Good Lattice Accelerates Physics-Informed Neural Networks

Takashi Matsubara [1] **Takaharu Yaguchi** [2]

Abstract

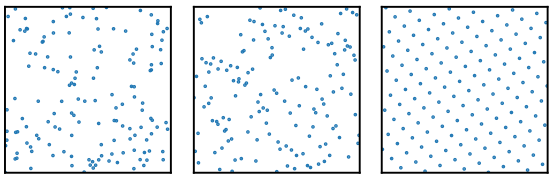
Physics-informed neural networks (PINNs) can solve partial differential equations (PDEs) by minimizing the physics-informed loss, ensuring the neural network satisfies the PDE at given points. However, the solutions to a PDE are infinite-dimensional, and the physics-informed loss is a finite approximation to a certain integral over the domain. This indicates that selecting appropriate points is essential. This paper proposes "good lattice training" (GLT), a technique inspired by number theoretic methods. GLT provides an optimal set of collocation points and can train PINNs to achieve competitive performance with smaller computational cost.

1. Introduction

Many real-world phenomena can be represented by partial differential equations (PDEs), and solving PDEs is an important topic in computational science. A PDE is defined as $\mathcal{N}[u] = 0$, where $\mathcal{N}$ is a (possibly nonlinear) differential operator, $u : \Omega \to \mathbb{R}$ is an unknown function, and $\Omega \subset \mathbb{R}^s$ is the domain. Various computational methods have been developed for solving PDEs. However, due to limited progress in computer architecture development, there is a need for computationally efficient alternatives. One promising approach is physics-informed neural networks (PINNs)(Raissi et al., 2019), which train a neural network to satisfy the PDE $\mathcal{N}[\tilde{u}] = 0$. Specifically, PINNs minimize the squared error $\|\mathcal{N}[\tilde{u}](\boldsymbol{x}_i)\|^2$ at a finite set of collocation points $\boldsymbol{x}_i$ to ensure that the output $\tilde{u}$ satisfies the equation $\mathcal{N}[\tilde{u}](\boldsymbol{x}_i) = 0$. This objective, known as the physics-informed loss, is the core idea of PINNs (Wang et al., 2022; Wang & Perdikaris, 2023).

[1]Graduate School of Engineering Science, Osaka University, Osaka, Japan [2]Graduate School of System Informatics, Kobe University, Hyogo, Japan. Correspondence to: Takashi Matsubara <matsubara@sys.es.osaka-u.ac.jp>.

Accepted after peer-review at the 1st workshop on Synergy of Scientific and Machine Learning Modeling, SynS & ML ICML, Honolulu, Hawaii, USA. July, 2023.



uniformly random LHS proposed GLT

Figure 1. Examples of 144 sampled collocation points.

However, the solutions u to PDEs are infinite-dimensional, and the distance involving $\tilde{u}$ or u must be defined by an integral over the domain Ω. In this sense, the physics-informed loss provides a finite approximation to the squared 2-norm $|\mathcal{N}[\tilde{u}]|_2^2 = \int_\Omega \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_i)\|^2 \mathrm{d}\boldsymbol{x}$ in the function space $L^2(\Omega)$, and the discretization errors affect the training efficiency. Using a smaller number N of collocation points reduces accuracy and performance, while a larger number N increases computational cost (Bihlo & Popovych, 2022; Sharma & Shankar, 2022). Hence, selecting appropriate collocation points is crucial. Previous studies have used Latin hypercube sampling (LHS) to determine collocation points (Raissi et al., 2019; Zeng et al., 2023) as well as uniformly random sampling (Jin et al., 2021; Krishnapriyan et al., 2022). Figure 1 illustrates these methods.

In the field of numerical analysis, the relationship between integral approximation and collocation points has been extensively studied. Inspired by number theoretic methods for numerical analysis, we propose *good lattice training (GLT)* for PINNs. GLT provides an optimal set of collocation points, as shown in Fig. 1, when the activation functions of the neural networks are sufficiently smooth. Our experiments demonstrate that GLT requires significantly fewer collocation points compared to other methods while achieving similar errors, leading to a substantial reduction in computational cost.

