Proceedings of the 1st international Symposium on AI and Fluid Mechanics Paper No S5 P8

Physics-Informed Adversarial Network for prediction of turbulent flow over rough surfaces

Jiasheng Yang[1], Simon Dalpke[1], and Alexander Stroh[1]

[1]Institute of Fluid Mechanics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany

Abstract

Surface roughness usually develops on degraded flow-related equipment, leading to various roughness characteristics that significantly alter surface drag. Predicting roughness-induced drag has been the focus of extensive research for decades. In this study, a Physics-Informed Adversarial Network (PIAN) is proposed to predict Reynolds-averaged 2D turbulent flow over irregular rough surfaces at Re_τ = 800. PIAN is trained to deliver predictions that comply with Reynolds-Averaged Navier-Stokes (RANS) equations combined with a critic convolutional neural network (CNN). The critic CNN is trained based on direct numerical simulation (DNS) data. Stability and convergence of the training process are ensured using the Wasserstein distance. Additionally, a Fourier feature extraction technique is implemented to capture multi-scale roughness and turbulence interactions. The PIAN model achieves an average prediction error below 10% for local Reynolds stresses and velocities, with roughness function deviations under 4% compared to DNS data, demonstrating its effectiveness in integrating physics-based constraints with machine learning for roughness-induced flow predictions.

1 Introduction

Various topological features of rough surfaces, for instance pronounced peaks of pits, can lead to significant alteration in the behavior of flow over these surfaces. The most pronounced effect of surface roughness, in a practical sense, is an increase in the skin-friction drag under turbulent flow conditions. Acknowledging the significance of the impact of roughness, the study of turbulent flow over rough surfaces has been an active area of research for nearly a century. The drag induced by surface roughness can be quantified through the downward shift in the logarithmic region of the inner-scaled mean velocity profile widely observed on rough walls, which is referred to as roughness function ΔU^+ (Hama, 1954). It is worth noting that ΔU^+ depends on the Reynolds number and the roughness topography. As a result, for any particular rough surface, it needs to be determined through a cost-intensive numerical or laboratory experiment. Therefore, a great amount of effort in the past few decades has been devoted to determining the drag of an arbitrary roughness *a priori*.

In the present work, to facilitate drag effect predictions, knowledge about the physics is incorporated into the model training process of a physics-informed neural networks (PINNs, Raissi et al., 2017) utilizing the Reynolds-averaged Navier-Stokes (RANS) equations. The model is designed to predict 2D Reynolds-averaged turbulent flow slices in *x*-*y* plane in the context of arbitrary 1-D roughness profile along streamwise direction. Here *x* and *y* represents streamwise and wall-normal directions, respectively. Due to the well-known closure problem in the unknown Reynolds stress terms, a specific variant of PINN, namely Physics-Informed Adversarial Network (PIAN), is proposed. As the name suggests, PIAN leverages an adversarial 2D convolutional neural network (CNN) critic to compete with the PINN model to distinguish PINN-generated flow results from ground truth data acquired from direct numerical simulations (DNS). With such approach, the spatial correlation of Reynolds stress in 2D *x*-*y* flow field subjected to roughness boundary conditions can be incorporated into the model training.

2 Methodology

2.1 Roughness database

The irregular roughness topographies are generated using the mathematical algorithm proposed by Pérez-Ràfols & Almqvist, 2019. This method provides flexibility in designing roughness profiles using a prescribed probability density function (p.d.f.) and power spectrum (PS) of the roughness height, while preserving the inherent randomness and stochastic nature of irregular roughness. Using this approach, a total of 4200 artificial roughness samples with diverse configurations of p.d.f. and PS are generated to construct the raw roughness repository. For a detailed description of the repository construction, readers are referred to J. Yang et al., 2023. Following the active learning procedure used in the previous work J. Yang et al., 2023, a total of 92 roughness samples from the repository are selected and simulated using DNS in a closed channel with a constant pressure gradient at $\mathrm{Re}_\tau = 800$ and a channel half height H. To address the unfavorable computational demands associated with DNS, the concept of minimal channels Chung et al., 2015 was employed. It is important to note that due to variations in the largest roughness length scales across different roughness patches, the minimal channel sizes are adjusted accordingly to accommodate these large structures. This results in different critical heights of minimal channels, below which the turbulent flow is accurately resolved. To account for this, only the flow statistics from minimal channel DNS below $y = 0.3H$ are used for training, which is the minimal critical height of the minimal channels utilized across the training database. From these simulations, 1848 2D planes with flow statistics are collected for the PIAN training.

