OVERLAP-RESAMPLED L-BFGS FOR PHYSICS-INFORMED NEURAL NETWORKS

FARS
Analemma
fars@analemma.ai

ABSTRACT

Physics-informed neural networks (PINNs) benefit from both L-BFGS optimization, which provides fast convergence via curvature information, and collocation resampling, which improves domain coverage. However, these techniques are fundamentally incompatible: L-BFGS requires consistent gradients across iterations, while resampling changes the loss function at each step. We propose *overlap-resampled L-BFGS*, which computes curvature pairs only on points that persist between consecutive collocation batches, combined with cautious updates that filter unreliable estimates. On the ice-shelf inverse problem, our method achieves $B_{\text{err}} = 8.06 \times 10^{-4}$, outperforming both Adam with resampling (7% improvement) and fixed-collocation L-BFGS (30% improvement). On the 2D Poisson forward problem, it provides 8.5× improvement over Adam-only while maintaining resampling capability. The method operates stably at overlap fractions as low as 25%, well below theoretical thresholds, demonstrating practical robustness for PINN applications.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Physics-informed neural networks (PINNs) have emerged as a powerful paradigm for solving partial differential equations by embedding physical constraints directly into neural network training (Raissi et al., 2017; Karniadakis et al., 2021). A key factor in PINN success is the choice of optimizer: while first-order methods like Adam provide robust initial training, the quasi-Newton L-BFGS algorithm typically achieves superior final accuracy by leveraging curvature information (Lu et al., 2019). The standard practice combines both in a two-phase approach—Adam warmstart followed by L-BFGS refinement—which has become the default in popular PINN libraries.

However, this two-phase approach faces a fundamental tension with collocation resampling, a technique that refreshes the training points at each iteration to improve domain coverage. Resampling has proven particularly valuable for inverse problems, where it can mitigate bimodal training outcomes and improve parameter inference (Iwasaki & Lai, 2022). Yet L-BFGS fundamentally requires consistent gradients across iterations to build valid curvature estimates: when collocation points change, the loss function changes, and the gradient differences that L-BFGS relies upon become meaningless. This incompatibility forces practitioners to choose between L-BFGS convergence (with fixed points) or resampling coverage (with Adam only).

We resolve this tension by adapting techniques from multi-batch L-BFGS optimization (Berahas et al., 2016; Berahas & Takác, 2017) to the PINN setting. The key insight is to compute curvature pairs only on the *overlap set*—points that persist between consecutive iterations—ensuring gradient differences reflect the same underlying loss. Combined with a cautious update rule that filters unreliable curvature estimates, this enables L-BFGS to run stably under collocation resampling. Our contributions are:

[1] https://gitlab.com/fars-a/overlap-lbfgs-collocation-resampling

- We propose *overlap-resampled L-BFGS*, an algorithm that computes curvature pairs on the overlap set between consecutive collocation batches, enabling L-BFGS optimization under collocation resampling.
- We introduce a *three-phase training pipeline* that transitions from Adam with resampling (exploration) through Adam with fixed points (curvature initialization) to overlap-resampled L-BFGS (refinement with continued resampling).
- We validate the method on complementary benchmarks: on the ice-shelf inverse problem, Overlap-LBFGS achieves 30% lower error than fixed-collocation L-BFGS; on the Poisson forward problem, it provides 8.5× improvement over Adam-only while maintaining resampling capability.

2 Related Work

2.1 PINN Optimization

The optimization of physics-informed neural networks presents unique challenges distinct from standard deep learning. The seminal DeepXDE library (Lu et al., 2019) established the two-phase Adam-to-L-BFGS training paradigm that remains standard practice, leveraging Adam's robustness for initial exploration followed by L-BFGS's fast local convergence. However, this approach requires fixing collocation points during the L-BFGS phase, sacrificing the coverage benefits of resampling.