Related Work Neural networks have attracted attention as computational tools for solving differential equations, particularly PDEs (Dissanayake & Phan-Thien, 1994; Lagaris et al., 1998). Recently, a refinement to this approach called PINNs was introduced by Raissi et al. (2019). The core idea of PINNs is the physics-informed loss (Wang et al., 2022; Wang & Perdikaris, 2023). This loss function measures how well the output $\tilde{u}$ of the neural network satisfies a given PDE

$\mathcal{N}[u] = 0$ along with its associated initial and boundary conditions $\mathcal{B}[u] = 0$. The physics-informed loss has also been incorporated into other models such as DeepONet (Lu et al., 2019; Wang & Perdikaris, 2023).

PINNs have been applied to various PDEs in recent studies (Bihlo & Popovych, 2022; Jin et al., 2021; Mao et al., 2020), and significant efforts have been made to improve learning algorithms and objective functions (Hao et al., 2023; Heldmann et al., 2023; Lu et al., 2022; Pokkunuru et al., 2023; Sharma & Shankar, 2022; Zeng et al., 2023). In these studies, the loss functions based on the PDE itself, rather than data, including the original physics-informed loss, are defined by evaluating errors at a finite set of collocation points. As shown in previous works (Bihlo & Popovych, 2022; Sharma & Shankar, 2022), there is a trade-off between the number of collocation points (and hence computational cost) and the accuracy of the solution. Therefore, selecting collocation points that effectively cover the entire domain is crucial for obtaining better results, although this aspect has often been overlooked. An exception is found in the work of Mao et al. (2020), where additional collocation points were sampled around regions with discontinuities to capture them efficiently. However, such an approach is not always feasible.

2. Method

2.1. Motivation

For simplicity, we consider the cases where the target PDE is defined on an s-dimensional unit cube $[0,1]^s$. Given a set of collocation points $\{\boldsymbol{x}_j \mid j = 1, \dots, N\}$, the physics-informed loss is defined as

$$\frac{1}{N}\sum_{j=1}^{N} \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_j)\|^2 \tag{1}$$

$\tilde{u}$ is an approximate solution represented by a neural network. However, for $\tilde{u}$ to become the exact solution, this physics-informed loss should be 0 for all $\boldsymbol{x}$, not just at collocation points. Therefore, the loss function defined by the following integral must be minimized:

$$\int_{[0,1]^s} \|\mathcal{N}[\tilde{u}](\boldsymbol{x})\|^2 \mathrm{d}\boldsymbol{x}. \tag{2}$$

In other words, the practical minimization of (1) essentially minimizes the approximation of (2) with the expectation that (2) will be small enough and hence $\tilde{u}$ becomes an accurate approximation to the exact solution.

2.2. Good Lattice Training

Definition 2.1 (Niederreiter (1992); Sloan & Joe (1994); Zaremba (2014))**.** A lattice L in $\mathbb{R}^s$ is defined as a finite set of points in $\mathbb{R}^s$ that is closed under addition and subtraction.

Given a lattice L, the set of collocation points is defined as $L \cap [0,1]^s =: \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$. In this paper, we consider an appropriate lattice for computing the physics-informed loss. Considering that the loss function to be minimized is (2), it is natural to determine the lattice L (the set of collocation points $\boldsymbol{x}_j$'s) so that the difference $|(2) - (1)|$ of the two loss functions is minimized.

