords: Machine Learning, Optimization Algorithms, SGD, Momentum SGD, Adam, L-BFGS, Optimization Stability, Problem Conditioning, Scalability, Convergence

INTRODUCTION

Optimization plays an important role in machine learning because it affects how quickly a model trains and whether the training process converges reliably. Optimizers are often compared using measures such as final loss, accuracy, or training time. These measures are useful, but they do not always show how consistently an optimizer behaves when the optimization problem becomes difficult.

Problem conditioning is one factor that can affect this behavior. In a poorly conditioned problem, the objective changes at very different rates in different directions. This can make gradient-based methods more sensitive to the learning rate and may lead to

slow convergence, oscillation, or divergence. First-order methods such as Stochastic Gradient Descent (SGD), Momentum SGD, and Adam are widely used because their updates are relatively inexpensive. L-BFGS, a quasi-Newton method, uses approximate curvature information and can behave differently on poorly conditioned problems, although its updates generally require more computation.

With real-world datasets, factors such as sample size, feature dimensionality, feature correlations, and conditioning can be related to each other. This makes it difficult to determine which factor is mainly responsible for optimizer instability. To avoid this problem, the present study uses controlled synthetic linear re-

Relevant conflicts of interest/financial disclosures: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.



gression problems in which sample size, feature dimensionality, condition number, and learning rate can be varied independently.

The main aim of this study is to examine whether increasing dataset size or increasing problem conditioning has a stronger relationship with optimizer instability. The study also compares SGD, Momentum SGD, Adam, and L-BFGS in terms of convergence reliability and computational cost under the same experimental conditions.

II. RELATED WORK

Problem conditioning has long been studied in numerical optimization because it can affect the convergence of gradient-based methods. When an objective is poorly conditioned, different directions can have very different rates of change, which can make optimization slower and more sensitive to the choice of update size. Quasi-Newton methods address this issue by using information about the curvature of the objective. L-BFGS is a limited-memory version of BFGS designed for large-scale problems where storing the full Hessian is expensive [1], [3].

First-order methods are widely used in machine learning because their updates are relatively inexpensive. SGD uses gradient information directly, Momentum SGD uses previous updates to influence the current update, and Adam adapts the update using estimates of the first and second moments of the gradients [2].

Previous studies have compared these methods in different application settings. Richardson [4] compared optimization methods for seismic full-waveform inversion and reported faster convergence with Adam than L-BFGS-B in the experiments considered. Xu and Darve [5] examined optimization problems involving PDE solvers and deep neural networks, including the effects of poor conditioning. Jena et al. [6] studied optimization on quadratic objectives with different condition numbers and reported a strong decline in SGD performance as conditioning became more severe, while Adam was less affected in their setting. Other studies have also shown that adaptive optimization methods do not consistently outperform standard gradient-based methods and that optimizer performance depends on the structure of the problem [7], [8].

Although these studies provide useful comparisons, many focus on particular applications or use experimental settings where factors such as sample size, feature dimensionality, and conditioning cannot be separated easily. The present study uses controlled synthetic linear regression problems to vary these factors independently. This allows the stability of SGD, Momentum SGD, Adam, and L-BFGS to be compared under the same conditions and provides a direct comparison of the effects of sample size and problem conditioning.

METHODOLOGY

A. Synthetic Data Generation

Synthetic linear regression problems were used so that sample size, feature dimensionality, and condition number could be controlled independently. This avoids some of the relationships between these factors that occur naturally in real-world datasets.

For each target condition number, a random Gaussian base matrix was generated. Its covariance structure was modified using eigenvalues distributed logarithmically between 1 and the target condition number. A random orthogonal rotation matrix obtained through QR decomposition was then applied. This procedure produced a design matrix with controlled conditioning while avoiding dependence on a particular coordinate system.

A coefficient vector (w) was generated with values uniformly distributed between -5 and 5 . The target values were calculated as

$$[y = Xw + \epsilon]$$

where (X) is the design matrix, (w) is the coefficient vector, and (ϵ) represents Gaussian noise.

The generated data were checked to determine whether the measured condition numbers were close to the requested values. A target condition number of 10 produced a measured value of approximately 9, while a target of 10,000 produced a measured value of approximately 8,900. These measurements indicate that the data-generation procedure produced problems with conditioning close to the intended values.

