جامعة ابن طفيل
Ibn Tofaïl University
Faculté des Sciences

Université Ibn Tofail
Faculté des Sciences, Kénitra

Mémoire de Projet de Fin d'Etudes

Master Intelligence Artificielle et Réalité Virtuelle

Super-resolution and Denoising of 4D-Flow MRI using PINN

Établissement d'accueil : **LS2N Laboratory**

Elaboré par : Mr. Ayyoub JAICHI

Encadré par : Mme Khaoula BOUKIR (FSK Ibn Tofail University)
Mr. Sebastien LEVILLY (LS2N Laboratory)
Mr. Simon PERRIN (LS2N Laboratory)

Soutenu le 01/10/2024, devant le jury composé de :

- Mme Raja TOUAHNI (FSK Ibn Tofail University)
- Mr Anass NOURI (FSK Ibn Tofail University)
- Mr Sébastien LEVILLY (LS2N Laboratory)
- Mme Khaoula BOUKIR (FSK Ibn Tofail University)

Année universitaire 2023/2024

This study implements and evaluates an innovative approach to enhance 4D Flow Magnetic Resonance Imaging (MRI) using Physics-Informed Neural Networks (PINNs). By integrating Navier-Stokes equations as physical constraints within a deep learning model, we develop a framework that leverages Gaussian Quadrature for efficient MRI signal simulation and utilizes complex image representation to encode both signal strength and velocity information.

Our approach is validated through a series of experiments, including Poiseuille flow in various pipe geometries and simulated aortic flow. Results demonstrate robust performance in predicting flow patterns, even in noisy environments, and show promising generalization capabilities to unseen regions.

Key findings include the successful implementation of the PINN framework for fluid dynamics problems, demonstrating effective noise handling and super-resolution capabilities. The research also highlights the potential for applying this approach to complex, physiologically relevant flow scenarios, paving the way for improved cardiovascular imaging and diagnostics.

This research contributes to the growing field of physics-guided deep learning in medical imaging, offering a pathway towards more accurate cardiovascular diagnostics and potentially reduced MRI acquisition times. While computational complexity and parameter tuning present challenges, the results suggest significant potential for enhancing 4D Flow MRI resolution and quality.

Future work will focus on advanced PINN architectures, multi-physics integration, and clinical validation studies.

Partner Organizations

Acknowledgement

As Robert Greene wisely said, "You gain more by finding a rich mine and mining it deeper, than by flitting from one shallow mine to another." I feel blessed to have discovered several brilliant 'mines' filled with golden ideas over these two years, which have expanded my imagination and provided new ways of thinking. I owe special thanks to my supervisors at LS2N:

Mr. Sébastien LEVILLY, a wellspring of brilliant ideas, who was generous with his guidance and had a magical way of transforming complex concepts into straightforward, intuitive ones. Mr. Simon PERRIN, whose important remarks were always invaluable; without his input, this work could not have been accomplished.

I'm deeply grateful to my pedagogical supervisor at UIT, Khaoula BOUKIR, a brilliant and logical mind with her wonderful motto, "Where focus goes, energy flows." Special thanks go to the coordinator of the Master's program, Rajaa TOUAHNI, a kind-hearted yet strict mind who kept us on track. I extend my gratitude to the entire pedagogical team of the Master in Artificial Intelligence and Virtual Reality program for their dedication and support. I am thankful to the entire IPI team for welcoming me during these five months, as well as to all the members of LS2N, who made this internship a pleasant and enriching experience. These individuals and teams have not just taught me, but have inspired me to dig deeper, think critically, and approach complex problems with creativity and rigor. Their guidance has been instrumental in shaping my academic journey and future career path.

Abstract

This study implements and evaluates a novel approach for super-resolution and denoising of 4D Flow Magnetic Resonance Imaging (MRI) using Physics-Informed Neural Networks (PINNs). Building upon the work of Fathi *et al.* [6], we develop a framework that integrates the Navier-Stokes equations as physical constraints within a deep learning model. Our method employs Gaussian Quadrature for efficient MRI signal simulation and utilizes complex image representation to encode both signal strength and velocity information.

We validate our approach through a series of experiments, including Poiseuille flow in various pipe geometries and simulated aortic flow derived from Computational Fluid Dynamics (CFD) data. The PINN demonstrates robust performance in predicting flow patterns, particularly in noisey environments, and shows promising generalization capabilities to unseen regions in half-pipe scenarios.

Key findings include the successful implementation of the PINN framework for fluid dynamics problems, effective noise handling, and the potential for application to more complex, physiologically relevant flow scenarios. While computational complexity and parameter tuning present challenges, the results suggest significant potential for enhancing 4D Flow MRI resolution and quality.

This research contributes to the growing field of physics-guided deep learning in medical imaging, offering a pathway towards more accurate cardiovascular diagnostics and potentially reduced MRI acquisition times. Future work will focus on advanced PINN architectures, multi-physics integration, and clinical validation studies.

List of Figures

Contents

Chapter 1

Introduction

1.1 Host Institution LS2N

The Laboratoire des Sciences du Numérique de Nantes (LS2N - UMR 6004), which hosts my internship, is a leading French research institute in digital sciences. Founded in 2017, LS2N excels in robotics, artificial intelligence, and data science. The lab's mission is to advance scientific knowledge and develop practical solutions for societal and industrial challenges. LS2N's commitment to excellence and interdisciplinary collaboration has quickly established it as a key player in the global scientific community, driving innovation in digital technologies.

Figure 1.1: LS2N logo

1.2 Research Team and Environment

LS2N is organized into specialized research teams, each concentrating on specific areas of digital sciences. I conducted my internship with the Image Perception Interaction (IPI) team, which focuses on advancing computer vision, image processing, and human-computer interaction. IPI's research is aimed at developing cutting-edge algorithms and systems for interpreting and interacting with visual data. This work paves the way for innovative applications in fields such as augmented reality and medical imaging, pushing the boundaries of how we perceive and interact with visual information in both virtual and real-world environments.

Figure 1.2: IPI logo

1.3 Project Overview

This internship project focuses on the application of Physics-Informed Neural Networks (PINNs) for super-resolution and denoising of 4D Flow MRI data. 4D Flow MRI is a powerful technique for assessing cardiovascular function, but it often suffers from limited spatial resolution and noise. Our research aims to leverage the power of PINNs to enhance the quality and resolution of these images, potentially improving diagnostic capabilities in clinical settings. This work is conducted under the supervision of M. Sébastien LEVILLY and M. Simon PERRIN, whose expertise and guidance are instrumental in navigating the complex intersection of deep learning, fluid dynamics, and medical imaging.

Chapter 2

4D Flow MRI

2.1 4D Flow MRI Importance

Magnetic Resonance Imaging (MRI) has revolutionized medical diagnostics by providing detailed, non-invasive visualization of internal body structures. Within this field, 4D Flow MRI has emerged as a powerful technique for assessing blood flow dynamics in three-dimensional space over time. This advanced imaging modality offers unmatched insights into cardiovascular function, allowing clinicians to evaluate complex flow patterns, quantify blood velocity, and assess hemodynamic(fuild mechanics of blood) parameters. By analyzing the velocity field, physicians can detect abnormal flow characteristics associated with conditions such as valvular diseases, aortic aneurysms, and congenital heart defects. For instance, increased flow velocities or turbulent flow patterns can indicate the presence of stenosis or regurgitation in heart valves. The velocity data also enables the computation of critical hemodynamic parameters, including wall shear stress, pressure gradients, and vorticity, which are essential for understanding the biomechanics of blood flow and its impact on vessel walls. These parameters can help in assessing the risk of atherosclerosis, aneurysm formation, or thrombus development. Furthermore, the ability to calculate metrics like flow rate, kinetic energy, and flow displacement allows for a comprehensive evaluation of cardiac function and efficiency. This wealth of information derived from the velocity field not only aids in more accurate diagnoses but also assists in treatment planning and monitoring disease progression. The importance of 4D Flow MRI in the medical environment cannot be overstated, as it enables more accurate diagnosis of cardiovascular diseases, aids in surgical planning, and facilitates the monitoring of treatment efficacy for conditions such as congenital heart defects, aneurysms, and

valvular disorders. Appeal to Appendix B for detailed information about MRI.