Suppose that $\varepsilon(\boldsymbol{x}) \coloneqq \|\mathcal{N}[\tilde{u}](\boldsymbol{x})\|^2$ is smooth enough and admits the Fourier series expansion:

$$\varepsilon(\boldsymbol{x}) \coloneqq \|\mathcal{N}[\tilde{u}](\boldsymbol{x})\|^2 = \textstyle\sum_{\boldsymbol{h}} \hat{\varepsilon}(\boldsymbol{h}) \exp(2\pi \mathrm{i} \boldsymbol{h} \cdot \boldsymbol{x}),$$

where i denotes the imaginary unit and $\boldsymbol{h} = (h_1, h_2, \dots, h_s) \in \mathbb{Z}^s$. Substitution of this Fourier series yields

$$|(2) - (1)| = |\tfrac{1}{N}\textstyle\sum_{j=1}^{N} \sum_{\boldsymbol{h}\in\mathbb{Z}^s, \boldsymbol{h}\neq 0} \hat{\varepsilon}(\boldsymbol{h}) \exp(2\pi \mathrm{i} \boldsymbol{h} \cdot \boldsymbol{x}_j)| \tag{3}$$

where the equality follows from the fact that the Fourier mode of $\boldsymbol{h} = 0$ is equal to the integral $\int_{[0,1]^s} \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_j)\|^2 \mathrm{d}\boldsymbol{x}$. In addition, the dual lattice of lattice L and an insightful lemma are introduced:

Definition 2.2 (Niederreiter (1992); Sloan & Joe (1994); Zaremba (2014))**.** A dual lattice $L^\top$ of a lattice L is defined as $L^\top \coloneqq \{\boldsymbol{h} \in \mathbb{R}^s \mid \boldsymbol{h} \cdot \boldsymbol{x} \in \mathbb{Z},\ {}^\forall\boldsymbol{x} \in L\}$.

Lemma 2.3 (Niederreiter (1992); Sloan & Joe (1994); Zaremba (2014))**.** *For $\boldsymbol{h} \in \mathbb{Z}^s$, it holds that*

$$\tfrac{1}{N}\textstyle\sum_{j=1}^{N} \exp(2\pi \mathrm{i} \boldsymbol{h} \cdot \boldsymbol{x}_j) = \begin{cases} 1 & (\boldsymbol{h} \in L^\top) \\ 0 & (\text{otherwise.}) \end{cases}$$

Based on this lemma, we restrict collocation points $\boldsymbol{x}_j$ to the form $L = \{\text{decimal part of } \frac{j}{N}\boldsymbol{z} \mid j = 0, \dots, N-1\}$ with a fixed integer vector $\boldsymbol{z}$. We consider to search a vector $\boldsymbol{z}$ instead of $\boldsymbol{x}_j$'s. Then, the optimal collocation points $\boldsymbol{x}_j$'s do not need to be stored as a table of numbers. Then, (3) is upper-bounded by

$$\textstyle\sum_{\boldsymbol{h}\in\mathbb{Z}^s, \boldsymbol{h}\neq 0, \boldsymbol{h}\cdot\boldsymbol{z}\equiv 0 \pmod N} |\hat{\varepsilon}(\boldsymbol{h})|, \tag{4}$$

and hence the collocation points $\boldsymbol{x}_j$'s should be determined so that (4) becomes small.

In particular, optimal solutions have been investigated for integrands in the Korobov spaces, which are spaces of functions that satisfy a certain smoothness condition.

Definition 2.4 (Niederreiter (1992); Sloan & Joe (1994); Zaremba (2014))**.** The function space that is defined as $E_\alpha = \{f : [0,1]^s \to \mathbb{R} \mid {}^\exists c, |\hat{f}(\boldsymbol{h})| \le c/(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha\}$ is called the Korobov space, where $\hat{f}(\boldsymbol{h})$ is the Fourier coefficients of f and $\bar{k} = \max(1, |k|)$ for $k \in \mathbb{R}$.

It is known that if α is an integer, for a function f to be in E_α, it is sufficient that f has continuous partial derivatives $\partial^{q_1+q_2+\cdots+q_s} f / \partial_1^{q_1} \cdot \partial_2^{q_2} \cdots \partial_s^{q_s}, 0 \le q_k \le \alpha\ (k = 1, \dots, s)$. For example, if a function $f(x, y) : \mathbb{R}^2 \to \mathbb{R}$ has continuous f_x, f_y, f_{xy}, then $f \in E_1$.

We also introduce the following theorem.