B. Optimizers Evaluated

Four optimization methods were compared: SGD, Momentum SGD, Adam, and L-BFGS. Momentum SGD used a momentum coefficient of 0.9, while

Adam used its standard default beta parameters. L-BFGS was used as the quasi-Newton method because it approximates curvature information without storing the complete Hessian. All four methods were implemented using the built-in optimization functions provided by PyTorch.

C. Experimental Design

A full factorial design was used to vary sample size, feature dimensionality, condition number, optimizer, and learning rate.

- **Sample sizes:** 1,000, 10,000, 100,000
- **Feature dimensionalities:** 10, 100, 1,000
- **Condition numbers:** 10, 100, 10,000
- **Learning rates:** 0.001, 0.01, 0.1

Each configuration was trained for 50 optimization steps using the same random initialization and fixed seed. This resulted in

[$3 \times 3 \times 3 \times 4 \times 3 = 324$]

experimental runs. The 50-step limit provided the same number of optimization steps for all methods, although the computational cost of each step differed between optimizers.

D. Stability Metrics

Several measures were used to evaluate the behavior of the optimizers.

- 1) **Failure-to-Converge:** A run was classified as divergent if the loss became NaN or infinite, or if the final loss exceeded 1×10^6 . The same threshold was applied to all optimizers.
- 2) **Loss Oscillation:** Loss oscillation was measured using the standard deviation of the changes in loss between consecutive optimization steps. This was used to examine how strongly the loss varied during training.
- 3) **Gradient-Norm Dynamics:** The mean L2 norm of the gradient across the training steps was used to examine changes in gradient magnitude during optimization.
- 4) **Computational Cost:** Wall-clock time per run was recorded to compare the computational requirements of the four optimization methods.

Failure-to-converge was treated as the main stability measure because a high oscillation value does not necessarily mean that an optimizer has failed. Some

methods may make large corrective updates during training and still converge successfully.

RESULTS AND DISCUSSION

A. Effect of Conditioning on Optimizer Stability

The experimental results show a clear relationship between problem conditioning and optimizer stability. Figure 1 compares the divergence rates of the four optimizers at condition numbers of 10, 100, and 10,000, with the results averaged across sample sizes, feature dimensionalities, and learning rates.

At lower condition numbers, the first-order methods show relatively better stability. However, as the condition number increases, the difference between the optimizers becomes much more noticeable. At a condition number of 10,000, both SGD and Momentum SGD have a divergence rate of 100%. Adam performs better, but its divergence rate increases to 29.6%. L-BFGS remains the most stable method, with a divergence rate of 11.1% at the same condition number.

These results indicate that severe ill-conditioning has a strong effect on the stability of the first-order methods tested in this study. L-BFGS is affected to a much smaller extent, although it requires considerably more computation.