Several works have characterized PINN training difficulties. Wang et al. (2020a) identified gradient pathologies arising from imbalanced loss components, proposing adaptive weighting schemes. Wang et al. (2020b) analyzed training failures through the neural tangent kernel framework, revealing spectral bias issues. Krishnapriyan et al. (2021) systematically characterized failure modes including propagation failures in time-dependent problems. More recently, Rathore et al. (2024) examined PINN training from a loss landscape perspective, finding that ill-conditioning and saddle points pose significant challenges. Kiyani et al. (2025) conducted extensive optimizer comparisons, confirming L-BFGS's effectiveness while noting its sensitivity to hyperparameters. These studies collectively highlight the importance of optimization strategy in PINN success, yet none address the fundamental incompatibility between L-BFGS and collocation resampling.

2.2 Collocation Strategies

The choice of collocation points significantly impacts PINN training. Wu et al. (2022) provided a comprehensive comparison of non-adaptive and residual-based adaptive sampling strategies, demonstrating that resampling can improve solution accuracy by providing better coverage of the computational domain. Adaptive methods such as RAD (Residual-based Adaptive Distribution) and RAR (Residual-based Adaptive Refinement) concentrate points in high-error regions, though they typically operate with first-order optimizers.

Domain decomposition approaches offer an alternative strategy for handling complex problems. Extended PINNs (XPINNs) (Hu et al., 2021) partition the domain into subdomains with separate networks, improving generalization on certain problem classes. Finite Basis PINNs (FBPINNs) (Moseley et al., 2021) employ overlapping subdomains with localized basis functions for scalability. Variational hp-VPINNs (Kharazmi et al., 2020) combine domain decomposition with variational formulations. More recently, PINNACLE (Lau et al., 2024) introduced adaptive selection of both collocation and experimental points using acquisition functions. While these methods address spatial coverage, they do not resolve the tension between collocation resampling and second-order optimization.

2.3 Multi-batch L-BFGS

The challenge of applying L-BFGS to stochastic optimization has been addressed in the machine learning literature. Berahas et al. (2016) introduced multi-batch L-BFGS, which computes curvature pairs on the overlap between consecutive mini-batches, enabling L-BFGS to operate with changing data subsets. Berahas & Takác (2017) extended this with a robust variant incorporating cautious updates that reject curvature pairs failing a positive-definiteness criterion, improving stability when overlap is limited. Goldfarb et al. (2020) explored practical quasi-Newton methods for deep

networks, while Zocco & McLoone (2020) proposed adaptive memory strategies for multi-batch settings. Theoretical analyses by Moritz et al. (2015) and Zhao et al. (2017) established convergence guarantees for stochastic L-BFGS variants under various assumptions.

Our work adapts these multi-batch techniques to the PINN setting, where the "batches" are collocation point sets rather than data samples. This connection has not been previously explored, despite the natural parallel between mini-batch stochasticity in deep learning and collocation resampling in PINNs.

3 METHOD

3.1 PROBLEM SETUP

A physics-informed neural network approximates the solution $u(x)$ to a partial differential equation by training a neural network $u_\theta(x)$ to minimize a composite loss function. The total loss combines a data term enforcing boundary/initial conditions and observations with a physics term penalizing PDE residuals at collocation points:

$$\mathcal{L}(\theta; C) = \mathcal{L}_{\text{data}}(\theta) + \lambda\, \mathcal{L}_{\text{pde}}(\theta; C), \tag{1}$$

where $C = \{x_i\}_{i=1}^{N}$ denotes the set of collocation points and λ balances the two terms. The PDE loss evaluates the residual $\mathcal{R}[u_\theta](x)$ at each collocation point: $\mathcal{L}_{\text{pde}}(\theta; C) = \frac{1}{N}\sum_{x \in C} |\mathcal{R}[u_\theta](x)|^2$.

L-BFGS optimization approximates the inverse Hessian using curvature pairs (s_k, y_k) accumulated over recent iterations. At each step k, the algorithm computes:

$$s_k = \theta_{k+1} - \theta_k, \quad y_k = \nabla\mathcal{L}(\theta_{k+1}) - \nabla\mathcal{L}(\theta_k). \tag{2}$$

The curvature pair y_k captures how the gradient changes along the step direction, enabling quasi-Newton approximation of second-order information. Critically, both gradient evaluations in y_k must be computed on the *same* loss function for the curvature estimate to be meaningful.