Theorem 2.5 (Sloan & Joe (1994)). *If $N \geq 2$ is any integer and $s \geq 2$, there exists a $\boldsymbol{z} \in \mathbb{Z}^s$ such that*

$$P_\alpha(\boldsymbol{z}, N) \leq \frac{(2\log N)^{\alpha s}}{N^\alpha} + O\left(\frac{(\log N)^{\alpha s - 1}}{N^\alpha}\right),$$

where

$$P_\alpha(\boldsymbol{z}, N) = \sum_{\boldsymbol{h}\cdot\boldsymbol{z} \equiv 0 \pmod N} \frac{1}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha}.$$

The main result of this paper is the following.

Theorem 2.6. *Suppose that the activation function of $\tilde{u}$ and hence $\tilde{u}$ itself are sufficiently smooth so that there exists an $\alpha > 0$ such that $\|\mathcal{N}[\tilde{u}]\|^2 \in E_\alpha$. Then, for given integers $N \geq 2$ and $s \geq 2$, there exists an integer vector $\boldsymbol{z} \in \mathbb{Z}^s$ such that $L = \{\text{decimal part of } \frac{j}{N}\boldsymbol{z} \mid j = 0, \ldots, N-1\}$ is a "good lattice" in the sense that*

$$\begin{aligned} &\int_{[0,1]^s} \|\mathcal{N}[\tilde{u}](\boldsymbol{x})\|^2 \mathrm{d}\boldsymbol{x} \\ &\quad \leq \frac{1}{N}\sum_{\boldsymbol{x}_j \in L} \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_j)\|^2 + O\left(\frac{(\log N)^{\alpha s}}{N^\alpha}\right). \end{aligned} \tag{5}$$

The proof is as follows. When $\|\mathcal{N}[\tilde{u}]\|^2 \in E_\alpha$, Because of Definition 2.4 and (3), there exists a $\boldsymbol{z} \in \mathbb{Z}^s$ such that

$$|(2) - (1)| \leq \sum_{\boldsymbol{h} \in \mathbb{Z}^s, \boldsymbol{h} \neq 0, \boldsymbol{h}\cdot\boldsymbol{z} \equiv 0 \pmod N} \frac{c}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha}$$

From Theorem 2.5, there exists a $\boldsymbol{z} \in \mathbb{Z}^s$ such that

$$|(2) - (1)| \leq c P_\alpha(\boldsymbol{z}, N)$$

and hence

$$(2) \leq (1) + O\left(\frac{(\log N)^{\alpha s}}{N^\alpha}\right).$$

In this paper, we call the learning method that minimizes

$$\frac{1}{N}\sum_{\boldsymbol{x}_j \in L} \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_j)\|^2$$

for a lattice L that satisfies (5) *the good lattice training (GLT)*. The proposed GLT, of which the order of the convergence is $O(\frac{(\log N)^{\alpha s}}{N^\alpha})$, can converge much faster to the integrated loss (2) than the uniformly random sampling (i.e., the Monte Carlo method, of which the order of the convergence is $O(1/\sqrt{N})$).

2.3. Implementation and Practice

One can take the integer vector $\boldsymbol{z}$ from numerical tables. For example, when $N = 13$, $\boldsymbol{z} = (1, 8)$ gives a good lattice $L = \{(0,0), (\frac{1}{13}, \frac{8}{13}), (\frac{2}{13}, \frac{3}{13}), (\frac{3}{13}, \frac{11}{13}), \ldots\}$.

The lattice can be shifted periodically. Therefore, the physics-informed loss based on GLT is as follows.

$$\frac{1}{N}\sum_{j=0}^{N-1} \left\|\mathcal{N}[\tilde{u}]\left(\left\{\frac{\boldsymbol{z}}{N} + \boldsymbol{r}\right\}\right)\right\|^2, \ \boldsymbol{r} \sim U([0,1])^s, \tag{6}$$

where $\{\cdot\}$ gives the decimal part of each element of the input, and $U([0,1]^s)$ is the uniform distribution over the interval $[0,1]^s$. Our preliminary experiments confirmed that, if using the stochastic gradient descent (SGD) algorithms, resampling the random numbers $\boldsymbol{r}$ at each training iteration prevents the neural network from overfitting and improves training efficiency.

