Soft Constraint in Local Structure Approximation-PINN

Jian Cheng Wong[1*], Pao-Hsiung Chiu[2*], Chinchun Ooi[3], My Ha Dao[4]
[1,2,3,4]*Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore*
[1]*School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore*
[3]*Center for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore*
Email: [1]wongj@ihpc.a-star.edu.sg, [2]chiuph@ihpc.a-star.edu.sg, [3]ooicc@cfar.a-star.edu.sg,
[4]daomh@ihpc.a-star.edu.sg
[*]Equal contribution

***Abstract*—We previously presented a novel loss formulation for efficient learning of complex dynamics from governing physics, typically described by partial differential equations (PDEs), using physics-informed neural networks (PINNs). In these experiments, the incorporation of a Boundary Connectivity (BCXN) loss function was shown to greatly improve physics-informed learning across many problems, especially those with complex geometries. However, this imposition may not always be ideal, as the previously presented BCXN-loss strongly enforces a linear local structure at the boundary. While this assumption helps facilitate faster learning with an order of magnitude sparser training samples, this can adversely impact convergence in other situations. Hence, we propose a modification of this BCXN-loss that reduces the imposed structure to a soft constraint, allowing for more flexible learning and convergence. We further demonstrate the potential for this method to improve the convergence and performance of LSA-PINN across additional numerical experiments, with much smaller errors than existing methods in terms of the standard L2-norm metric. In particular, we have applied this method to the modelling of flow past complex geometries such as airfoils, which serve as the basic building block for various applications in fluid dynamics and renewable energy (e.g., in wind and tidal turbine design).**

***Index Terms*—physics-informed neural networks, local structure approximation, boundary connectivity, fluid dynamics**

I. Introduction

The emerging topic of physics-informed neural networks (PINNs) [1] has drawn much interest in diverse areas of science and engineering due to its ability to ensure physically consistent predictions through direct incorporation of mathematically expressible laws of nature—usually in the form of partial differential equations (PDEs)—into learned models. For example, PINNs have been applied to fluid dynamics problems and renewable energy technologies such as wind turbines, both for predicting physical behaviour in either a single system or a set of parameterized systems, and in inverse inference and design optimization problems [2], [3].

Since many PDEs of practical interest are boundary-value problems, accurate representation of the near-boundary behaviour is particularly critical for PINNs. In this regard, we previously presented a Boundary Connectivity (BCXN) loss function [4] for enforcing structural biases and reducing overfitting near the boundary during PINN training. This BCXN-loss implicitly imposes a *linear local structure approximation*, thereby making it easier for PINNs to learn approximate gradients during training and accelerating convergence, especially in problems with challenging physics.

However, this imposition (direct enforcement of a linear local structure) may not always be ideal. While this simplified assumption helps facilitate faster learning with an order of magnitude sparser training samples for many fluid dynamics systems [4], this can adversely impact convergence in other situations. In this work, we propose a modification of this *hard constraint* BCXN-loss that reduces the imposed structure to a soft regularization, allowing for more flexible learning and convergence. This new *soft constraint* BCXN-loss function is key to a novel class of local structure approximation (LSA)-PINN method—based on the numerical differentiation (ND) or coupled-automatic-numerical (CAN) differentiation scheme for approximating the PINN loss [5])—which can more efficiently learn the solution to challenging PDE problems with sparser training samples, regardless of domain geometry.

The remaining sections are organized as follows: Section II reviews the LSA-PINN and elucidates the *soft constraint* approach. In Section III, we demonstrate the effectiveness of this *soft constraint* approach in improving the convergence and performance of LSA-PINN across multiple numerical experiments. We summarize the main findings and discuss potential avenues for future research in Section IV.

II. Methodology

A. LSA-PINN: enforcing linear constraint at near-boundary

Let us consider a LSA-PINN model with single input x and output $u(x;\boldsymbol{w})$ controlled by the network weights, $\boldsymbol{w}$. The LSA-PINN model utilizes CAN/ND-evaluated loss function to approximate the derivative terms in PDE constraint imposed during training based on the *output from neighbouring samples, i.e., finite difference-type stencils (local support points)*. This connects sparse training samples into piece-wise continuous regions to facilitate fast training across the entire problem domain. However, issues arise when dealing with irregular geometries because sample locations cannot perfectly connect the near-boundary training samples to the (irregular) domain boundary. As a result, the boundary condition (BC)

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00232

information may not successfully propagate to the inner domain. Moreover, the model output on external (out-of-domain) locations $u_{ES}(x; \boldsymbol{w})$ is under-defined and prone to *overfitting*, i.e., minimizing the PDE and BC constraints incorrectly.