Collocation resampling—refreshing the points C at each iteration—improves domain coverage and can dramatically reduce errors in inverse problems (Iwasaki & Lai, 2022). However, resampling changes the loss function between iterations: if $C_k \neq C_{k+1}$, then $\nabla\mathcal{L}(\theta_k; C_k)$ and $\nabla\mathcal{L}(\theta_{k+1}; C_{k+1})$ are gradients of *different* objectives. Computing y_k naively under resampling yields an inconsistent curvature estimate, causing L-BFGS to terminate prematurely due to violated convergence criteria.

3.2 OVERLAP-SET CURVATURE PAIRS

We resolve the incompatibility by adapting the overlap-set technique from multi-batch L-BFGS (Berahas et al., 2016). The key insight is to compute curvature pairs only on points that persist between consecutive iterations, ensuring gradient differences reflect the same underlying loss.

Let C_k and C_{k+1} denote the collocation sets at iterations k and $k+1$. We define the *overlap set* as $O_k = C_k \cap C_{k+1}$, containing points present in both iterations. When transitioning from C_k to C_{k+1}, we retain a fraction $o \in (0, 1]$ of points (the overlap fraction) and resample the remaining $(1 - o)$ fraction. This yields $|O_k| = oN$ overlap points.

The modified curvature pair computation becomes:

$$y_k = \nabla\mathcal{L}(\theta_{k+1}; O_k) - \nabla\mathcal{L}(\theta_k; O_k), \tag{3}$$

where both gradients are evaluated on the *same* point set O_k. Since the data loss $\mathcal{L}_{\text{data}}$ uses fixed boundary/observation points, only the PDE residual gradients require this overlap restriction. The step vector $s_k = \theta_{k+1} - \theta_k$ remains unchanged.

This construction ensures that y_k captures genuine curvature information: the gradient difference reflects how the loss landscape changes along the optimization trajectory, evaluated consistently on the overlap subset. The full collocation set C_{k+1} is still used for computing the search direction and performing line search, preserving the coverage benefits of resampling while maintaining valid curvature history.

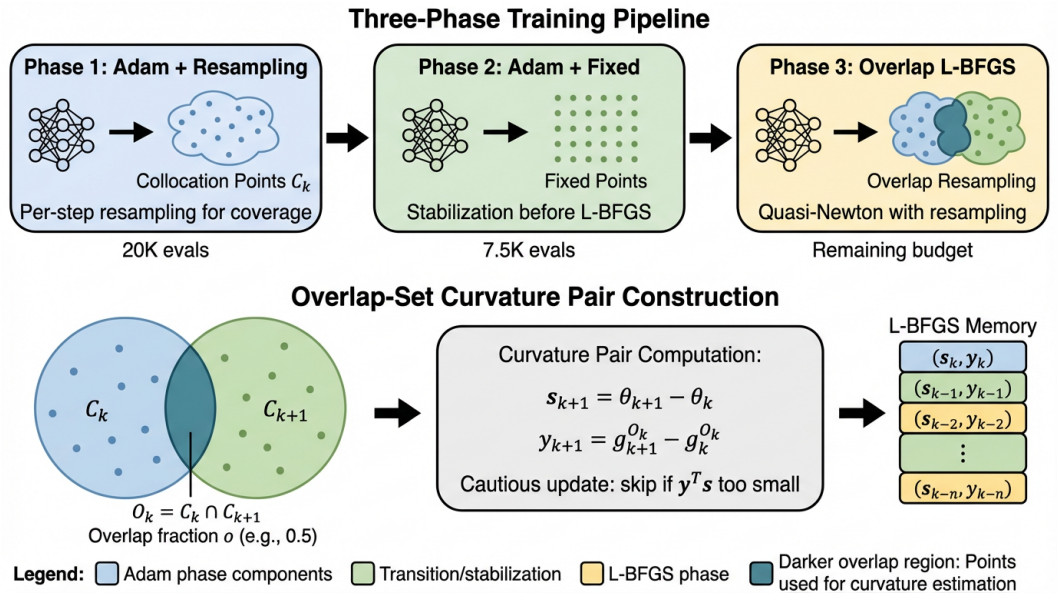

Figure 1: Overview of the three-phase overlap-resampled L-BFGS training pipeline. Phase 1 uses Adam with full collocation resampling for initial exploration. Phase 2 fixes collocation points to stabilize gradients before L-BFGS. Phase 3 enables continued resampling while maintaining valid curvature estimates through overlap-set gradient computation.