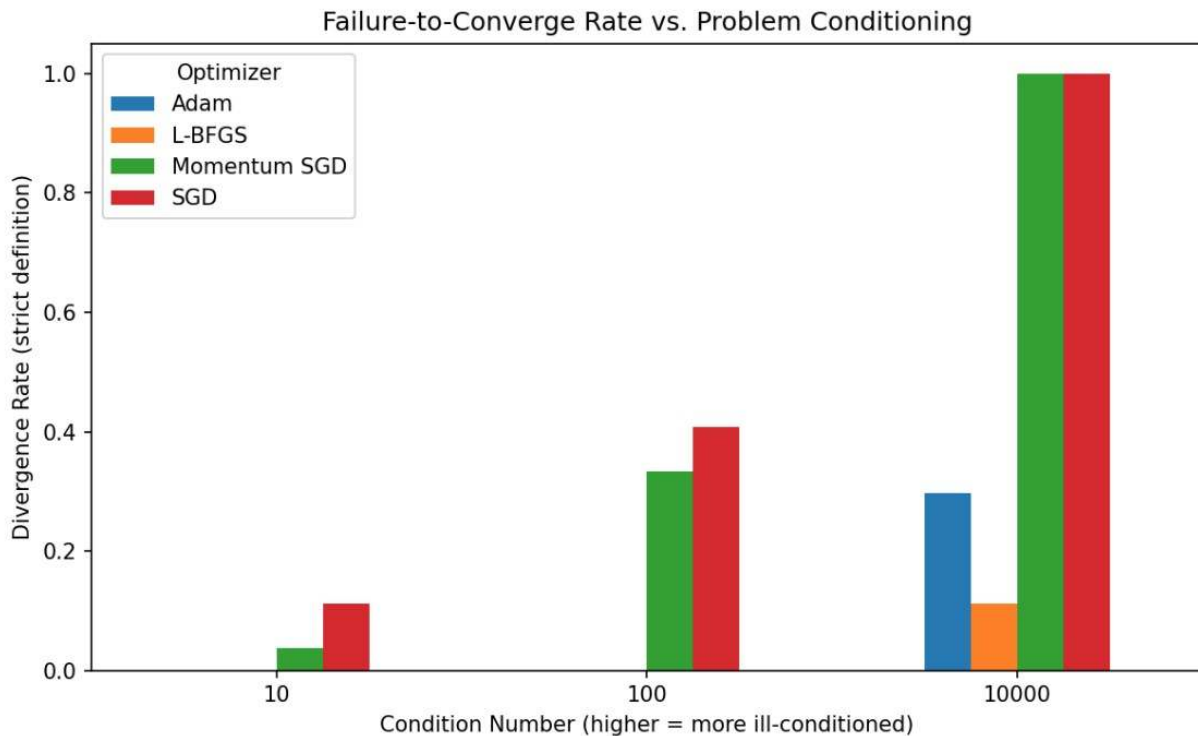


Figure 1: Divergence rate (strict definition) by optimizer across condition numbers 10, 100, and 10,000, averaged across all sample sizes, dimensionalities, and learning rates.

B. Effect of Sample Size on Stability

Figure 2 presents the divergence rates for sample sizes of 1,000, 10,000, and 100,000. The results are noticeably different from those observed when the condition number is changed.

For SGD, the divergence rate changes from 55.6% at 1,000 samples to 44.4% at 100,000 samples. Momentum SGD changes from 48.1% to 44.4%. Adam remains at 11.1% for both sample sizes, while L-BFGS remains at 3.7%.

The relatively small changes across the tested sample sizes suggest that increasing the number of samples alone did not produce a major increase in optimizer instability. This is different from the strong increase in divergence observed when the condition number was raised.

Within the ranges examined in this study, the results therefore suggest that conditioning is a more important factor in optimizer stability than sample size.