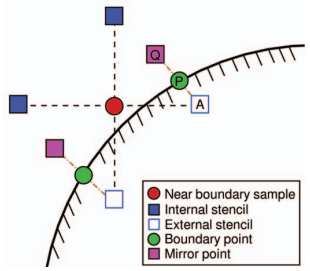

Fig. 1. Schematic of related definitions under the LSA-PINN framework. *BCXN-loss* enforces a constraint utilizing 3 types of points: external stencil point A, boundary point P, and mirror point Q along the normal direction inside the fluid domain.

The BCXN-loss is formulated to enforce linear constraints at the near-boundary region and restore connectivity between the domain boundary and near-boundary samples. The idea is to impose a linear constraint on the LSA-PINN output at the under-defined external stencil $u_{ES}(x; \boldsymbol{w})$ of a near-boundary sample (point A in Fig. 1), based on the Dirichlet BC, u_{BC}, defined at the boundary (point P) and the value, u_{MI}, of a chosen mirror location (point Q) inside the domain, during the evaluation of PDE constraint at such near-boundary sample using CAN/ND scheme. If point Q is chosen along the normal direction of boundary point P (ref. to Fig. 1), i.e., $\overline{AP} = \overline{PQ}$ where $\overline{AP}$ and $\overline{PQ}$ are the distances between A and P and between P and Q, the target value u_{ES} of point A can then be derived as:

$$u_{ES} = 2u_{BC} - u_{MI} \tag{1}$$

In practice, given a fixed set of training samples, all the external stencil points and their mirror locations can be pre-computed. While u_{MI} depends on the LSA-PINN output and needs to be evaluated every training iteration, $u_{MI}(\tilde{x}; \boldsymbol{w})$ can be easily obtained by directly evaluating the LSA-PINN at the corresponding mirror point location, $\tilde{x}$.

B. Hard constraint boundary connectivity (BCXN)-loss

The original LSA-PINN utilizes the following hard constraint (*hc*)-BCXN-loss:

$$\mathcal{L}_{\text{LSA-PINN}} = \lambda_{pde}\ \mathcal{L}_{pde} + \lambda_{\text{BCXN}}\ \mathcal{L}_{pde(hc-\text{BCXN})} \tag{2}$$

as PDE loss term *on near-boundary samples* whereby the PDE constraint in $\mathcal{L}_{pde(hc-\text{BCXN})}$ is directly modulated by the local linear condition *through employing Equation 1 to compute* u_{ES} *for any external stencil point*. Note that in $\mathcal{L}_{pde}$, the PDE constraint on the remaining training samples is evaluated by the usual CAN/ND scheme. This implementation implicitly infuses the BCs into the training loss, obviating the BC loss term commonly required for PINN training. This approach has an association with the *direct forcing immersed boundary methods* in numerical computing.

C. Soft constraint boundary connectivity (BCXN)-loss

The hard constraint BCXN-loss strongly enforces the linear condition during the evaluation of PDE constraint at near-boundary samples. However, the linear condition described by Equation 1 is not necessarily the best approximation to the local gradients. The imposition of such an over-simplified constraint can be effective in some scenarios but inappropriate in other problems, hence slowing convergence or reducing accuracy. To alleviate the issue, we *soften the linear constraint—which may conflict with physical local gradients near-boundary—by introducing a soft regularization*, in an approach referred to as *soft constraint* (*sc*)-BCXN-loss.

Let us denote u_{ES_i} as the target field value at the i-th external stencil point computed by Equation 1, and $u_{ES_i}(x; \boldsymbol{w})$ as LSA-PINN output at corresponding external stencil point. We then define *sc*-BCXN-loss as:

$$\mathcal{L}_{sf-\text{BCXN}} = \frac{1}{n_{ES}} \sum_{i=1}^{n_{ES}} \left(u_{ES_i} - u_{ES_i}(x; \boldsymbol{w})\right)^2 \tag{3}$$

for all the ES points $i = 1, ..., n_{ES}$ which are required for evaluating the PDE constraint using the CAN/ND scheme. The newly introduced $\mathcal{L}_{sf-\text{BCXN}}$ explicitly infuses BC information into the PDE samples, hence propagating the correct BC information during training. The $\mathcal{L}_{\text{LSA-PINN}}$ is then defined as:

$$\mathcal{L}_{\text{LSA-PINN}} = \lambda_{pde}\ \mathcal{L}_{pde} + \lambda_{bc}\ \mathcal{L}_{bc} + \lambda_{\text{BCXN}}\ \mathcal{L}_{sc-\text{BCXN}} \tag{4}$$

with additional weight, λ_{BCXN}, controlling the relative importance of the loss term $\mathcal{L}_{sc-\text{BCXN}}$ in the training loss function. When $\lambda_{BCXN} \rightarrow 0$, the $\mathcal{L}_{\text{LSA-PINN}}$ reverts to the usual CAN/ND-evaluated loss.