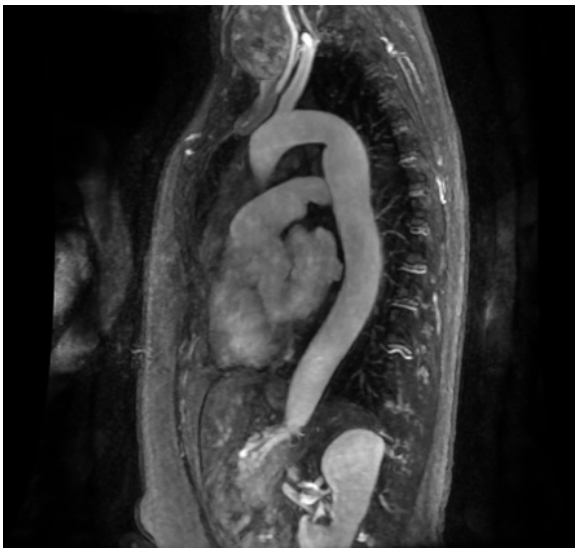

Figure 2.1: Magnetic Resonance Imaging (MRI) of the human torso

Image 2.1 displays the aorta, the body's main artery, which is clearly visible as a bright curvy tube. This MRI scan provides an excellent view of the aorta's path, showcasing its characteristic arch as it emerges from the heart and then descends through the chest cavity. This view allows medical professionals to assess the aorta's size, shape, and path, which is crucial for diagnosing conditions such as aortic aneurysms. Such detailed visualization of the aorta is essential for planning surgical interventions, evaluating the progression of aortic diseases, and monitoring treatment outcomes in patients with cardiovascular conditions.

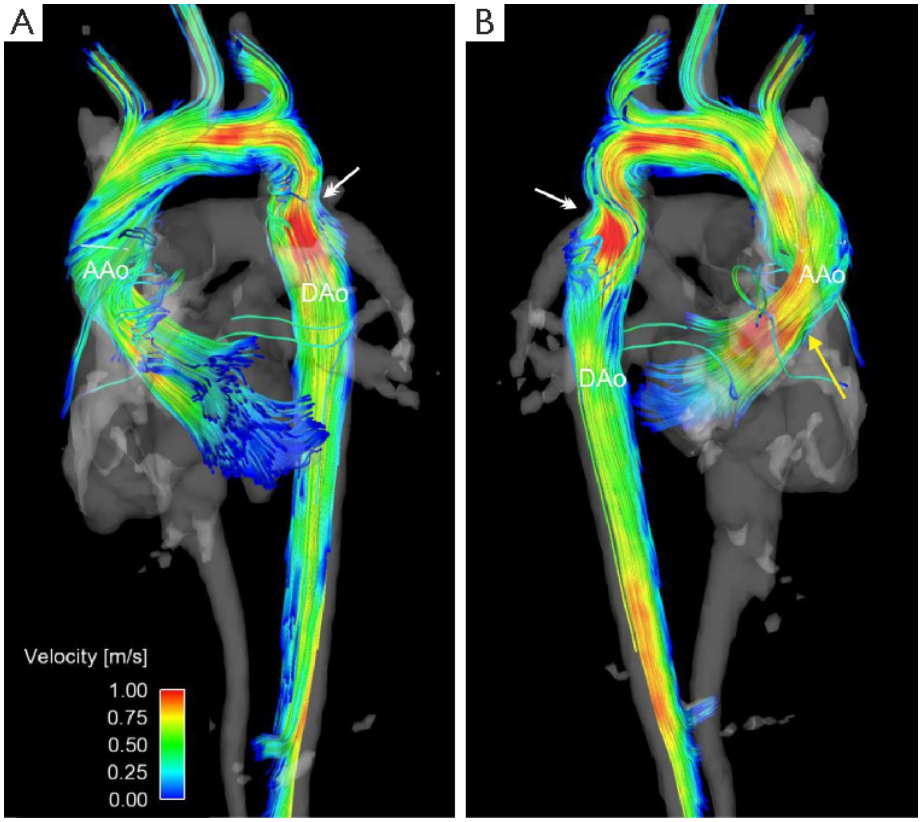

Figure 2.2: 4D Flow MRI of the human heart

Image 2.2 showcase a 4D flow MRI image of blood flow in a 3.5-year-old patient with bicuspid aortic valve (BAV) (congenital heart condition where the aortic valve has only two leaflets (or cusps) instead of the normal three) and aortic coarctation (congenital heart defect involving a narrowing of the aorta). The image, presented in anterior (A) and posterior (B) views, the ascending aorta (AAo) and descending aorta (DAo) are clearly labeled, it uses color-coded 3D streamlines to illustrate blood flow patterns.

2.2 4D Flow MRI Limitations

Despite its potential, 4D Flow MRI faces several significant limitations that can impact its clinical utility:

1. **Velocity aliasing:** This occurs when the blood flow velocity exceeds the

preset velocity encoding (VENC) value, resulting in phase wrapping and inaccurate velocity measurements. This can lead to misinterpretation of flow patterns and erroneous quantification of hemodynamic parameters.

2. **Low spatial and temporal resolution:** The need to balance scan time, spatial coverage, and temporal resolution often results in compromised image quality. This limitation can obscure fine flow details and hinder the detection of subtle abnormalities, particularly in smaller vessels or regions with complex flow patterns.

3. **Acquisition noise:** Various sources of noise, including thermal noise, and physiological motion, can degrade image quality and introduce errors in velocity measurements. This noise can mask important flow features and reduce the overall reliability of the data.

4. **Long acquisition times:** 4D Flow MRI scans typically require extended scanning periods, which can be challenging for patients and increase the likelihood of motion artifacts. This limitation also reduces the technique's practicality in clinical settings.

5. **Complex post-processing:** The vast amount of data generated by 4D Flow MRI necessitates sophisticated post-processing techniques, which can be time-consuming and require specialized expertise.

In summary, 4D flow MRI is a powerful tool for viewing blood flow in 3D over time, offering new insights into heart and blood vessel problems. It greatly improve how doctors diagnose and treat these issues. However, it has some big drawbacks. Scans take a long time, which is hard on patients. The images aren't always clear enough to see important details. Processing the data is complicated and time-consuming. These problems make hinder the use of 4D flow MRI in everyday patient care, even though it's very promising. Solving these issues is key to making the most of this advanced imaging method, in the next chapter we will discuss different approach to solve these limitations.

2.3 Problem and Approach

4D Flow MRI is an indispensable tool in modern medical diagnostics, Despite its potential, challenges such as low spatio-temporal resolution, acquisition noise, velocity aliasing, and phase-offset artifacts have hindered its clinical accessibility.

Fathi *et al.* [6] tries to solve these problems by introducing a new approach that is based on PINNs, velocity field, pressure, and MRI signal are modeled as a patient-specific neural network. For training, 4D-Flow MRI images in the complex Cartesian space are used to impose data-fidelity. Physical constraints were imposed through regularization. Creative loss function terms have been introduced to handle noise and super-resolution. The trained patient-specific DNN can be sampled to generate noise-free high-resolution flow images.

The primary objective is twofold: first, to implement and reproduce the super-resolution technique for 4D Flow MRI as presented in Fathi's paper, and second, to critically analyze the vanilla Physics-Informed Neural Networks (PINNs) approach used in the paper. Our goal is to identify fundamental flaws in the vanilla PINN methodology and present theoretical solutions to these issues. While we do not experimentally explore these solutions, we aim to provide a comprehensive theoretical framework for potential improvements. This work seeks to contribute to the field by not only validating existing research but also by highlighting areas for future advancements in the application of PINNs to medical imaging and fluid dynamics problems.