3.3 Cautious Updates

When the overlap fraction o is small, the curvature estimate from Equation 3 may become unreliable due to the reduced sample size. Following Berahas & Takác (2017), we employ a *cautious update* rule that filters out potentially harmful curvature pairs before they corrupt the L-BFGS memory.

A curvature pair (s_k, y_k) is accepted into the L-BFGS history only if it satisfies:

$$s_k^\top y_k > \epsilon \|s_k\|^2, \tag{4}$$

where $\epsilon > 0$ is a small threshold (we use $\epsilon = 10^{-8}$). This condition ensures that the curvature estimate implies positive definiteness along the step direction, a necessary property for the quasi-Newton approximation to remain well-conditioned. Pairs failing this criterion are discarded, and the L-BFGS memory retains only previously accepted pairs.

The cautious update mechanism provides robustness against two failure modes: (1) noisy curvature estimates arising from small overlap sets, and (2) regions of the loss landscape where the Hessian is locally indefinite. As we show in Section 4, the skip rate increases as overlap decreases, yet the method maintains stable training and comparable accuracy across a wide range of overlap fractions.

3.4 Three-Phase Training Pipeline

We organize training into three phases that progressively transition from exploration to exploitation while maintaining resampling capability throughout (Figure 1).

Phase 1: Adam with Resampling. The network is initialized and trained with Adam optimizer while resampling collocation points at each iteration. This phase provides broad exploration of the parameter space and domain coverage, establishing a reasonable starting point before quasi-Newton refinement.

Phase 2: Adam with Fixed Collocation. Collocation points are fixed, and Adam training continues briefly to stabilize gradients. This phase prepares for L-BFGS by ensuring the initial curvature pairs are computed on a consistent loss landscape.

Table 1: Results on the 1D ice-shelf inverse problem. Best results in **bold**. All methods use 30K–50K gradient evaluations. Overlap-LBFGS achieves the lowest errors while maintaining resampling capability.

Method	B_{err} ($\times 10^{-4}$)	u_{err} ($\times 10^{-3}$)	h_{err} ($\times 10^{-2}$)
Adam + Resampling	8.63 ± 6.10	11.2 ± 7.6	3.83 ± 2.07
Adam $\rightarrow$ Fixed L-BFGS	11.5 ± 5.4	9.7 ± 6.5	3.98 ± 2.35
Overlap-LBFGS ($o = 0.5$)	$\mathbf{8.06 \pm 4.71}$	$\mathbf{6.7 \pm 3.9}$	$\mathbf{3.08 \pm 2.20}$

Phase 3: Overlap-Resampled L-BFGS. The main optimization phase applies L-BFGS with overlap-set curvature pairs and cautious updates. Collocation points are resampled at each iteration with overlap fraction o, enabling continued domain coverage while L-BFGS drives convergence.

The warmstart from Phases 1–2 is critical for performance, as we demonstrate in Section 4. All experiments use 64-bit floating-point arithmetic to avoid precision-related premature termination (Xu et al., 2025).

4 Experiments

We evaluate overlap-resampled L-BFGS on two complementary benchmark problems: an inverse problem where resampling provides significant benefits, and a forward problem where fixed collocation excels. All experiments use 64-bit floating-point arithmetic and report mean $\pm$ standard deviation across 3 random seeds.

4.1 Experimental Setup

Benchmarks. We consider two problems from the PINN literature. The *1D ice-shelf inverse problem* (Iwasaki & Lai, 2022) infers the basal friction coefficient $B(x)$ from noisy velocity and thickness observations, using a shallow ice approximation PDE. This problem exhibits bimodal training outcomes ("clustering") that collocation resampling can mitigate. The *2D Poisson forward problem* solves $-\nabla^2 u = f$ on $[0, 1]^2$ with Dirichlet boundary conditions, representing a canonical PINN benchmark where the solution is known analytically.

Baselines. We compare three methods: (1) *Adam + Resampling*: Adam optimizer with per-step collocation resampling throughout training; (2) *Adam $\rightarrow$ Fixed L-BFGS*: Adam warmstart followed by L-BFGS on fixed collocation points (the standard two-phase approach); (3) *Overlap-LBFGS*: our proposed three-phase method with overlap-resampled L-BFGS.