2.2 Physics-Informed Adversarial Network

The present machine learning framework, termed the Physics-Informed Adversarial Network (PIAN), comprises two competing machine learning models: the PINN introduced by Raissi et al., 2017 model is responsible for generating predictions of flow statistics over roughness and the critic convolutional neural network is responsible for separating them from ground-truth data (L. Yang et al., 2020). These models engage in an adversarial process, formulated as a zero-sum game, where the improvement of one model comes at the expense of the other.

2.2.1 Physics-Informed Neural Network

PINNs are a class of deep learning models that integrate physical laws into the training process. Specifically, these physical principles, often expressed in the form of (partial) differential equations, are embedded in the loss function of the neural network using automatic differentiation (AD). By incorporating these constraints, PINNs can be employed to achieve predictions adhere to fundamental physical laws. A visual illustration of the present PINN model is depicted in Figure 1. The model receives as input the one-dimensional streamwise roughness profile along with the spatial coordinates x and y and is designed to predict the 2D Reynolds-averaged flow statistics within half of a $3.6H$ long closed channel Ω. To enhance the network's ability to capture spatial variations of small scales, Fourier feature extraction technique (Tancik et al., 2020) is applied on the coordinate inputs. Additionally, one-dimensional convolution layers are employed to extract spatial features from the roughness profile. The resulting latent feature vector is then merged into the PINN model through fully connected layers. Following this, densely connected layers are appended to map the spatial coordinates and roughness features to the predictions. The details of the present PINN architecture of the is documented in Table 1. The final predicted solution vector for a given roughness profile k at a spatial position $\mathbf{x} = (x, y)$ consists of the predicted mean velocity components U and V, the mean pressure $\overline{p}$, and the Reynolds stress components $\overline{u'u'}$, $\overline{v'v'}$, and $\overline{u'v'}$. Please note, that the stresses with velocity

component normal to the considered x-y-plane are neglected. Subsequently, the partial derivatives of the predicted solutions are computed using AD and utilized during the computation of the loss functions. The governing partial differential equations (PDEs) and corresponding boundary conditions employed in the present work, formulated as loss functions ϵ, are summarized in Table 2. The logarithmic law condition ϵ_{Log}, in the form of a diagnostic plot, is enforced in the region $y \geq 0.3H(\Omega_{\text{Log}})$, where a logarithmic linear mean velocity profile is prescribed to calculate ΔU^+.

Table 1: Architecture details of PINN

Component	Description
Fourier Feature Extraction	Two set of Fourier mappings with 75 pairs of sin&cos nodes each. Respective weight matrices $\mathbf{W_1}$, $\mathbf{W_2}$ are normal-randomized with scaling factors of 1 and 15. Each branch consists of 3 fully connected layers with 150 neurons.
Convolution layers	Conv1D: 3×3, output dimension: 1 → 5 Conv1D: 3×3, output dimension: 5 → 9 Conv1D: 5×5, output dimension: 9 → 15 Conv1D: 5×5, output dimension: 15 → 20
Dense layers	Fourier and CNN features concatenated 6 fully connected layers contain 1000 neurons each.
Activation Functions	LeakyReLU for convolution layers Sine activation for fully connected layers.
Output Layer	Linear transformation to 6 output values.

The PINN optimization task falls under the category of multi-objective optimization. The total loss function is formulated by weighting individual loss terms as follows:

$$\mathbf{L}_{\text{total}} = \sum_i \lambda_i |\epsilon_i|, \tag{1}$$

where λ_i represents the weight factor associated with the i-th loss function. These weight factors are dynamically adjusted during training using a gradient-based tuning approach to balance the contributions of different physical constraints (Bischof & Kraus, 2021).

2.2.2 *Critic convolutional neural network*

The Critic CNN follows a classical 2D convolutional neural network (CNN) architecture, consisting of multiple 2D convolutional layers followed by fully connected layers, which map the extracted features to the final prediction. The input to the Critic CNN consists of 2D plane statistics of five flow quantities, namely U, V, $\overline{u'v'}$, $\overline{u'u'}$, and $\overline{v'v'}$, along with a binary mask distinguishing roughness and fluid regions. To manage memory efficiency and data augmentation, a sliding inspection window is implemented along the streamwise direction. The streamwise extent of the inspection window is set to $0.6H$, while the wall-normal coverage spans from $y = 0$

Table 2: Summary of loss functions. Here Φ_i represents arbitrary predicted flow statistics. Ω_{fluid} and Ω_{solid} are the 2D sub-domain of fluid and roughness, respectively. $\partial\Omega_{\text{fluid}}$ is the boundary of the fluid domain.