Chapter 3

State of the Art

Super-resolution and denoising of 4D flow MRI have been critical areas of research, aiming to enhance the quality and resolution of cardiovascular imaging data. Over the years, researchers have explored various approaches to tackle these challenges, each with its own strengths and limitations. Traditional methods, such as interpolation and filtering techniques, were initially used to improve image quality. However, these approaches often struggled with preserving fine details and handling complex flow patterns. This limitation paved the way for more advanced techniques, particularly those leveraging the power of artificial intelligence and deep learning. Convolutional Neural Networks (CNNs) have emerged as a powerful tool in this domain. Shit *et al.* (2022) [18] introduced SRflow, a deep learning-based approach for super-resolution of 4D flow MRI data. This method demonstrated significant improvements in spatial resolution while maintaining the accuracy of hemodynamic measurements. CNNs excel at capturing spatial features and have shown remarkable success in enhancing image quality. Building upon the success of CNNs, researchers have explored more sophisticated architectures. U-Net based models, originally developed for biomedical image segmentation, have been adapted for super-resolution tasks in 4D flow MRI. These architectures are particularly effective due to their ability to capture and combine features at different scales. Generative Adversarial Networks (GANs) represent another significant advancement. By pitting two neural networks against each other – one generating enhanced images and the other discriminating between real and generated images – GANs have shown promise in producing high-quality, realistic 4D flow MRI data. Ensemble learning approaches have also gained traction [7]. Ericsson *et al.* (2024) [5] proposed a method that extends super-resolution techniques across the entire cardiovascular system using

ensemble learning. This approach demonstrates the potential for more generalized and robust super-resolution methods. As the field progressed, researchers began to recognize the importance of incorporating domain-specific knowledge into these AI models. This realization led to the development of physics-informed approaches, most notably Physics-Informed Neural Networks (PINNs). PINNs, introduced by Raissi *et al.* [16], integrate physical laws directly into the neural network architecture, ensuring that the enhanced images not only look better but also adhere to the underlying fluid dynamics principles, PINN was shown to performe well in the context of super-resolution and denoising of 4D Flow MRI using PINN Shone *et al.* [19]. Attention mechanisms and Transformers, which have revolutionized natural language processing, are now being applied to medical imaging tasks. He et al. (2020) [10] explored the use of deep attention networks for MRI super-resolution, showcasing the potential of these techniques in capturing long-range dependencies in imaging data, also Wang *et al.* (2022) [24] introduced TransFlowNet, a physics-constrained Transformer framework for spatio-temporal super-resolution of flow simulations. While not specifically developed for 4D flow MRI, this approach demonstrates the potential of combining advanced AI architectures with physical constraints.

As the field continues to evolve, there's a growing trend towards integrating multiple techniques and leveraging domain-specific knowledge. The future of super-resolution and denoising in 4D flow MRI likely lies in approaches that can effectively combine the strengths of deep learning and physical principles to produce accurate, high-resolution, and physically consistent flow fields.

Physics-Informed Neural Networks (PINNs) are showing great promise in improving 4D flow MRI, especially compared to other AI methods. Working with 4D flow MRI data is like trying to find your way in a dark, unfamiliar neighborhood - the data is often noisy and unreliable. While traditional approaches like Convolutional Neural Networks (CNNs) can spot some patterns, and Generative Adversarial Networks (GANs) can create realistic-looking images they're like using a weak flashlight - helpful, but limited. Attention mechanisms and transformers, which are good at focusing on important details, are like having a spotlight, but they might miss the bigger picture. In this tricky situation, adding physical principles from fluid mechanics to the neural network is like giving it a steady-burning candle to light the way. This "candle" of knowledge, mainly in the form of Navier-Stokes equations, helps PINNs navigate through the messy data and find realistic solutions. While other AI methods might stumble around or get distracted by misleading patterns, PINNs use the light of physics to understand blood flow patterns, even where the data is as unclear as a pitch-black street.

Recent studies have shown that this physics-guided approach works better than older methods, much like someone with a reliable light source would find their way better than someone with unreliable or partial lighting.

The main reason we choose to work with PINNs is their remarkable flexibility. They can be seamlessly integrated with Convolutional Neural Networks (CNNs), leveraging CNNs' ability to capture spatial features while enforcing physical constraints. PINNs also work well with attention mechanisms and transformers, allowing the model to focus on critical flow regions while ensuring physical consistency. Moreover, PINNs can be incorporated into Generative Adversarial Network (GAN) frameworks, combining the realism of GAN-generated images with physics-based accuracy. This integration capability means PINNs don't replace other AI methods but enhance them, acting like a physics-aware plugin that can be added to various neural network architectures. By doing so, PINNs address the limitations of pure data-driven approaches, adding a layer of physical understanding to these powerful AI tools. This flexibility allows researchers to leverage the strengths of different AI methods while ensuring the results adhere to fundamental fluid dynamics principles, potentially leading to more accurate, physically consistent, and reliable 4D flow MRI enhancements.

Physics-Informed Neural Networks (PINNs) represent a significant leap in the intersection of machine learning and scientific computing. These innovative neural network architectures integrate well-established physical laws directly into the learning process, ensuring that the solutions they generate adhere to fundamental physical principles. By embedding these physical constraints, PINNs can efficiently solve forward and inverse problems across a wide range of domains, including fluid dynamics [14], quantum mechanics [17], reinforcement learning [13] [8], etc. This approach not only enhances the accuracy and reliability of the solutions but also reduces the dependency on large datasets, making PINNs a powerful and versatile tool for tackling complex scientific and engineering challenges

3.1 Physical Knowledge

When we refer to physical knowledge, we mean **differential equations**, **intuitive physics**, and **symmetry principles**. Differential equations are fundamental mathematical expressions that relate the rate of change of a function to the function itself. Intuitive physics comprises principles commonly employed by physicists when analyzing or solving problems, such as energy or momentum conservation laws, the principle of least action, Newton's laws of motion, and gravity. Sym-

metry refers to the various ways in which a phenomena exhibit independence. For instance, the laws of motion remain consistent regardless of time, meaning they exhibit time symmetry. Similarly, translation symmetry, scale symmetry, and rotational symmetry are examples of how phenomena maintain consistency across different spatial and dimensional scales.

How we can combine physical knowledge and deep learning ?

Physical knowledge can be seamlessly integrated into deep learning models through diverse avenues such as **data augmentation**, **architectural design**, **objective formulation**, and **optimization techniques**. For instance, we can enhance datasets by leveraging symmetries inherent in physical problems. Moreover, embedding physics into model architecture fitted to specific problems is a promising approach. Incorporating physical insights into loss functions through regularization terms or by redefining them to exploit physics further enriches model training. Additionally, refining optimization algorithms to achieve faster convergence and greater stability is another avenue for integrating physical understanding into machine learning frameworks.

3.2 Vanilla PINN

In this section will provide a detailed presentation of the original PINN algorithm as firstly described in the Raissi *et al* [16]. PINN generally deals with Partial Differential Equations, we will formulate the Partial Differential Equations as follow:

$$\frac{\partial f(x, y, z, t)}{\partial t} + \mathcal{N}[f, C] = 0 \tag{3.1}$$

f represents a function that could represents any physical phenomena, $\mathcal{N}$ represents non-linear operator acting on f where C represents constants belong to the theory of the phenomena e.g gravitational constant or coulomb constant.

3.2.1 PINN Steps

Informally

Our goal is to find f, we will use a neural network to approximate f_{NN} and train it on data points $\mathbf{x}_1, ..., \mathbf{x}_N$ that are gathered from experiences, we use this data point to train the model as a normal supervised learning task, we force the model to satisfy the physics by using a regularization term in the loss function, the same strategy is used to satisfy the boundary conditions. The idea is that generally the nonlinear part of PDEs involve doing nested operations to f_{NN} with respected to the input parameters, which auto-differentiation offered by machine learning libraries like TensorFlow and PyTorch make it straight forward.