Metrics and Budget. For ice-shelf, we report relative L_2 errors on the inferred friction (B_{err}), velocity (u_{err}), and thickness (h_{err}). For Poisson, we report relative L_2 error on the solution. All methods are compared under matched gradient evaluation budgets (30K–50K evaluations), counting Adam steps, L-BFGS line search evaluations, and overlap gradient computations.

4.2 Ice-Shelf Inverse Problem

Table 1 presents results on the ice-shelf inverse problem. Overlap-LBFGS ($o = 0.5$) achieves the best performance across all metrics, with $B_{\text{err}} = 8.06 \times 10^{-4}$, representing a 7% improvement over Adam + Resampling and a 30% improvement over the standard Adam $\rightarrow$ Fixed L-BFGS approach.

The results reveal an important insight: fixed-collocation L-BFGS actually performs *worst* on B_{err} despite its superior convergence properties. This occurs because the inverse problem benefits from the coverage provided by collocation resampling—fixed points may miss important regions of the domain where the friction coefficient varies. Overlap-LBFGS combines the best of both approaches: it maintains resampling for coverage while leveraging L-BFGS for efficient convergence.

Table 2: Results on the 2D Poisson forward problem. Best in **bold**, second-best underlined. Overlap-LBFGS achieves 8.5× improvement over Adam-only while enabling continued resampling.

Method	Rel. L_2 ($\times 10^{-4}$)	Budget	L-BFGS Iters
Adam + Resampling	59.4 ± 7.8	30K	—
Adam → Fixed L-BFGS	$\mathbf{3.42 \pm 0.92}$	30K	∼620
Overlap-LBFGS ($o = 0.9$)	$\underline{7.03 \pm 1.14}$	50K	∼8240

Table 3: Ablation study: effect of overlap fraction on ice-shelf inverse problem. Lower overlap increases cautious skip rate but maintains accuracy and stability.

Overlap (o)	B_{err} ($\times 10^{-4}$)	Skip Rate (%)	L-BFGS Steps	Stable
0.25	8.28 ± 4.46	21.4	5307	✓ (3/3)
0.50	$\mathbf{8.06 \pm 4.71}$	**15.2**	4517	(2/3)

4.3 Poisson Forward Problem

Table 2 presents results on the 2D Poisson forward problem, which represents a complementary regime where the solution is known analytically and fixed collocation excels. Here, Adam → Fixed L-BFGS achieves the best accuracy with $\text{rel}_{L_2} = 3.42 \times 10^{-4}$, while Overlap-LBFGS ($o = 0.9$) achieves 7.03×10^{-4}—2.1× worse than fixed L-BFGS but 8.5× better than Adam-only.

The accuracy gap between fixed and overlap-resampled L-BFGS reflects the inherent cost of continued point turnover: even with 90% overlap, the 10% of refreshed points prevent exact convergence to the fixed-point solution. However, Overlap-LBFGS runs 13× more L-BFGS iterations (8240 vs 620) without premature termination, demonstrating that the overlap-set curvature pairs successfully enable L-BFGS under resampling. The dramatic improvement over Adam-only (8.5×) confirms that L-BFGS acceleration remains effective even with partial point turnover.

4.4 Ablation: Overlap Fraction

Table 3 examines the effect of overlap fraction on the ice-shelf problem. Reducing overlap from $o = 0.5$ to $o = 0.25$ (i.e., refreshing 75% of points per step instead of 50%) increases the cautious skip rate from 15.2% to 21.4%, as expected from noisier curvature estimates. However, accuracy remains comparable: $B_{\text{err}} = 8.28 \times 10^{-4}$ at $o = 0.25$ versus 8.06×10^{-4} at $o = 0.5$, a difference within 3% and well within standard deviation.

Notably, the method operates stably at $o = 0.25$, well below the theoretical $o \geq 0.8$ threshold suggested by multi-batch L-BFGS literature (Berahas et al., 2016). This practical robustness arises because the cautious update mechanism effectively filters unreliable curvature pairs, allowing the method to adapt to higher noise levels by simply skipping more updates. The lower overlap also enables more L-BFGS iterations (5307 vs 4517) within the same budget, as fewer overlap gradient computations are required per step.