Physical information	Loss function	Domain	Initial weight factor λ
Mass conservation	$\epsilon_{\text{mass}} = \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y}$	$(x, y) \in \Omega_{\text{fluid}}$	50
Momentum conservation	$\epsilon_x = U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + \frac{1}{\rho}\frac{\partial P}{\partial x} - \nu(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}) + \frac{\partial \overline{u'u'}}{\partial x} + \frac{\partial \overline{u'v'}}{\partial y} + P_x$ $\epsilon_y = U\frac{\partial V}{\partial x} + V\frac{\partial U}{\partial y} + \frac{1}{\rho}\frac{\partial P}{\partial y} - \nu(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}) + \frac{\partial u'v'}{\partial x} + \frac{\partial v'v'}{\partial y}$	$(x, y) \in \Omega_{\text{fluid}}$	50
Log-law	$\epsilon_{\text{Log}} = (\nu^+ - k^+_{\text{md}})\frac{\partial U^+}{\partial y^+} - \frac{1}{\kappa}$	$(x, y) \in \Omega_{\text{Log}}$	50
Periodic BC	$\epsilon_{\text{periodic}} = \sum_i \lvert\Phi_i(0, y) - \Phi_i(L_x, y)\rvert$	$(x, y) \in \partial\Omega_{\text{fluid}}$	1
No-slip BC	$\epsilon_{\text{no-slip}} = \sum_i \lvert\Phi_i(x, y)\rvert$	$(x, y) \in \Omega_{\text{solid}}$	100
Symmetry BC	$\epsilon_{\text{symm}} = \sum_{i \in (V, \overline{u'v'})} \lvert\frac{\partial \Phi_i(x,H)}{\partial y}\rvert + \sum_{i \in (V, \overline{u'v'})} \lvert\Phi_i(x, H)\rvert$	$(x, y) \in \partial\Omega_{\text{fluid}}$	1

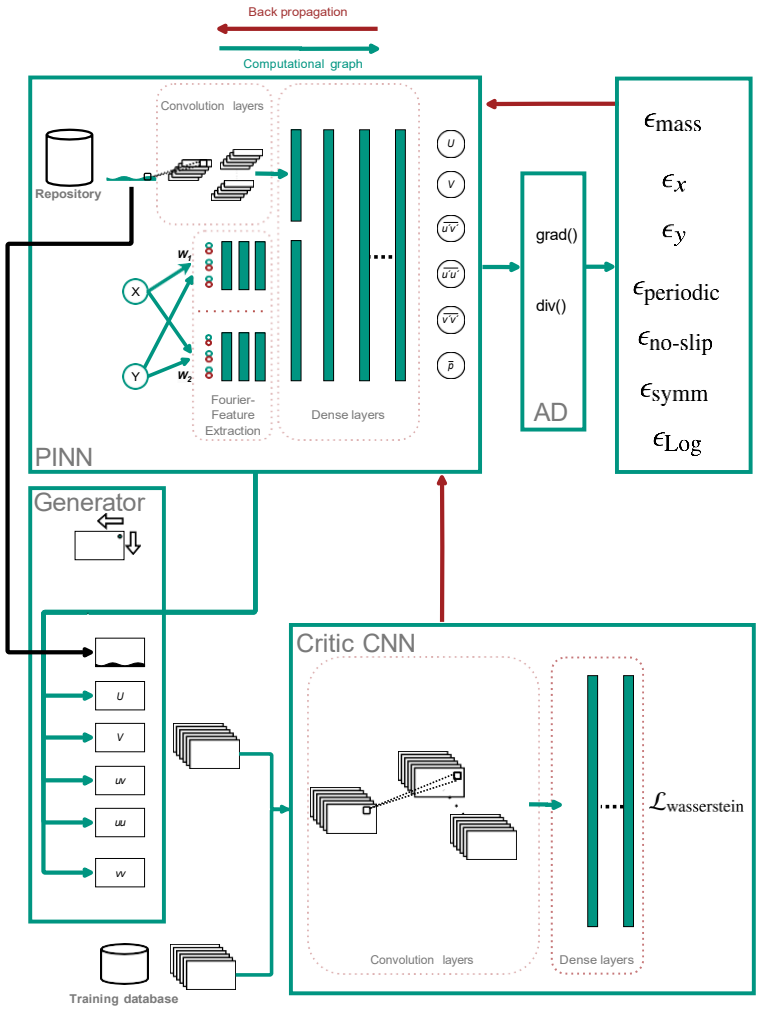

Figure 1: Schematic representation of the PIAN training procedure. In the Fourier-Feature Extraction module, green and red nodes denote the sine and cosine activations, respectively.