Formally

Data Gathering

According to the problem at hand, various strategies could be used to gather relevant data points, e.g if data is hard to get we could use invariants that are present in the physical phenomena e.g scale and orientation to augment the size of data set

Network Architecture

A neural network $f_{NN}(x, y, z, t; \theta)$ is used to approximate to a function $f(x, y, z, t)$ for some phenomena. Note : we could choose from a variety of architecture but some architectures are more suitable for some problems

Loss Function

We guide the network to the right solution by adding regulations terms. The loss function is as follow :

$$\mathcal{L}_{PINN} = \lambda_d \mathcal{L}_{Data} + \lambda_r \mathcal{L}_{PDE} + \lambda_b \mathcal{L}_{BC} + \lambda_i \mathcal{L}_I \tag{3.2}$$

where

$$\mathcal{L}_{Data} = \frac{1}{N_d} \sum_{i=1}^{N_d} \|f(x_i, y_i, z_i, t_i) - f_{NN}(x_i, y_i, z_i, t_i; \theta)\|^2 \tag{3.3}$$

represents normal supervised learning.

$$\mathcal{L}_{PDE} = \frac{1}{N_r}\sum_{i=1}^{N_r} \| \frac{\partial f_{NN}(x_i, y_i, z_i, t_i; \theta)}{\partial t} + \mathcal{N}[f_{NN}, C; \theta] \|^2 \tag{3.4}$$

represents the physical constraints, this term gets minimal if the network satisfies the partial differential equations.

$$\mathcal{L}_{BC} = \frac{1}{N_b}\sum_{i=1}^{N_b} \| f(x_i, y_i, z_i, t_i) - f_{NN}(x_i, y_i, z_i, t_i; \theta) \|^2 \tag{3.5}$$

represents the initial conditions loss

$$\mathcal{L}_{I} = \frac{1}{N_i}\sum_{i=1}^{N_i} \| f(x_i, y_i, z_i, t_0) - f_{NN}(x_i, y_i, z_i, t_0; \theta) \|^2 \tag{3.6}$$

this terms are similar to the data loss term ,it informs the network about the behavior of the phenomena in boundaries and initial conditions.

Note: the $\lambda_{x\in\{d,r,b,i\}}$ are used to control the contribution of the corresponding regularization term.

Optimization

Any optimization algorithm could be used but Adam optimizer is usually the choice.

3.2.2 Use cases of PINN

The applications of Physics-Informed Neural Networks (PINNs) span an impressively broad spectrum, demonstrating their versatility and power across various scientific and engineering domains. From intricate fluid mechanics problems to cutting-edge deep reinforcement learning tasks, PINNs have proven to be a valuable tool in integrating physical knowledge with data-driven approaches. This wide-ranging applicability is comprehensively explored in "Physics-Informed Machine Learning: A Survey on Problems, Methods and Applications" by Hao *et al.* [9]. This seminal work provides a detailed and insightful overview of the diverse applications of physics-informed machine learning techniques, with a particular focus on PINNs. The survey not only catalogues the extensive use cases

but also delves into the various methodological approaches within the physics-informed machine learning paradigm. By examining both the practical implementations and theoretical foundations, the survey offers a complete view of the field, highlighting how PINNs and related techniques are revolutionizing the intersection of physical sciences and artificial intelligence. While the effectiveness and diverse utility of PINNs have been extensively highlighted, it is crucial to examine the various innovative approaches researchers have taken within this framework. Berg *et al* [2]., for instance, applied PINNs to predict fluid dynamics in complex geometries, concluding that the framework's performance is optimized with deeper networks and extended training epochs. This insight underscores the importance of architectural decisions in PINN implementations. In the context of medical imaging, Shone *et al* [19]. enhanced the vanilla PINN framework with loss reweighting techniques to predict complex synthetic blood flows, specifically addressing super-resolution challenges in 4D Flow MRI. Their work demonstrates the adaptability of PINNs to specialized medical applications. Arzani *et al*[1]. further extended the application of vanilla PINNs to the intricate task of computing near-wall blood flow and wall shear stress in complex aneurysm geometries, showcasing the framework's capability in handling highly specific and critical medical scenarios. Taking a different approach, Zhao *et al* [26]. introduced a novel PINN variant based on the Transformer architecture, designed to account for complex temporal relations in fluid dynamics. Chiu *et al.* [3] used PINN framework to accuratly model electrophysiology in 3D geometries in fibrillatory conditions(a type of cardiac arrhythmia characterized by rapid, irregular, and uncoordinated contractions of heart muscle fibers). This innovations highlights the ongoing evolution of PINNs, as researchers continue to integrate advanced machine learning concepts to enhance the framework's capabilities in capturing intricate physical phenomena across various domains.

3.2.3 Advantages of PINN

Physics-Informed Neural Networks (PINNs) are a powerful approach for solving problems that are governed by partial differential equations (PDEs) and other types of differential equations. They have several advantages compared to traditional numerical methods and other machine learning approaches:

1. Incorporation of Physical Laws : PINNs incorporate physical laws directly into the loss function, ensuring that the solutions respect known physical principles. By embedding the PDEs into the neural network's training process, PINNs can leverage known physics to guide the learning process, leading to more

accurate and reliable solutions.

2. High-Dimensional Problems : PINNs can efficiently handle high-dimensional problems. Traditional numerical methods suffer from the "curse of dimensionality," where computational cost grows exponentially with the number of dimensions. PINNs, leveraging the power of neural networks, can manage high-dimensional spaces more effectively.

3. Parallel Computing : PINNs can leverage modern parallel computing architectures. Neural networks can be trained on GPUs or TPUs, which allows for efficient parallel computation and significantly speeds up the training process compared to traditional CPU-based numerical methods.

4. Reduced Computational Cost for Complex Simulations : PINNs can reduce computational costs for certain complex simulations. For problems where traditional methods are computationally expensive, PINNs can offer a more efficient alternative, especially in cases requiring repeated simulations under varying conditions.

3.2.4 Disadvantages of PINN

While PINNs offer many advantages, they also come with certain disadvantages and limitations. Here are some of the key challenges and drawbacks associated with PINNs:

1. Training Complexity and Stability :

Training PINNs can be complex and unstable. The loss functions in PINNs often involve multiple terms (e.g., data loss, boundary condition loss, and physics loss), which can lead to challenges in balancing these terms. Training might require careful tuning of hyperparameters and can be sensitive to initial conditions and network architecture.

2. Lack of Convergence Guarantee :

PINNs do not always guarantee convergence to the correct solution, Due to the complexity of the loss landscapes in neural networks and the intricacies of incorporating physical laws, PINNs might converge to local minima or suboptimal solutions. The optimization process can be highly sensitive to initial conditions, hyperparameters, and network architecture, making it challenging to ensure that the network will find the correct solution.

3. Approximate Solutions :

PINNs often provide approximate rather than exact solutions to differential equations. PINNs typically produce approximate solutions. The accuracy of these solutions depends on various factors, including the quality of the training data,

the complexity of the neural network, and the effectiveness of the optimization algorithm. In practice, achieving high precision may require extensive tuning and significant computational resources.

4. Limited Generalizability :

PINNs may not generalize well beyond the conditions for which they were trained. While PINNs can interpolate within the range of the training data, their ability to extrapolate to unseen conditions or different physical scenarios is limited. This is particularly problematic when the system behavior outside the training range is critical for practical applications. The neural network may fail to capture the correct physical behavior in regions where it has not been trained, leading to unreliable predictions.

3.3 PINN Variants

The following section explores different techniques that could mitigate the drawbacks of PINN;

3.3.1 Loss Reweighing

Reweighing is a technique employed in PINNs to solve the training stability problem [23], by dynamically adjust the weights of loss terms. This adjustment prevents any individual loss term, particularly the one representing differential equations, from dominating the training process. Formally, we compute the maximum update value for each neural network weight corresponding to a particular loss term. Then, we compare this value with the update value according to the differential equations. The ratio obtained provides the necessary information for dynamically updating the weights of the loss terms.

$$\hat{\lambda}_x = \frac{\max\{\nabla_\theta \mathcal{L}_x\}}{\nabla_\theta \mathcal{L}_r} \tag{3.7}$$

$$\lambda_x = (1 - \alpha)\lambda_x + \alpha\hat{\lambda}_x \tag{3.8}$$

3.3.2 Data Resampling

The concept of resampling involves dynamically selecting points at which we enforce physics Das *et al.* [4]. These points can be chosen randomly each time or according to a specified distribution. Moreover, we can refine this approach

by adjusting the probabilistic distribution so that areas with higher error rates are more likely to be selected for enforcing physics.