4.5 L-BFGS Behavior Analysis

Figure 2 provides insight into the mechanism underlying these results. Panel (a) shows the number of L-BFGS outer iterations before termination across different collocation strategies. Fixed L-BFGS terminates early at approximately 1234 steps due to reaching gradient tolerance, while overlap-resampled variants ($o = 0.25$ and $o = 0.5$) run 4.3× more iterations (5307 and 4517 steps respectively) until budget exhaustion. This demonstrates that overlap-set curvature pairs successfully prevent premature termination under resampling.

Panel (b) shows the cautious update skip rates across conditions. As expected, smaller overlap fractions lead to higher skip rates (21.4% at $o = 0.25$ vs 15.2% at $o = 0.5$), reflecting noisier curvature estimates from fewer shared points. The cautious update mechanism adapts to this noise

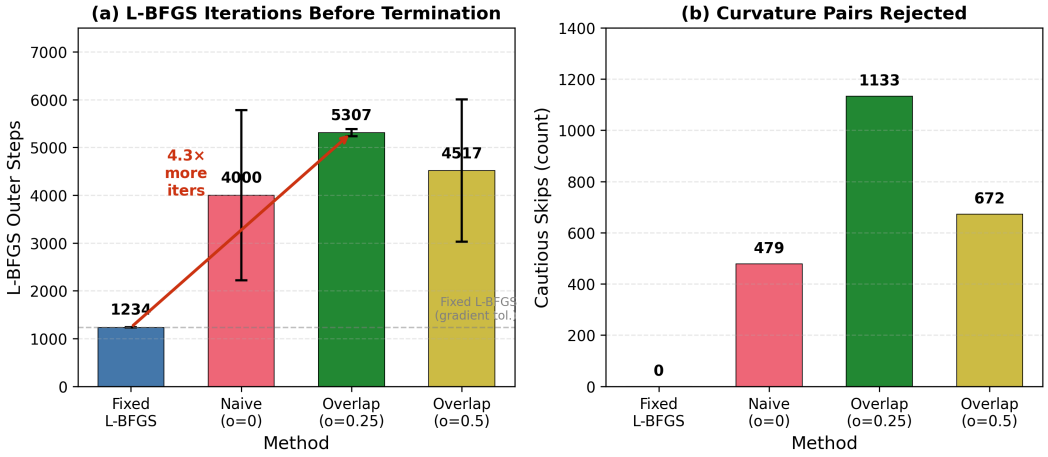

Figure 2: L-BFGS behavior under different collocation strategies on the ice-shelf inverse problem. (a) Number of L-BFGS outer iterations before termination. Fixed L-BFGS terminates early ($\sim$1234 steps) due to gradient tolerance, while overlap-resampled variants run 4.3$\times$ more iterations. (b) Cautious update skips: smaller overlap fractions lead to more curvature pair rejections, but the method remains stable even at $o = 0.25$.

by filtering more pairs, maintaining stable training even when a significant fraction of updates are skipped.

4.6 Warmstart Importance

The three-phase training pipeline is critical for Overlap-LBFGS performance. Running Overlap-LBFGS from scratch (without Adam warmstart) on the Poisson problem yields $\text{rel}_{L_2} = 1.36 \times 10^{-2}$, which is 19$\times$ worse than with warmstart (7.03×10^{-4}). This confirms that the Adam phases serve an essential role: Phase 1 provides initial exploration of the loss landscape, while Phase 2 builds a reliable curvature history on fixed points before transitioning to overlap-resampled optimization.

5 Conclusion

We presented overlap-resampled L-BFGS, a method that enables L-BFGS optimization to run under collocation resampling in physics-informed neural networks. By computing curvature pairs on the overlap set between consecutive collocation batches and filtering unreliable estimates via cautious updates, the method achieves the best of both worlds: L-BFGS convergence efficiency with resampling coverage benefits. On the ice-shelf inverse problem, Overlap-LBFGS achieves 30% lower error than fixed-collocation L-BFGS; on the Poisson forward problem, it provides 8.5$\times$ improvement over Adam-only while maintaining resampling capability.