to $y = 0.3H$. The streamwise position of the inspection window is randomly selected for both the realistic DNS flow fields and the PINN-predicted flow fields. The data within the sliding inspection window are taken as input for the CNN model. The resolution of the Critic CNN is 76×70 in the streamwise and wall-normal directions, respectively, leading to an input tensor of dimension $6 \times 76 \times 70$. The architecture of the present CNN model is documented in Table 3. It is worth noting that for training the PINN model under supervision of the critic CNN, the Wasserstein distance is employed as the loss metric to improve the stability of adversarial training. To facilitate this, a linear activation function is applied to the output neuron of the CNN, ensuring the loss function correctly reflects the distance metric.

2.2.3 *Model training*

As introduced in the previous sections, the training of PIAN consists of two sequential steps in each epoch: (1) weight optimization based on the loss function gradient, incorporating both the governing equations and (2) feedback from the Critic CNN. The first optimization step is realized by randomly querying the loss function of 5000 sample points in terms of arbitrary roughness topography from the repository, at different spatial positions $(x, y) \in \Omega$.

Table 3: Architecture of critic CNN

Component	Layer	Description	Output Dimensions
Initial Convolution and Pooling	Convolution Layer	7×7, ReLU	$6 \rightarrow 64$
	Max pooling	3×3	
Residual net	Residual Block	$2 \times$Conv layers, 3×3, ReLU	$64 \rightarrow 64$
	Residual Block	$2 \times$Conv layers, 3×3, ReLU	$64 \rightarrow 128$
	Residual Block	$2 \times$Conv layers, 3×3, ReLU	$128 \rightarrow 256$
	Residual Block	$2 \times$Conv layers, 3×3, ReLU	$256 \rightarrow 512$
Dense layers	Fully Connected	ReLU	$3072 \rightarrow 4096$
	Fully Connected $\times 2$	ReLU	$4096 \rightarrow 4096$
	Output Layer	Linear	$4096 \rightarrow 1$

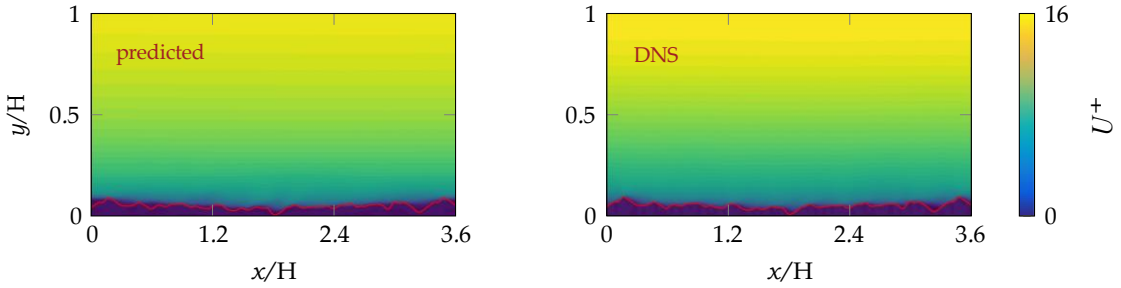

Figure 2: Comparison of mean velocity fields over roughness height (marked in red).

In the second step, the PINN generates sets of 2D flow statistics along with the 2D roughness binary masks within a randomly positioned sliding window. This is realized through the generator module which queries predictions at each spatial location within Ω and $y \leq 0.3H$, subsequently, the sliding window is applied at an arbitrarily selected streamwise position to crop the 2D statistics according to the size and resolution required for the critic CNN. Additional to the predicted 2D statistics, the critic CNN is also provided with real 2D flow fields from DNS, which are downsampled to match the CNN's resolution. The same sliding window technique is applied to the DNS data, allowing the critic CNN to effectively compare between physically realistic flow fields and those generated by the PINN based on local roughness features. The difference between the distributions of generated and real flow statistics is quantified by Wasserstein distance $L_{WS}(\mu_G, D)$. where μ_G represents the distribution of flow statistics generated by the PINN. The function D is the output of the critic CNN, which evaluates how realistic a given flow field appears. The objective of training is designed so that the critic CNN assigns lower values to predicted data and higher values to DNS-based data. As training progresses, this mechanism pushes the PINN to generate flow statistics that closely resemble the real data, thereby improving both the accuracy and physical consistency of the predictions. Consequently, the training objective can be expressed as $\min_{\mu} \max_{D} L_{WS}(\mu_G, D)$.