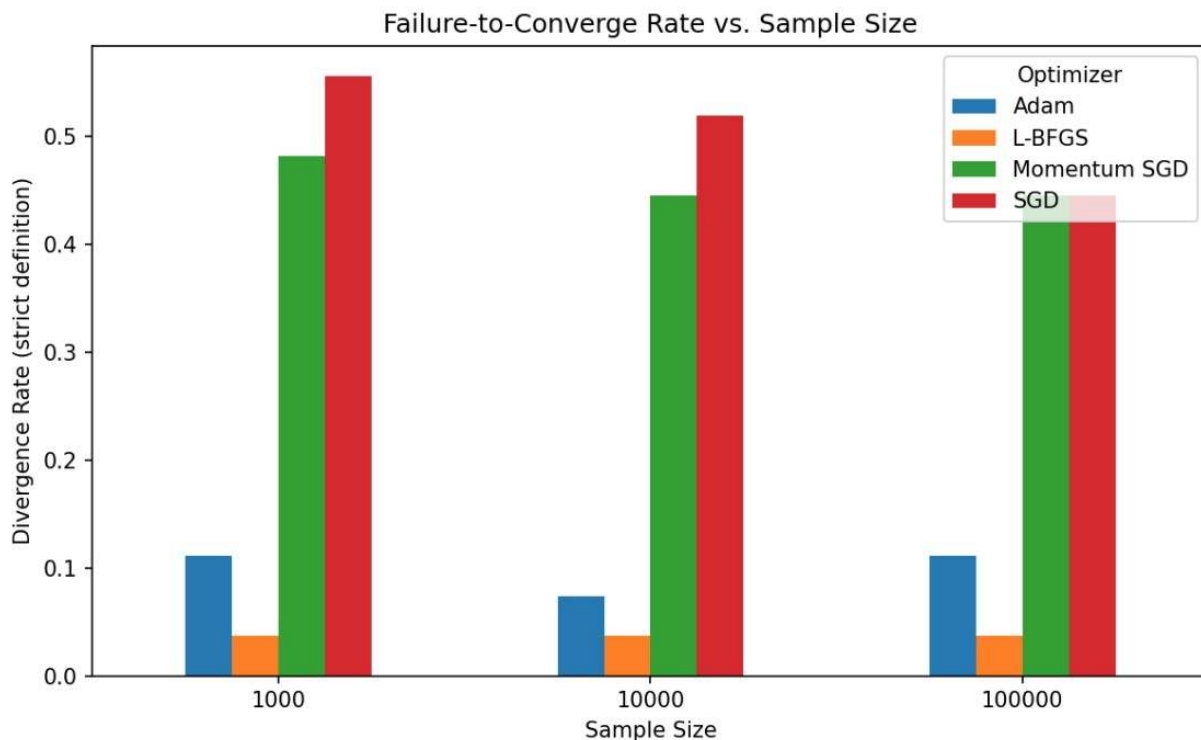


Figure 2: Divergence rate by optimizer across sample sizes 1,000, 10,000, and 100,000, averaged across all dimensionalities, condition numbers, and learning rates.

C. Final Loss of Non-Divergent Runs

The final loss was analyzed separately for runs that did not meet the divergence criterion. Figure 3 shows the mean final loss across different feature dimensionalities.

Momentum SGD achieves the lowest average final loss among the successful runs at lower dimensionalities. However, this result needs to be considered together with the divergence results. As dimensionality and conditioning increase, a larger number of

Momentum SGD runs fail and are therefore not included in the final-loss calculation.

This means that the lower final loss observed for Momentum SGD does not necessarily indicate better overall performance. The calculation only considers runs that successfully completed the experiment. L-BFGS, in comparison, shows a more consistent final-loss profile across the tested dimensionalities.

The results demonstrate why final loss should not be considered independently of convergence reliability when comparing optimization methods.

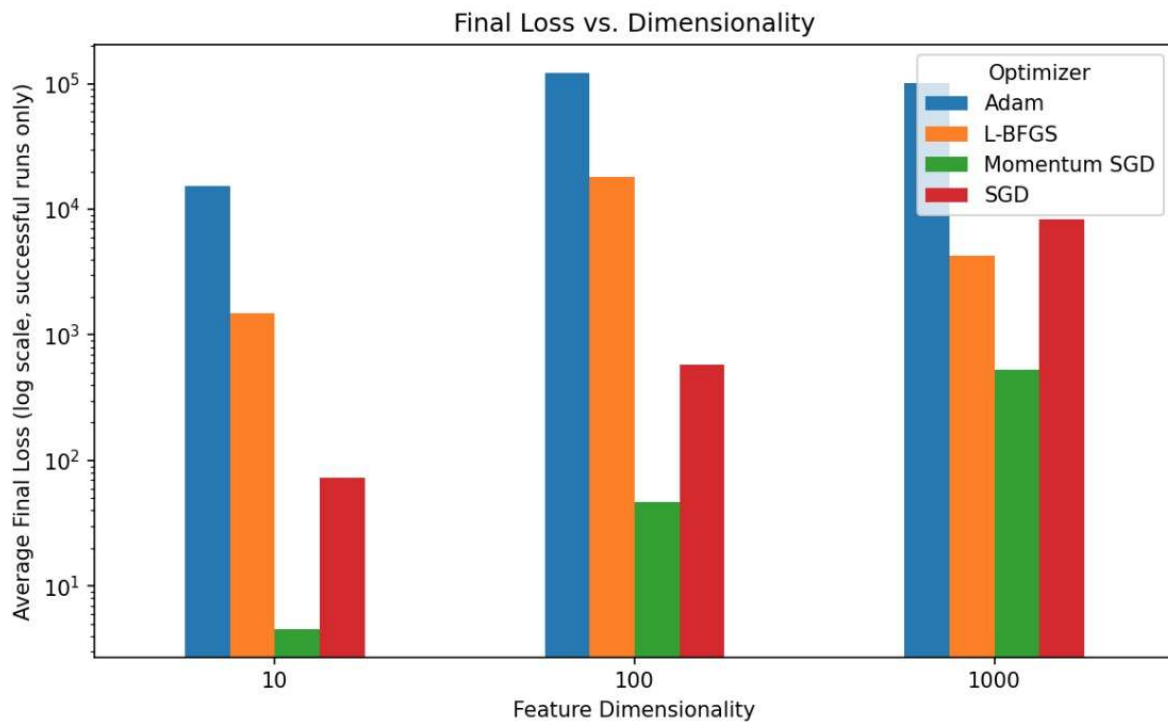


Figure 3: Mean final loss (log scale) by feature dimensionality, restricted to non-divergent runs only.