Because GLT is based on the Fourier series expansion, it is basically assumed that the loss function is periodic. This problem can be addressed by variable transformations (Sloan & Joe, 1994). Moreover, our preliminary results demonstrated that GLT is effective even for non-periodic functions in practice.

For GLT to be effective, the neural network should be smooth, and as is well-known, this condition is satisfied if the activation function is smooth enough. However, if the true solution u is not a smooth function, the smoothness of the neural network $\tilde{u}$ is expected to decrease as $\tilde{u}$ approaches u. Actually, in such cases, although the differentiability of the activation function guarantees that $u \in E_\alpha$, the Fourier coefficients may increase. Therefore, the convergence in the final stage of training may be slowed down if the true solution is not smooth.

3. Experiments and Results

3.1. Experimental Settings

We modified the code from the official repository[1] of Raissi et al. (2019), the original paper of PINNs. We obtained the datasets of the nonlinear Schrödinger (NLS) and Korteweg–De Vries (KdV) equations. These datasets provide numerical solutions to initial value problems with periodic boundary conditions. Although they contain numerical errors, we treated them as the true solutions u.

Unless otherwise stated, we followed the repository's experimental settings for the NLS equation. The physics-informed loss was defined as $\mathcal{L}(\tilde{u}) = \frac{1}{N}\sum_{i=1}^{N} \|\mathcal{N}[\tilde{u}](\boldsymbol{x}_i)\|^2$ given N collocation points $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$. This can be regarded as a finite approximation to the squared 2-norm $|\mathcal{N}[\tilde{u}]|_2^2 = \int_\Omega \|\mathcal{N}[\tilde{u}](\boldsymbol{x})\|^2 \mathrm{d}\boldsymbol{x}$. We used other losses to learn the initial and boundary conditions for the NLS equation, while we used the technique in Lagaris et al. (1998) to ensure the conditions for KdV equations. The state of the NLS equation is complex; we simply treated it as a 2D real vector for training and used its absolute value for evaluation and visualization. We trained PINNs using the Adam optimizer with cosine decay of a single cycle to zero (Loshchilov & Hutter, 2017) for 200,000 iterations and sampling a different set of collocation points at each iteration; we found that this strategy improves the final performance. We evaluated the performance using the relative error, which is the normalized squared error $\mathcal{L}(\tilde{u}, u; \boldsymbol{x}_i) =$

[1] https://github.com/maziarraissi/PINNs

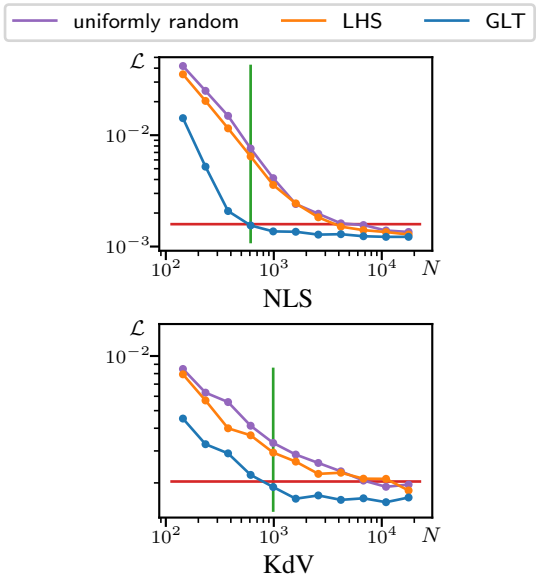


Figure 2. The number N of collocation points and relative error $\mathcal{L}$.

Table 1. Number of Points at Competitive Performance.