III. Results

A. Experimental setting

We study the performance of the LSA-PINN with the newly proposed *soft constraint* BCXN-loss on the following fluid dynamical systems with irregular geometry:

1) **2D wavy channel flow,** $Re = 100$**.** The fluid flow through a long wavy channel is studied. The inlet profile at left boundary is defined as $u(0, y) = -\frac{3}{2}y^2 + \frac{3}{2}, v = 0$. A non-slip condition is applied to the top and bottom walls, while outlet boundary conditions ($\frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = p = 0$) are applied to the right boundary.
2) **2D flow past airfoil shape,** $Re = 500$**.** To further validate the applicability and efficacy of the LSA-PINN method for real-world complex geometries, we study the incompressible flow past an airfoil with a NACA0012 profile. The airfoil is placed in a modelling domain $x \in [-3, 5]$, $y \in [-1, 2]$ with zero angle of attack. A uniform inlet velocity, $u_{inlet} = 1$, is specified at $x = -3$, and a zero pressure outlet boundary condition is specified at $x = 5$. Typical slip boundary conditions are specified at the side walls.

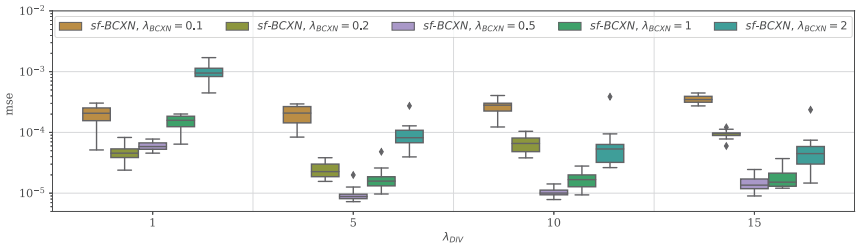

Fig. 2. Distribution of solution MSE (average of u and v-velocity components) obtained by a LSA-PINN for the wavy channel flow problem, $Re = 100$, when using different λ_{div} and λ_{BCXN} values in its training loss function $\mathcal{L}_{\text{LSA-PINN}}$ with a *soft constraint* BCXN-loss.

The governing physics for these fluid dynamical systems are the 2D steady-state incompressible N-S equations:

$$\frac{du}{dx} + \frac{dv}{dy} = 0 \tag{5a}$$

$$u\frac{du}{dx} + v\frac{du}{dy} = \frac{1}{Re}\left(\frac{d^2u}{dx^2} + \frac{d^2u}{dy^2}\right) - \frac{dp}{dx} \tag{5b}$$

$$u\frac{dv}{dx} + v\frac{dv}{dy} = \frac{1}{Re}\left(\frac{d^2v}{dx^2} + \frac{d^2v}{dy^2}\right) - \frac{dp}{dy} \tag{5c}$$

To evaluate the performance of PINN models, ground truth solutions are obtained by an in-house numerical solver based on the IDFC method [6].

The LSA-PINN with the incorporation of *sc*-BCXN-loss is compared with the baseline *hc*-BCXN-loss [4], CAN/ND-PINN (without BCXN-loss) [5] and SF-PINN (AD-evaluated loss) [7] models, based on mean squared error (MSE) and relative L2 error of the respective PINN solution's velocity (u, v) relative to ground truth. To facilitate comparison, identical neural architectures and initialization distribution are employed. All models were trained on a relatively sparse set of equidistantly spaced sample points: 15,200 and 79,886 points for 2D wavy channel flow and flow past airfoil problems, respectively, for 300,000 and 500,000 iterations using the Adam algorithm. For each experiment, we present the results from 10 independent runs with different initialization instances.

B. Effect of hyper-parameter tuning

The LSA-PINN training can be sensitive to hyper-parameters, namely the relative weights λ_{pde} and λ_{BCXN} in the loss function $\mathcal{L}_{\text{LSA-PINN}}$ (Equation 2 and Equation 4). Specifically, we can further decompose λ_{pde} into λ_{div} for the divergence free condition (Equation 5a) and λ_{mom} for the conservation of momentum equations (Equation 5b- 5c) in the PDE loss. With the *soft constraint* BCXN-loss, λ_{BCXN} controls the extent of linear constraint regularization being imposed on the near-boundary samples by the $\mathcal{L}_{sc-\text{BCXN}}$. We observe a noticeable improvement in solution quality with appropriately chosen λ_{BCXN} for the *sc*-BCXN-loss in both test cases.