D. Loss Oscillation

Figure 4 shows the average loss oscillation among the non-divergent runs. L-BFGS shows relatively high oscillation values compared with some of the first-order methods.

However, the oscillation values should not be interpreted as a direct measure of failure. L-BFGS can make relatively large corrective updates during the early stages of optimization while its curvature approximation is being adjusted. These changes can produce higher oscillation values even when the optimizer eventually settles near a minimum.

The divergence results provide a more useful measure of reliability in this case. L-BFGS has the lowest overall divergence rate despite showing comparatively large loss oscillations. For this reason, divergence rate was treated as the primary stability measure, while oscillation was considered a supporting metric.

At a condition number of 10,000, no bars are shown for SGD and Momentum SGD in Figure 4 because all runs for these two methods diverged at that condition number.

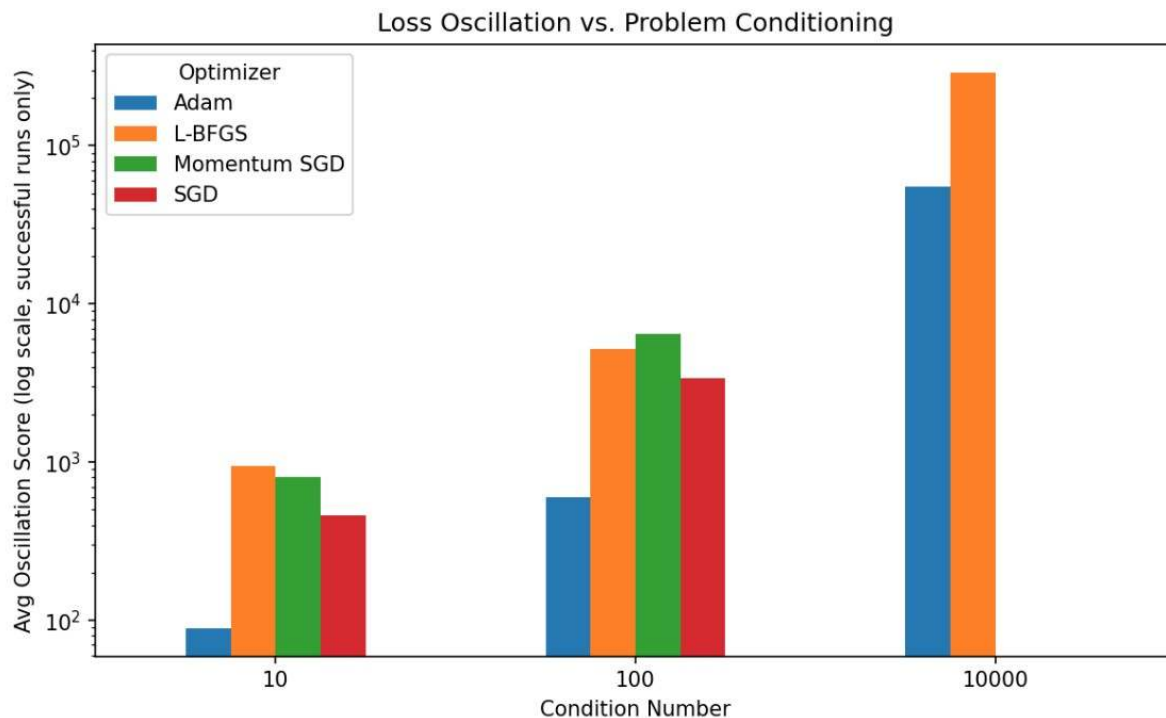


Figure 4: Mean loss oscillation score (log scale) by condition number, restricted to non-divergent runs. No bar is shown for SGD/Momentum SGD at condition number 10,000 because 100% of their runs diverged at that setting.

E. Reliability and Computational Cost

Table 1 summarizes the overall reliability–cost trade-off across all four optimizers, aggregated across the full experimental sweep.

Table 1: Reliability–cost trade-off summary across all 324 experiment runs.

Optimizer	Overall Divergence Rate	Divergence Rate at $\kappa=10,000$	Avg. Time / Run (s)
L-BFGS	3.7%	11.1%	8.92
Adam	9.9%	29.6%	0.40
Momentum SGD	45.7%	100%	0.25
SGD	50.6%	100%	0.26

The results show a clear trade-off between reliability and computational cost. L-BFGS has the lowest overall divergence rate, but its average time per run is much higher than that of the first-order methods. Adam occupies a middle position: its divergence rate is considerably lower than those of SGD and Momentum SGD, while its computational cost remains relatively low.