	NLS	KdV
• uniformly random	6,765	10,946
• LHS	4,181	17,711
• GLT (proposed)	**610**	**987**

Table 2. Relative Error at Competitive Number of Points

	NLS	KdV
• uniformly random	7.60×10^{-3}	3.32×10^{-3}
• LHS	6.45×10^{-3}	2.94×10^{-3}
• GLT (proposed)	$\mathbf{1.55}\times10^{-3}$	$\mathbf{1.90}\times10^{-3}$

$(\sum_{i=1}^{N_e} \|\tilde{u}(\boldsymbol{x}_i) - u(\boldsymbol{x}_i)\|^2)^{1/2}/(\sum_{i=1}^{N_e} \|u(\boldsymbol{x}_i)\|^2)^{1/2}$ at predefined N_e collocation points $\boldsymbol{x}_i$. This is also a finite approximation to $|\tilde{u} - u|_2/|u|_2$.

We determined the collocation points using uniformly random sampling, LHS, and the proposed GLT. For the GLT, we took the number N of collocation points and the corresponding integer vector $\boldsymbol{z}$ from numerical tables in (Fang & Wang, 1994; Keng & Yuan, 1981). To maintain consistency, we used the same value of N for both uniformly random sampling and LHS. We conducted five trials for each number N and each method.

3.2. Performance Results

Figure 2 shows the average relative error $\mathcal{L}$ with solid lines and the maximum and minimum errors with shaded areas. These results demonstrate that as N increases, the relative error $\mathcal{L}$ decreases and eventually reaches saturation. This is because of numerical errors in the datasets, discretization errors in relative error $\mathcal{L}$, and the network capacity.

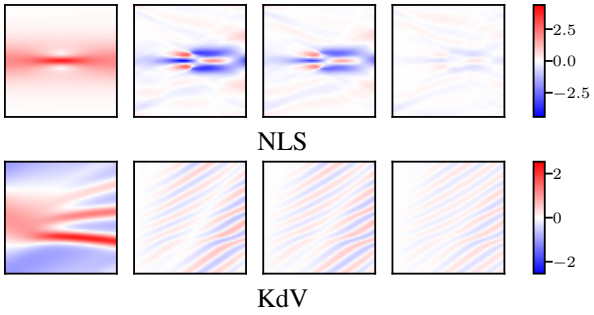


Figure 3. Example results with competitive numbers N of collocation points (on vertical green line in Fig. 2). The leftmost panel shows the true solution. The remaining panels show the residuals of PINNs' results multiplied by 100, obtained using uniformly random sampling, LHS, and GLT, from left to right.

We report the minimum numbers N of collocation points with which the relative error $\mathcal{L}$ was saturated in Table 1. Specifically, we consider a relative error $\mathcal{L}$ below 130 % of the minimum observed one as saturated; the thresholds are denoted by horizontal red lines in Fig. 2. The proposed GLT demonstrated equivalent performance with significantly fewer collocation points. This suggests that the proposed GLT can achieve comparable performance with significantly less computational cost. Next, we equalized the number N of collocation points (and therefore, the computational cost). In Table 2, we report the relative error $\mathcal{L}$ for collocation points with which the relative error $\mathcal{L}$ of one of the comparison methods saturated, as denoted by vertical green lines in Fig. 2. A smaller loss $\mathcal{L}$ indicates that the method performed better than others with the same computational cost. The proposed GLT yielded nearly half or less of the relative error $\mathcal{L}$. We show the true solutions and the residuals of example results with such N in Fig. 3.

4. Conclusion

This paper highlighted that the physics-informed loss, commonly used in PINNs and their variants, is a finite approximation to the integrated loss. From this perspective, we proposed good lattice training (GLT) to determine collocation points. This method enables a more accurate approximation of the integrated loss with a smaller number of collocation points. Experimental results demonstrated that the GLT can achieve competitive or superior performance with much fewer collocation points. These results imply a significant reduction in computational cost and contribute to the large-scale computation of PINNs.

This study was partially supported by JST CREST (JPMJCR1914), PRESTO (JPMJPR21C7), Moonshot R&D (JPMJMS2033), and JSPS KAKENHI (20K11693).