Results from joint optimization of λ_{div} and λ_{BCXN} for the wavy channel flow test case are presented in Fig. 2. The best LSA-PINN solutions are achieved for λ_{BCXN}=0.5 and λ_{div}=5 (λ_{mom}=1). The results highlight that an overly stringent imposition of a linear constraint by the LSA-PINN can adversely affect convergence without an accompanying increase in the emphasis (λ_{div}) on the divergence-free condition. For the flow past airfoil test case, λ_{BCXN}=0.1 and λ_{div}=λ_{mom}=1 is found to be appropriate.

We also noticed that reducing the λ_{BCXN} for the *hard constraint* BCXN-loss $\mathcal{L}_{pde(hc-\text{BCXN})}$ can cause adverse outcomes because the imposition of BCs solely relies on this term.

C. Summary of forward simulation results

TABLE I
MODEL PERFORMANCE ON FORWARD SIMULATION TASKS.

Test case	Model	MSE[1]	Rel. Error[1]
Wavy channel flow, Re=100	LSA-PINN (*hc*-BCXN-CAN-loss)	1.34e−4	7.28e−2
	LSA-PINN (*sc*-BCXN-CAN-loss)	9.00e−6	4.09e−2
	CAN-PINN (CAN-loss)	5.96e−1	2.55e+0
	SF-PINN (AD-loss)	3.52e−2	7.19e−1
Flow past airfoil shape, Re=500	LSA-PINN (*sc*-BCXN-CAN-loss)	4.65e−5	8.10e−2
	LSA-PINN (*sc*-BCXN-ND-loss)	5.03e−5	6.93e−2
	CAN-PINN (CAN-loss)	3.32e−4	1.92e−1
	ND-PINN (ND-loss)	3.54e−4	1.84e−1
	SF-PINN (AD-loss)	1.18e−3	3.17e−1

[1] We compute the error for u, v-velocity components and take the average.

The average MSE and relative L2 error across multiple independent runs ($n_{run} = 10$) are summarized in TABLE I. The results indicate that the LSA-PINN is effective in learning a good solution with low MSE from the given set of sparse training samples on the fluid dynamical systems with irregular domain and complex physics. Both test cases show noticeable improvement in solution accuracy (1-3 orders of magnitude lower MSE) with the LSA-PINN method with *sc*-BCXN-loss, as compared to the baseline CAN/ND-PINN and SF-PINN methods. We also observe 1 order of magnitude lower MSE when we compare the *soft constraint* with *hard constraint* BCXN-loss in the wavy channel flow test case.

The visual comparison in Fig. 3 and Fig. 4 shows that the baseline PINN methods (CAN-PINN and SF-PINN) fail to produce a reasonable solution, i.e., they cannot learn the correct flow with similar settings. On the other hand, the solutions obtained from LSA-PINN (with either *hc* or *sc*-BCXN-loss) have very good agreement with the ground truth.

IV. CONCLUSIONS

In this study, we presented a strategy to improve LSA-PINN by introducing a *sc*-BCXN-loss term which provides

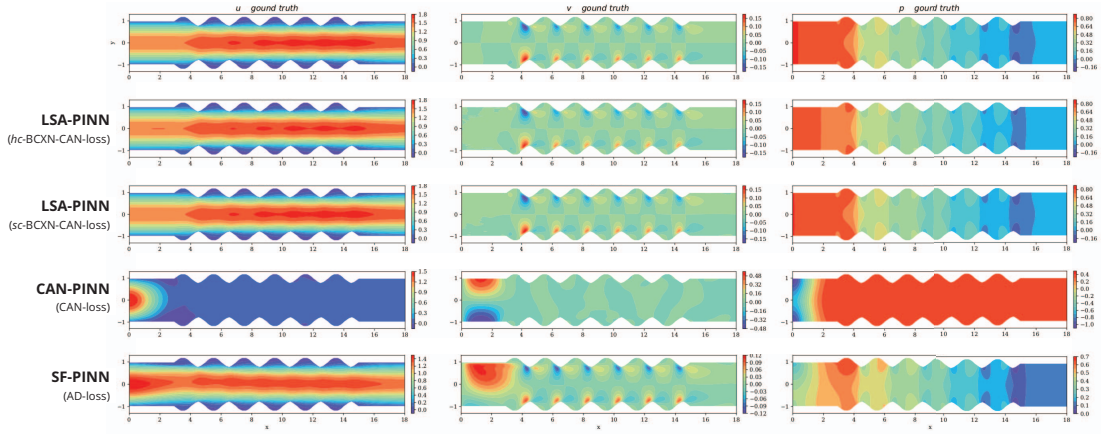

Fig. 3. Comparison of u, v-velocity and pressure p contour between the ground truth and PINN solutions (median MSE from 10 runs), for the wavy channel flow problem, $Re = 100$.