SGD and Momentum SGD have the lowest average computational times, but their stability decreases sharply under severe conditioning. Both methods fail in every tested run at a condition number of 10,000.

DISCUSSION

The main finding is that, within the tested ranges, problem conditioning showed a much stronger relationship with optimizer instability than sample size. Increasing the number of samples from 1,000 to 100,000 did not produce a consistent increase in divergence, while increasing the condition number led to much higher divergence, particularly for SGD and Momentum SGD.

Adam was more stable than the two SGD-based methods, although its divergence rate reached 29.6% at a condition number of 10,000. L-BFGS had the

lowest overall divergence rate at 3.7%, but its average runtime of 8.92 seconds per run was much higher than Adam at 0.40 seconds and SGD and Momentum SGD at approximately 0.25 seconds.

These results show a clear trade-off between convergence reliability and computational cost. However, the findings are limited to the synthetic linear regression problems, parameter ranges, and 50-step training budget used in this study. Different results may occur with longer training, different learning rates, other objectives, or non-convex models.

CONCLUSION

This study compared SGD, Momentum SGD, Adam, and L-BFGS using 324 controlled synthetic linear regression experiments. The experiments varied sample size, feature dimensionality, condition number, and learning rate to examine how these factors were related to optimizer stability and computational cost.

The results show that, within the ranges tested, sample size did not produce the same change in divergence as problem conditioning. Increasing the condition number was associated with a substantial increase in divergence, particularly for SGD and Momentum SGD. Both methods reached a 100% divergence rate at a condition number of 10,000. Adam was more stable, with an overall divergence rate of 9.9%, although its divergence rate increased to 29.6% at the same condition number.

L-BFGS had the lowest overall divergence rate at 3.7% and a divergence rate of 11.1% under the most severe conditioning tested. This higher reliability came with a much greater computational cost, with an average time per run of 8.92 seconds compared with 0.40 seconds for Adam and approximately 0.25 seconds for SGD and Momentum SGD.

These results show that optimizer selection involves a trade-off between computational cost and convergence reliability. Under the conditions examined in this study, first-order methods were considerably less expensive computationally, while L-BFGS provided greater stability as conditioning became more severe. Adam occupied an intermediate position, offering better stability than SGD and Momentum SGD without the computational cost observed for L-BFGS.

The results are limited to the synthetic linear regression problems, parameter ranges, and 50-step iteration budget used in the experiments. Further work using

longer training periods, a broader range of learning rates, and non-convex machine-learning models would help determine whether the same patterns are observed in more complex optimization settings.

ACKNOWLEDGMENT

The authors would like to thank their respective institutions for providing the academic environment and resources that supported this work.

REFERENCES

1. D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, pp. 503–528, 1989.
2. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. International Conference on Learning Representations (ICLR)*, 2015.
3. J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.
4. A. Richardson, "Seismic Full-Waveform Inversion Using Deep Learning Tools and Techniques," arXiv:1801.07232, 2018.
5. K. Xu and E. Darve, "Trust Region Method for Coupled Systems of PDE Solvers and Deep Neural Networks," arXiv:2105.07552, 2021.
6. R. Jena, P. Chaudhari, and J. C. Gee, "FireANTs: Adaptive Riemannian Optimization for Multi-Scale Diffeomorphic Matching," arXiv:2404.01249, 2024.
7. Z. Zhou, Q. Zhang, G. Lu, H. Wang, W. Zhang, and Y. Yu, "AdaShift: Decorrelation and Convergence of Adaptive Learning Rate Methods," in *Proc. International Conference on Learning Representations (ICLR)*, 2019.
8. I. Ng, S. Lachapelle, N. R. Ke, S. Lacoste-Julien, and K. Zhang, "On the Convergence of Continuous Constrained Optimization for Structure Learning," in *Proc. 25th International Conference on Artificial Intelligence and Statistics*, PMLR, vol. 151, pp. 8176–8198, 2022.

HOW TO CITE: Mohammed Murtaza Tajammul, Iyad Mohiuddin, Syed Sulayman, Quantifying the Stability –Scalability Trade-off of First-Order and Quasi-Newton Optimization Methods Under Controlled Conditioning, *Int. J. Sci. R. Tech.*, 2026, 3 (10), 827-835. <https://doi.org/10.5281/zenodo.23339577>