Broader Impact

This study accelerates the training of PINNs, and potentially motivates further studies that combine numerical analysis and deep learning.

References

Bihlo, A. and Popovych, R. O. Physics-informed neural networks for the shallow-water equations on the sphere. *Journal of Computational Physics*, 456:111024, May 2022. ISSN 0021-9991.

Dissanayake, M. W. M. G. and Phan-Thien, N. Neural-network-based approximations for solving partial differential equations. *Communications in Numerical Methods in Engineering*, 10(3):195–201, 1994. ISSN 1099-0887.

Fang, K.-t. and Wang, Y. *Number-Theoretic Methods in Statistics*. Series Title Monographs on Statistics and Applied Probability. Springer New York, NY, 1994.

Hao, Z., Ying, C., Su, H., Zhu, J., Song, J., and Cheng, Z. Bi-level Physics-Informed Neural Networks for PDE Constrained Optimization using Broyden's Hypergradients. In *International Conference on Learning Representations (ICLR)*, February 2023.

Heldmann, F., Berkhahn, S., Ehrhardt, M., and Klamroth, K. PINN training using biobjective optimization: The trade-off between data loss and residual loss. *Journal of Computational Physics*, pp. 112211, May 2023. ISSN 0021-9991.

Jin, X., Cai, S., Li, H., and Karniadakis, G. E. NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 426:109951, February 2021. ISSN 0021-9991.

Keng, H. L. and Yuan, W. *Applications of Number Theory to Numerical Analysis*. Springer, Berlin, Heidelberg, 1981. ISBN 978-3-642-67831-8 978-3-642-67829-5.

Krishnapriyan, A., Gholami, A., Zhe, S., Kirby, R., and Mahoney, M. W. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1–13, January 2022.

Lagaris, I. E., Likas, A., and Fotiadis, D. I. Artificial neural networks for solving ordinary and partial differential equations. *IEEE Transactions on Neural Networks*, 9(5): 987–1000, 1998. ISSN 10459227.

Loshchilov, I. and Hutter, F. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, pp. 1–16, 2017.

Lu, L., Jin, P., and Karniadakis, G. E. DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv*, 3(3):218–229, 2019.

Lu, Y., Blanchet, J., and Ying, L. Sobolev Acceleration and Statistical Optimality for Learning Elliptic Equations via Gradient Descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, October 2022.

Mao, Z., Jagtap, A. D., and Karniadakis, G. E. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*, 360:112789, March 2020. ISSN 0045-7825.

Niederreiter, H. *Random Number Generation and Quasi-Monte Carlo Methods*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, 1992.

Pokkunuru, A., Rooshenas, P., Strauss, T., Abhishek, A., and Khan, T. Improved Training of Physics-Informed Neural Networks Using Energy-Based Priors: A Study on Electrical Impedance Tomography. In *International Conference on Learning Representations (ICLR)*, February 2023.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, February 2019. ISSN 0021-9991.

Sharma, R. and Shankar, V. Accelerated Training of Physics-Informed Neural Networks (PINNs) using Meshless Discretizations. In *Advances in Neural Information Processing Systems*, October 2022.

Sloan, I. H. and Joe, S. *Lattice Methods for Multiple Integration*. Clarendon Press, 1994.

Wang, C., Li, S., He, D., and Wang, L. Is $L\hat{2}$ Physics Informed Loss Always Suitable for Training Physics Informed Neural Network? In *Advances in Neural Information Processing Systems (NeurIPS)*, October 2022.

Wang, S. and Perdikaris, P. Long-time integration of parametric evolution equations with physics-informed DeepONets. *Journal of Computational Physics*, 475:111855, February 2023. ISSN 0021-9991.

Zaremba, S. K. *Applications of Number Theory to Numerical Analysis*. Academic Press, 2014.

Zeng, Q., Kothari, Y., Bryngelson, S. H., and Schaefer, F. T. Competitive Physics Informed Networks. In *International Conference on Learning Representations (ICLR)*, pp. 1–12, February 2023.