3.3.3 Novel Objective

The concept aims to leverage the higher-order derivatives of the differential equations loss term to accelerate the convergence of the model.

3.3.4 Novel Architecture

Numerous approaches are employed to tailor architectures for problems intertwined with physics. These include optimizing activation functions, utilizing embeddings, employing multiple neural networks, employing sequential and convolutional architectures, and employing domain decomposition strategies.

Activation functions

In general ReLu, tanh, Sigmoid and sin are the most used activation functions used in neural network, in the context of PINN usually functions that are differentiable multiple times are chosen. The Swish activation function is used as a smoothed approximation of ReLU

$$\text{Swish}(x) = x.\,\text{Sigmoid}(\beta\, x) \tag{3.9}$$

β is a hyperparameter.
Some studies suggest using adaptive activation functions Jagtape *et al.* [11],

$$\sigma(na.x) \tag{3.10}$$

such that n is a hyperparameter and a is a learnable parameter.

Sequential architectures

Sequential architectures such as RNN, LSTM, or GRU offer the potential to enhance the model's capability to capture the intricate dynamics within physical systems.

Convolutional architectures

Additionally, CNN architectures can provide a spatially broad solution rather than focusing on individual points. Combining convolutional and sequential architectures may present a logical approach for addressing certain problems.

Transformer architecture

Physics-Informed Neural Networks (PINNs), which typically rely on multilayer perceptrons (MLP), often overlook the important temporal dependencies that are fundamental in practical physics systems. This oversight can lead to difficulties in accurately propagating initial conditions across the system and capturing the correct solutions under different circumstances. In Zhao *et al.* (2024) [26] paper, a new Transformer-based framework called PINNsFormer 3.3.4 was introduced, designed to overcome these limitations. PINNsFormer utilizes multi-head attention mechanisms to effectively capture temporal dependencies and accurately approximate solutions to partial differential equations (PDEs). It transforms point-wise inputs into pseudo sequences and adopts a sequential loss instead of the typical point-wise PINNs loss. Additionally, PINNsFormer incorporates a novel activation function named Wavelet, $wavelet(x) = w_1 \sin(x) + w_2 \cos(x)$ designed to facilitate Fourier decomposition within deep neural networks. Empirical results show that PINNsFormer offers enhanced generalization capabilities and accuracy in various scenarios, including those where traditional PINNs struggle and in solving high-dimensional PDEs. Furthermore, PINNsFormer is versatile, allowing for the integration of existing learning approaches for PINNs, which boosts its overall effectiveness.

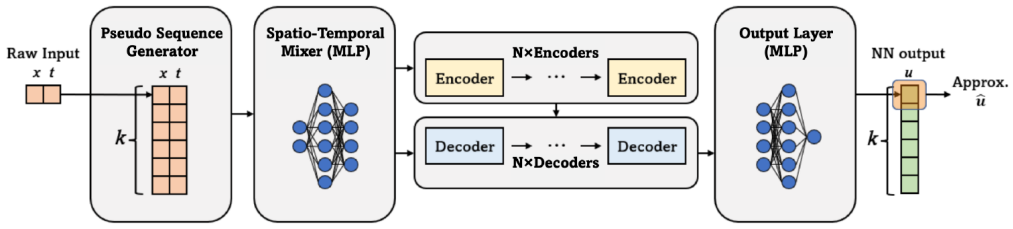

Figure 3.1: PINNsFormer architecture

Chapter 4

Methodology

4.1 Introduction

This chapter describes our implementation and testing of a super-resolution method for 4D Flow MRI, based on the work presented in Fathi *et al.* [6].

Our work focused on implementing and validating this method through a series of steps:

- Poiseuille Flow
- Navier-Stokes Equations
- Gaussian Quadratures
- Complex Image Representation
- Test on Poiseuille Flow
- Test on CFD

4.1.1 Navier-Stokes Equations in Neural Network Framework

We implemented the Navier-Stokes equations within our neural network framework as described in Fathi's paper.

Validation Test:

- We selected points exclusively from the surface of the pipe, including the base discs.

- Using these boundary points, we trained our neural network to predict the velocity field inside the pipe guided mainly by the Navier stokes residual.
- We then compared the network's predictions with the analytical Poiseuille flow solution.
- Convergence of the network predicted velocity field to the known analytical solution confirmed the correct implementation of this foundational step.

4.1.2 Gaussian Quadrature for MRI Signal Simulation

We implemented Gaussian Quadrature to simulate how MRI machines measure average velocity. Appeal to Appendix C.4 for more information about Gaussian Quadrature.

Validation Test:

- We tested our Gaussian quadrature implementation on functions with known analytical solutions.
- we used these quadratures to compute average velocities from our neural network outputs and verified that they matched expected patterns for Poiseuille flow averaged.

4.1.3 Complex Image Representation of MRI Signals

Following Fathi's approach, we implemented a method to represent MRI signals as complex images, with magnitude representing signal strength and phase encoding velocity information.

Validation Test:

- We generated complex images from known velocity fields (e.g., Poiseuille flow) and verified that the magnitude and phase correctly represented the expected signal strength and velocity patterns.
- We then trained our neural network using these complex images as input and verified that it could accurately reconstruct the original velocity fields.

4.1.4 Noise Robustness Testing

Throughout our implementation, we tested the method's robustness to noise.

Validation Test:

- We added 5% of maximum velocity noise to our data in the fidelity term.
- This noisy data was used in training and testing across all previous steps (Poiseuille flow, Navier-Stokes implementation, Gaussian quadrature, and complex image representation).
- We verified that our implementation could still produce accurate results despite this added noise, demonstrating robustness to realistic MRI noise levels.

4.1.5 Validation with Simulated Aortic Flow

As a final test, we applied our implementation to a more physiologically relevant scenario: simulated blood flow in an aorta Figure 4.1.5.

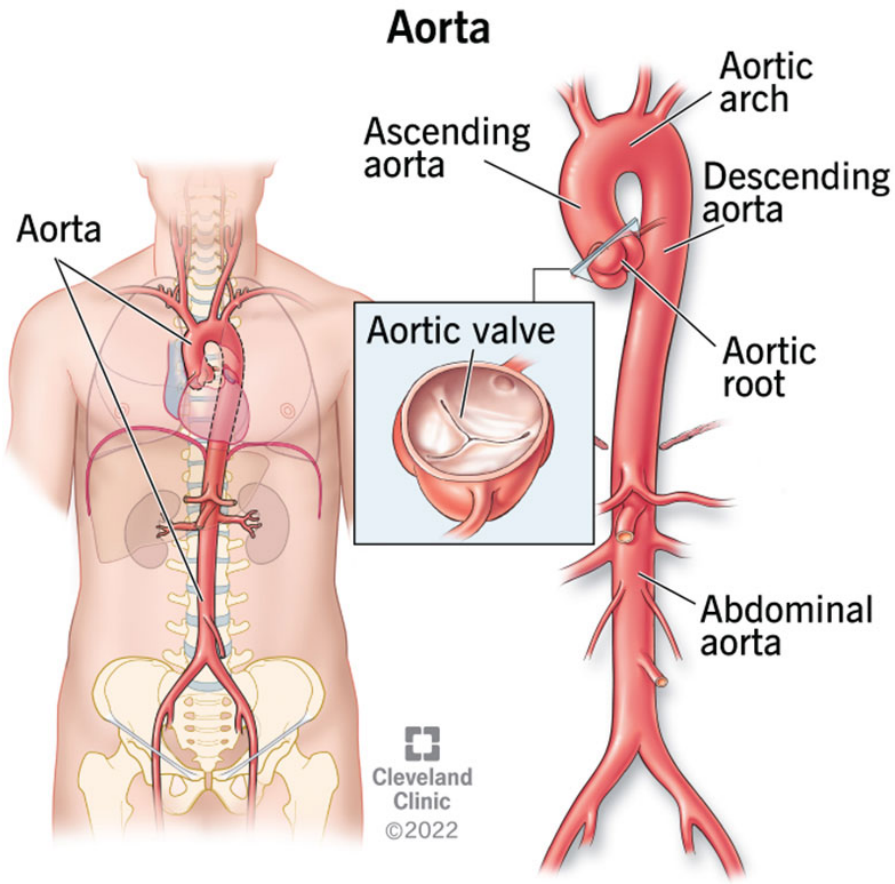

Figure 4.1: Aorta anatomy

Validation Test:

- We used synthetic data representing aortic flow, which is more complex than Poiseuille flow but still has known characteristics.
- We applied our full pipeline - from complex image generation to neural network reconstruction - to this aortic flow data.
- We evaluated our reconstructed flow fields by comparing them to the ground truth synthetic data.
- We also verified that the reconstructed flow adhered to expected physiological patterns and satisfied the Navier-Stokes equations.