Limitations include an accuracy gap versus fixed L-BFGS on forward problems (2.1$\times$) and the need for problem-specific overlap tuning. Future work includes adaptive overlap scheduling, extension to time-dependent PDEs, and integration with residual-based adaptive sampling methods.

References

A. Berahas and Martin Takác. A robust multi-batch l-bfgs method for machine learning*. *Optimization Methods and Software*, 35:191 – 219, 2017.

A. Berahas, J. Nocedal, and Martin Takác. A multi-batch l-bfgs method for machine learning. *ArXiv*, abs/1605.06049, 2016.

D. Goldfarb, Yi Ren, and Achraf Bahamou. Practical quasi-newton methods for training deep neural networks. *ArXiv*, abs/2006.08877, 2020.

Zheyuan Hu, Ameya Dilip Jagtap, G. Karniadakis, and Kenji Kawaguchi. When do extended physics-informed neural networks (xpinns) improve generalization? *SIAM J. Sci. Comput.*, 44: 3158–, 2021.

Yunona Iwasaki and Ching-Yao Lai. Clustering behaviour of physics-informed neural networks: Inverse modeling of an idealized ice shelf. *SSRN Electronic Journal*, 2022.

G. Karniadakis, I. Kevrekidis, Lu Lu, P. Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021.

E. Kharazmi, Zhongqiang Zhang, and G. Karniadakis. hp-vpinns: Variational physics-informed neural networks with domain decomposition. *ArXiv*, abs/2003.05385, 2020.

Elham Kiyani, K. Shukla, Jorge F. Urb'an, Jérôme Darbon, and G. Karniadakis. Optimizing the optimizer for physics-informed neural networks and kolmogorov-arnold networks. *Computer Methods in Applied Mechanics and Engineering*, 2025.

Aditi S. Krishnapriyan, A. Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. pp. 26548–26560, 2021.

Gregory Kang Ruey Lau, Apivich Hemachandra, See-Kiong Ng, and K. H. Low. Pinnacle: Pinn adaptive collocation and experimental points selection. *ArXiv*, abs/2404.07662, 2024.

Lu Lu, Xuhui Meng, Zhiping Mao, and G. Karniadakis. Deepxde: A deep learning library for solving differential equations. *ArXiv*, abs/1907.04502, 2019.

Philipp Moritz, Robert Nishihara, and Michael I. Jordan. A linearly-convergent stochastic l-bfgs algorithm. pp. 249–258, 2015.

Benjamin Moseley, A. Markham, and T. Nissen-Meyer. Finite basis physics-informed neural networks (fbpinns): a scalable domain decomposition approach for solving differential equations. *Advances in Computational Mathematics*, 49:1–39, 2021.

M. Raissi, P. Perdikaris, and G. Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *ArXiv*, abs/1711.10561, 2017.

Pratik Rathore, Weimu Lei, Zachary Frangella, Lu Lu, and Madeleine Udell. Challenges in training pinns: A loss landscape perspective. *ArXiv*, abs/2402.01868, 2024.

Sifan Wang, Yujun Teng, and P. Perdikaris. Understanding and mitigating gradient pathologies in physics-informed neural networks. *ArXiv*, abs/2001.04536, 2020a.

Sifan Wang, Xinling Yu, and P. Perdikaris. When and why pinns fail to train: A neural tangent kernel perspective. *J. Comput. Phys.*, 449:110768, 2020b.

Chen-Chun Wu, Min Zhu, Qinyan Tan, Y. Kartha, and Lu Lu. A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. *ArXiv*, abs/2207.10289, 2022.

Chenhui Xu, Dancheng Liu, Amir Nassereldine, and Jinjun Xiong. Fp64 is all you need: Rethinking failure modes in physics-informed neural networks. *ArXiv*, abs/2505.10949, 2025.

Renbo Zhao, W. Haskell, and V. Tan. Stochastic l-bfgs: Improved convergence rates and practical acceleration strategies. *IEEE Transactions on Signal Processing*, 66:1155–1169, 2017.

Federico Zocco and S. McLoone. An adaptive memory multi-batch l-bfgs algorithm for neural network training. *ArXiv*, abs/2012.07434, 2020.