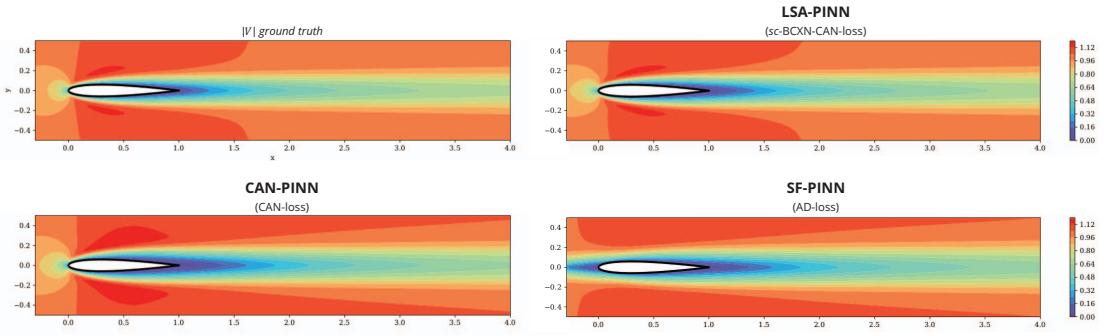

Fig. 4. Comparison of velocity magnitude $|V| = \sqrt{(u^2 + v^2)}$ contour between the ground truth and PINN solutions (median MSE from 10 runs), for the 2D flow past airfoil shape problem, $Re = 500$. The weight of BCXN-loss λ_{BCXN} is set to 0.1 for this problem.

soft regularization (in the form of a linear constraint) to the near-boundary gradient behaviour during training. This approach enhances the LSA-PINN's ability to achieve more flexible learning and convergence.

In particular, we have applied this *soft constraint* BCXN-loss to improve the convergence and performance of the LSA-PINN relative to existing methods, with much smaller errors across multiple numerical experiments. These experiments include the prediction of flow past complex geometries such as airfoils, which are a key canonical test problem for future extension to various real-world problems in fluid dynamics with potential application to sustainability and renewable energy (e.g., in wind and tidal turbine design).

Although linear approximations have been shown to be useful in the existing LSA-PINN framework as a trade-off between convergence and sampling density, we plan to explore the potential for non-linear approximation schemes to yield even better performance in future work.

Acknowledgment

This research is supported by the Agency for Science Technology and Research (A*STAR) under the AME Programmatic project: Explainable Physics-based AI for Engineering Modelling & Design (ePAI) [Award No. A20H5b0142].

References

[1] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics*, vol. 378, pp. 686–707, 2019.

[2] B. Huang and J. Wang, "Applications of physics-informed neural networks in power systems-a review," *IEEE Transactions on Power Systems*, vol. 38, no. 1, pp. 572–588, 2022.

[3] P. Sharma, W. T. Chung, B. Akoush, and M. Ihme, "A review of physics-informed machine learning in fluid mechanics," *Energies*, vol. 16, no. 5, p. 2343, 2023.

[4] J. C. Wong, P.-H. Chiu, C. Ooi, M. H. Dao, and Y.-S. Ong, "Lsa-pinn: Linear boundary connectivity loss for solving pdes on complex geometry," in *2023 International Joint Conference on Neural Networks (IJCNN)*, 2023, pp. 1–10.

[5] P.-H. Chiu, J. C. Wong, C. Ooi, M. H. Dao, and Y.-S. Ong, "Can-pinn: A fast physics-informed neural network based on coupled-automatic–numerical differentiation method," *Computer Methods in Applied Mechanics and Engineering*, vol. 395, p. 114909, 2022.

[6] P.-H. Chiu, "An improved divergence-free-condition compensated method for solving incompressible flows on collocated grids," *Computers & Fluids*, vol. 162, pp. 39–54, 2018.

[7] J. C. Wong, C. Ooi, A. Gupta, and Y.-S. Ong, "Learning in sinusoidal spaces with physics-informed neural networks," *IEEE Transactions on Artificial Intelligence*, 2022.