Through these validation steps, we ensured that each component of our implementation of Fathi's super-resolution method for 4D Flow MRI was functioning correctly. This systematic approach allowed us to confidently reproduce the method's effectiveness in improving the resolution and accuracy of 4D Flow MRI data, potentially contributing to better analysis of blood flow patterns in cardiovascular studies.

In the following sections, we will begin by rigorously describing the Poiseuille Flow, Gaussian quadratures, Complex images, and the metrics used to evaluate the algorithm. For readers seeking a foundational understanding of fluid mechanics principles, we recommend referring to Appendix B.2.2, which provides a comprehensive overview of the subject.

4.1.6 Poiseuille Flow :

Poiseuille flow, also known as Hagen–Poiseuille flow, describes the laminar flow of a viscous, incompressible fluid through a cylindrical pipe with constant circular cross-section. This flow regime is characterized by a parabolic velocity profile, with the maximum velocity at the center of the pipe and zero velocity at the walls due to the no-slip condition. The flow is driven by a pressure gradient along the pipe's axis, and its behavior is governed by the Navier-Stokes equations. In Poiseuille flow, the velocity at any point in the pipe is directly proportional to the pressure gradient and the square of the radial distance from the pipe's center, and inversely proportional to the fluid's viscosity. This model is widely used in various fields, including physiology to understand blood flow in vessels, and in engineering applications involving pipe flows. The simplicity of its analytical solution makes Poiseuille flow an ideal benchmark for testing computational fluid dynamics models and algorithms.

Imagine blood flow through a pipe of length L and radius R We define a systeme of Cartesian coords in which the center is the center of the pipe x,y axis define the transverse planes and z axis is the direction of the flow.

The following formula is the poiseuille flow :

$$\vec{V}(t,x,y,z) = \vec{u}(t,x,y,z), \vec{v}(t,x,y,z), \vec{w}(t,x,y,z) \tag{4.1}$$

$$\vec{u}(t,x,y,z) = \vec{v}(t,x,y,z) = 0 \tag{4.2}$$

$$\vec{w}(t,x,y,z) = \frac{\Delta P}{4\mu L}(R^2 - x^2 - y^2) \tag{4.3}$$

Where ΔP is the difference of pressure between the inlet and outlet of the pipe, and μ is the viscosity of blood.

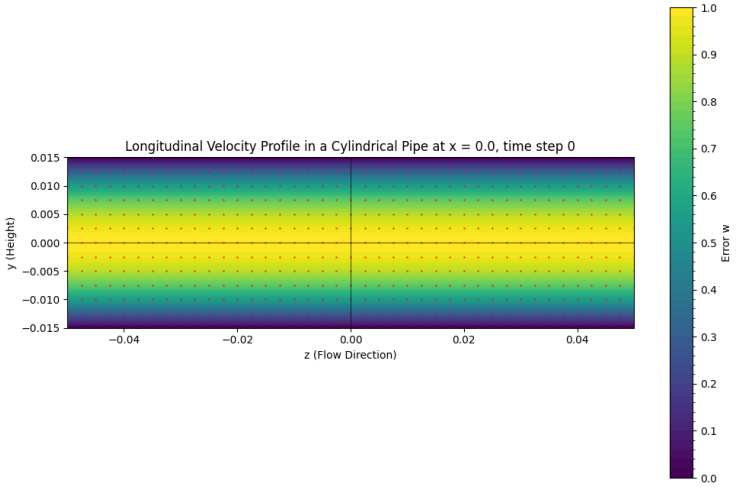

Figure 4.2: Longitudinal Velocity Profile in a Cylindrical Pipe at x=0, time step t=0

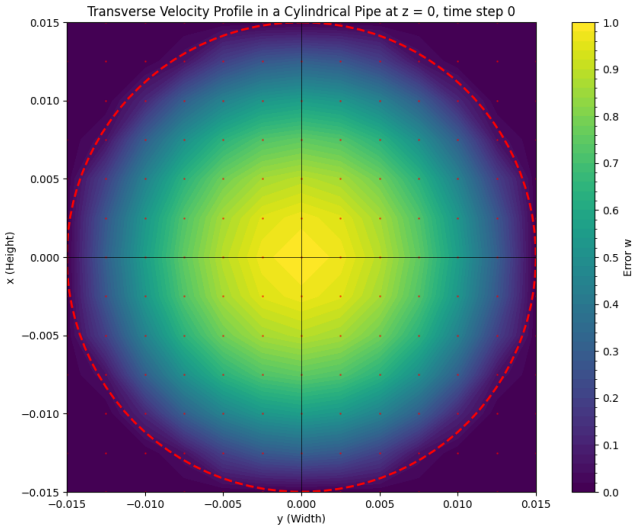

Figure 4.3: Transverse Velocity Profile in a Cylindrical Pipe at z=0, time step t=0

4.1.7 Normalization

Normalization plays a crucial role in the effective implementation of Physics-Informed Neural Networks (PINNs) for fluid dynamics problems . By expressing these complex equations in terms of dimensionless variables, normalization brings several advantages to the PINN framework. It enhances numerical stability by scaling variables to similar ranges, typically between 0 and 1, which prevents issues arising from vastly different magnitudes in the neural network's architecture. This scaling also improves learning efficiency, as neural networks generally perform better with uniformly scaled inputs and outputs. Moreover, normalized equations introduce important dimensionless parameters, such as the Reynolds number, which encapsulate the essential physics of the flow. This allows PINNs to capture fundamental flow characteristics without being tied to specific dimensional values, thereby improving their generalization capabilities across different flow scenarios. The normalized form also simplifies the implementation of physical constraints, like incompressibility, and makes it easier to balance the influence of various terms in the equations when formulating the loss function. Consequently, normalization not only aids in the numerical aspects of training but also helps PINNs to more accurately and efficiently represent the underlying physics

of fluid flow, making it an indispensable step in applying neural networks to complex fluid dynamics problems. Coordinates and Velocity filed also pressure will be Normalized according to the following formulas:

$$\begin{cases} x = L\hat{x} & u = U\hat{u} & t = T\hat{t} \\ y = L\hat{y} & v = U\hat{v} & p = \rho U^2\hat{p} \\ z = L\hat{z} & w = U\hat{w} & \end{cases} \tag{4.4}$$

4.1.8 Incompressible Navier Stokes Equations

The Incompressible Navier-Stokes equations are a set of partial differential equations that describe the motion of incompressible fluids. They are fundamental to fluid dynamics and are used to model a wide range of phenomena, from blood flow in veins to weather patterns in the atmosphere.

Components of Equations

1. **Two Navier-Stokes Equations:** These govern the momentum of the fluid.
2. **Conservation of Mass Equation:** Also known as the continuity equation.

The following are the Incompressible Navier Stokes Equations in vectorized form:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) = -\boldsymbol{\nabla} p + \mu\nabla^2\mathbf{u} + \mathbf{F} \tag{4.5}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u} = 0 \tag{4.6}$$

F represents external forces(gravitation or electromagnetic forces) which will be neglected, Appendix B.2.2 present the derivation of incompressible Navier Stokes equations.