3 Results

A set of DNS data is hold out of training process to evaluate the performance of the model. The PINN and the generator module are applied to predict 2D flow statistics over these testing roughness. Visual comparison of two cases are illustrated in figure 2, where it is evident that the prediction successfully reflect the mean flow distribution in the context of wall roughness. A pixel-wise averaged error of maximum 10% is achieved for 2D flow statistics. These predicted flow fields can then be averaged in streamwise direction and across all planes from the same

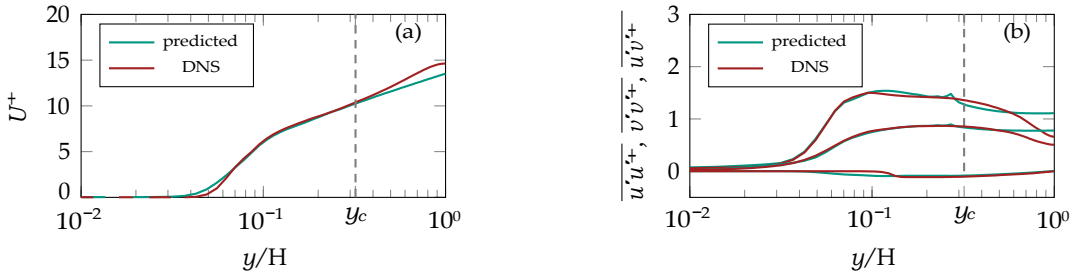

Figure 3: Comparison of (a) mean velocity profiles and (b) Reynolds stresses. The overshoot of mean velocity profile in DNS data above y_c is attributed to the nature of the minimal channel. The deviation of Reynolds stress due to the absence of the Critic CNN in this region.

roughness to calculate flow statistics profiles as functions of wall-normal coordinate y and shown in figure 3. A mean absolute percentage error (MAPE) of 4% is achieved for ΔU^+ by the present model for the considered data.

4 Conclusion

This study presents the Physics-Informed Adversarial Network (PIAN) as a robust framework for predicting turbulent flow over rough surfaces. By integrating physics-based constraints through the Reynolds-Averaged Navier-Stokes (RANS) equations and employing adversarial training with a critic CNN, the proposed PIAN accurately predicts flow statistics while maintaining spatial correlations in the Reynolds stress field. The model achieves an average local prediction error below 10% for velocity and stress components, with deviations in ΔU^+ values remaining under 4% compared to DNS data. These findings underscore the potential of physics-informed deep learning as a computationally efficient and physically consistent alternative to conventional numerical simulations. Future work will aim to extend PIAN to three-dimensional roughness and enhance its generalizability across diverse flow conditions, e.g., at various Re_τ.

References

Bischof, R., & Kraus, M. (2021). Multi-objective loss balancing for physics-informed deep learning. https://doi.org/10.13140/RG.2.2.20057.24169

Chung, D., Chan, L., MacDonald, M., Hutchins, N., & Ooi, A. (2015). A fast direct numerical simulation method for characterising hydraulic roughness. *Journal of Fluid Mechanics*, 773, 418–431. https://doi.org/10.1017/jfm.2015.230

Hama, F. R. (1954). Boundary-layer characteristics for smooth and rough surfaces. *Society of Naval Architects and Marine Engineers*.

Pérez-Ràfols, F., & Almqvist, A. (2019). Generating randomly rough surfaces with given height probability distribution and power spectrum. *Tribology International*, 131, 591–604. https://doi.org/https://doi.org/10.1016/j.triboint.2018.11.020

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2017). Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. https://arxiv.org/abs/1711.10561

Tancik, M., Srinivasan, P. P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J. T., & Ng, R. (2020). Fourier features let networks learn high frequency functions in low dimensional domains. https://arxiv.org/abs/2006.10739

Yang, J., Stroh, A., Lee, S., Bagheri, S., Frohnapfel, B., & Forooghi, P. (2023). Prediction of equivalent sand-grain size and identification of drag-relevant scales of roughness – a data-driven approach. *Journal of Fluid Mechanics*, 975, A34. https://doi.org/10.1017/jfm.2023.881

Yang, L., Zhang, D., & Karniadakis, G. E. (2020). Physics-informed generative adversarial networks for stochastic differential equations. *SIAM Journal on Scientific Computing*, 42(1), A292–A317. https://doi.org/10.1137/18M1225409