Normalized Navier Stokes equations

$$\frac{L}{UT}\frac{\partial \hat{\mathbf{u}}}{\partial \hat{t}} + \hat{\mathbf{u}}\cdot\nabla\hat{\mathbf{u}} = -\nabla\hat{p} + \frac{\mu}{\rho UL}\nabla^2\hat{\mathbf{u}} \tag{4.7}$$

$$\boldsymbol{\nabla}\cdot\hat{\mathbf{u}} = 0 \tag{4.8}$$

Navier Stokes Equations (scalar form)

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right) \tag{4.9}$$

$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}\right)=-\frac{\partial p}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right) \tag{4.10}$$

$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=-\frac{\partial p}{\partial z}+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right) \tag{4.11}$$

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{4.12}$$

Normalized Navier Stokes Equations (scalar form)

$$E_u=\frac{L}{UT}\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}+\frac{\partial p}{\partial x}-\frac{\mu}{\rho UL}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)=0 \tag{4.13}$$

$$E_v=\frac{L}{UT}\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}+\frac{\partial p}{\partial y}-\frac{\mu}{\rho UL}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right)=0 \tag{4.14}$$

$$E_w=\frac{L}{UT}\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}+\frac{\partial p}{\partial z}-\frac{\mu}{\rho UL}\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)=0 \tag{4.15}$$

$$E_c=\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{4.16}$$

This is the form of navier stokes equations that will be used to enforce physics in the neural networks.

4.2 Algorithm Pipeline

This section showcases the algorithm presented in Fathi *et al.* [6], see Figure 4.2 which represent the algorithm in the form of a pipeline, in the following we will briefly explain each part.

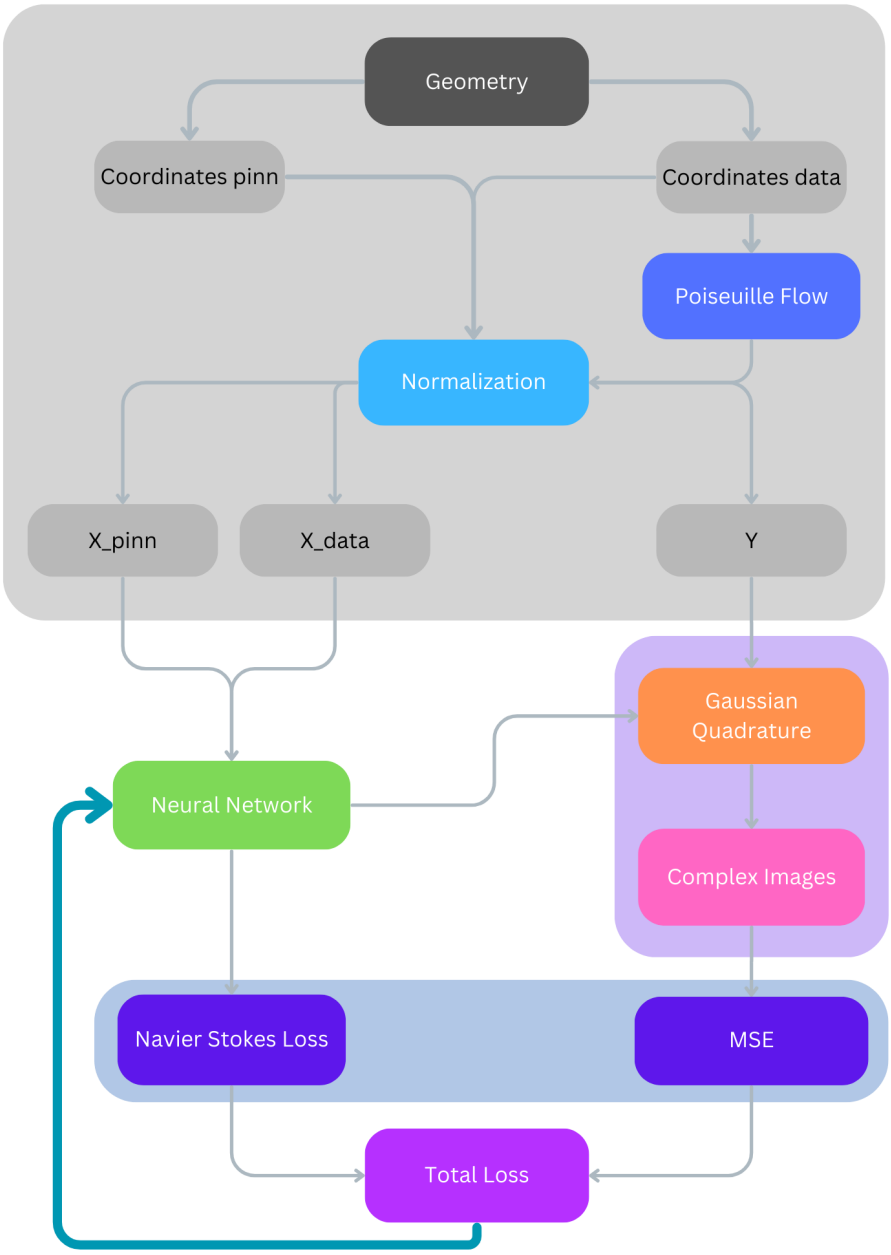

Figure 4.4: The complete pipeline of the algorithm to be tested

1. **Geometry:** Defines the physical structure or domain for fluid flow study.
2. **Coordinate Systems:**
    - Coordinates pinn: Discretized coordinates for the PINN model.
    - Coordinates data: Coordinate system of actual or simulated data.
3. **Poiseuille Flow:** Incorporates Analytical Poiseuille flow solution.
4. **Normalization:** Ensures all data is on a comparable scale for neural network training.
5. **Neural Network (NN):** The core PINN model trained to solve the fluid flow problem, see Figure 4.2.1.
6. **Gaussian Quadrature:** Numerical integration technique for efficient computation of averages or integrals.
7. **Complex Images:** Processing of flow data into complex-valued images, similar to 4D Flow MRI.
8. **Loss Functions:**
    - Navier-Stokes Loss: Enforces physics constraints based on Navier-Stokes equations.
    - MSE (Mean Squared Error): Compares predicted flow complex representation with expected results.
9. **Total Loss:** Combination of Navier-Stokes Loss and MSE for network training.
10. **Feedback Loop:** Iterative process updating network parameters based on computed loss.

4.2.1 Network Architecture

The neural network architecture employed in this study is a straightforward fully connected design, comprising an input layer, 4-8 hidden layers, and an output layer. The input layer accepts a 4D vector representing the normalized spatio-temporal coordinates. Each of the four hidden layers consists of 29 neurons and

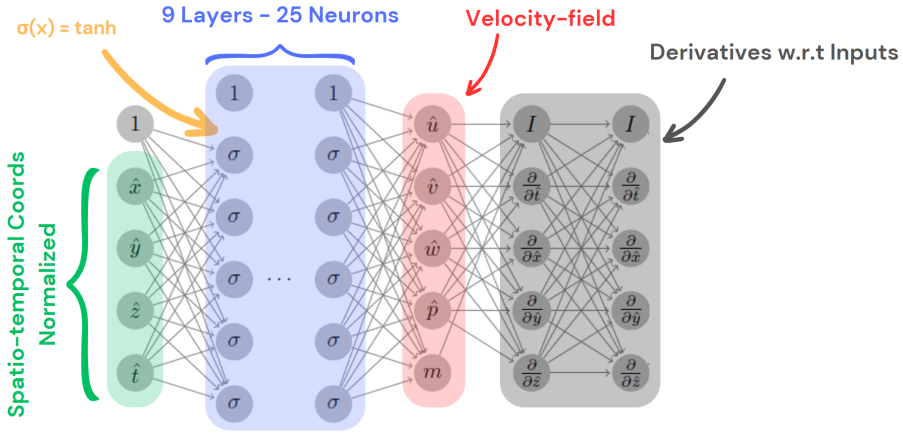

Figure 4.5: The structure of the neural network

utilizes the hyperbolic tangent (tanh) activation function, which introduces non-linearity and allows the network to learn complex patterns. The output layer produces a 5D vector, encompassing three components of the velocity field, the pressure, and the MRI signal. This architecture enables the network to map the input coordinates to the desired flow characteristics comprehensively. The output of the network is not only used directly but also serves as a basis for calculating additional quantities of interest, such as gradient or Laplacian with respect to the input, which are essential for enforcing physical constraints and evaluating the flow behavior in subsequent analyses.

4.2.2 Average Velocity Via Gaussian Quadratures

The simulation of 4D Flow MRI data using average velocities as targets represents a sophisticated approach to bridging the gap between high-resolution fluid dynamics simulations and the realities of medical imaging. This method involves calculating the average velocity for each voxel by integrating over a small surrounding volume in both space and time, mimicking the data acquisition process in MRI. However, computing this 4D integral directly is computationally intensive and often impractical. To address this challenge, Gaussian Quadrature is employed as an efficient numerical approximation technique. This approach carefully selects specific sampling points within the 4D space-time volume of each voxel and assigns them appropriate weights, allowing for an accurate estimation of the average velocity with significantly reduced computational cost. By uti-

lizing this method, researchers can create more realistic simulations that closely align with actual MRI measurements, potentially improving the accuracy of blood flow models and enhancing the development of diagnostic tools in cardiovascular medicine. Furthermore, this technique offers a valuable framework for validating computational fluid dynamics models against clinical MRI data, thereby advancing our understanding of complex flow phenomena in the human body. the following integral define the average velocity in a point by the integral over the spatio-temporel coordinates of velocities around it:

$$\bar{\mathbf{u}}_{nn} = \frac{1}{\Delta t \Delta x \Delta y \Delta z} \int_{\Delta t} \int_{\Delta x} \int_{\Delta y} \int_{\Delta z} \mathbf{u}_{nn} \, dt \, dx \, dy \, dz \tag{4.17}$$

this integral can be approximated by the Gaussian quadrature :

$$\bar{\mathbf{u}}_{nn} \approx \frac{1}{\Delta t \Delta x \Delta y \Delta z} \sum_{i=0}^{s} \sum_{j=0}^{n} \sum_{k=0}^{m} \sum_{l=0}^{p} w_i w_j w_k w_l \, \mathbf{u}_{nn}(t_i, x_j, y_k, z_l) \tag{4.18}$$

For more details about the theoretical basis of Gaussian quadratures refer to Appendix C.

4.2.3 Complex Images

complex images is a way to simulate the acquisition process of 4D Flow MRI, we begin by defining *kr* which is directly linked to *v_enc* which represent the 3D vector that represents the maximum speed of flow in each direction. In practice we use v_enc is set to be the same for all directions.

$$k_r = \frac{1}{2 \, v_{enc}} \tag{4.19}$$

$$S^r_{NN} = \overline{m}_{NN} \cdot e^{-i2\pi k_r \cdot \overline{u}_{NN}} = \overline{m}_{NN}[\cos(2\pi k_r \cdot \overline{u}_{NN}) - i \sin(2\pi k_r \cdot \overline{u}_{NN})] \tag{4.20}$$

4.3 Metrics

The metrics used to evaluate the performance of the network are versatile and enable different viewpoints on the quality of the predictions. These metrics fall into two main categories, each serving a distinct purpose in assessing the network's output. The first category consists of similarity indices, which calculate the

similarity between the true velocity field and the predicted field. These indices provide a measure of how well the predicted field aligns with the actual field in terms of direction and magnitude. The second category comprises normalized mean square error metrics, which quantify the error between the velocity field and the predicted field. These error metrics offer a more direct measure of the discrepancy between the predicted and actual values, providing insight into the accuracy of the network's predictions. By employing both types of metrics, we can gain a comprehensive understanding of the network's performance, assessing both the overall similarity of the predicted flow patterns and the precise accuracy of the velocity predictions.

Normalized mean square error

speed Normalized Mean Root Square Error quantify the difference between the predicted and real speed.

$$\text{sNMRSE} = \frac{1}{\max|v_{ref}|}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(|v| - |v_{ref}|)_i^2} \tag{4.21}$$

velocity Normalized Mean Root Square Error quantify the angle between the predicted and real velocity.

$$\text{vNMRSE} = \frac{1}{\max|v_{ref}|}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v - v_{ref})_i^2} \tag{4.22}$$

Similarity Indices

ASI compute angle similarity between predicted and real velocity field, ASI=0 means predicted and real velocity are in different ways, ASI=1 they are facing the same direction.

$$\text{ASI} = \frac{1}{2}\left(1 + \frac{\mathbf{v}_{rec}\cdot\mathbf{v}_{ref}}{|\mathbf{v}_{rec}||\mathbf{v}_{ref}|}\right) \tag{4.23}$$

MSI compute the magnitude similarity between predicted and real velocity field.

$$\text{MSI} = 1 - \left|\frac{|\mathbf{v}_{rec}|}{\max(|\mathbf{v}_{rec}|)} - \frac{|\mathbf{v}_{ref}|}{\max(|\mathbf{v}_{ref}|)}\right| \tag{4.24}$$

SI quantify the overall similarity between the predicted and real velocity field.

$$\text{SI} = \text{ASI}\cdot\text{MSI} \tag{4.25}$$

4.4 Computational Resources

Training was conducted on both a high-performance server and a personal computer for 10,000 epochs. The server was equipped with an Intel(R) Xeon(R) Silver 4116 CPU 2.10GHz with 20 cores, and an NVIDIA GeForce RTX 2080 Ti GPU with 11GB of VRAM. The personal computer used was powered by an AMD Ryzen 9 5900HX processor running at 3.30 GHz, 32 GB of RAM, and an NVIDIA GeForce RTX 3070 GPU. This dual-setup approach allowed for efficient large-scale computations on the server while enabling rapid prototyping and testing on the local machine. The combination of these resources provided a robust environment for training our complex PINN models for 4D Flow MRI super-resolution and denoising tasks.

Chapter 5

Results

In this chapter we will present results we get on the test we conducted, in where we demonstrate the capabilities of the approach.

5.1 Poiseuille Flow

5.1.1 Straight Pipe

coordinates are only taken from the inside and the outside of the pipe, we can observe that the network near perfectly predict the poiseuille flow.

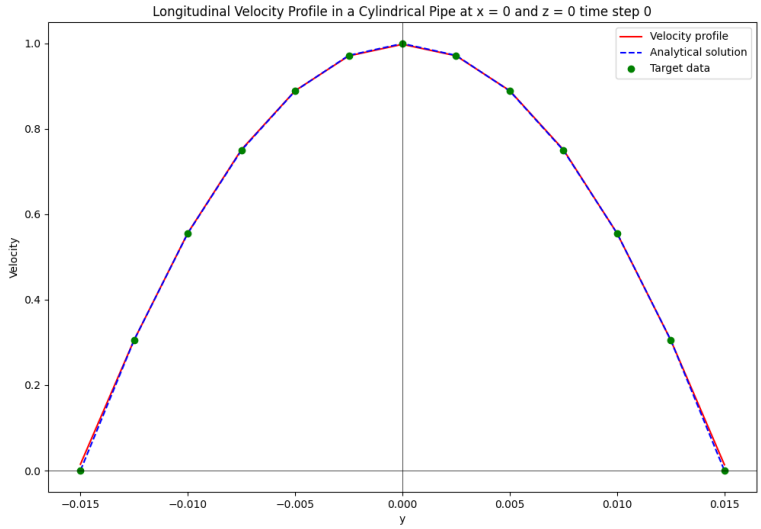

Figure 5.1: Comparison of Predicted and Analytical Velocity Profiles with Data in Poiseuille Flow

5.1.2 Straight Pipe + Noise

We made the previous test more difficult by adding noise of 5% of the maximum speed.

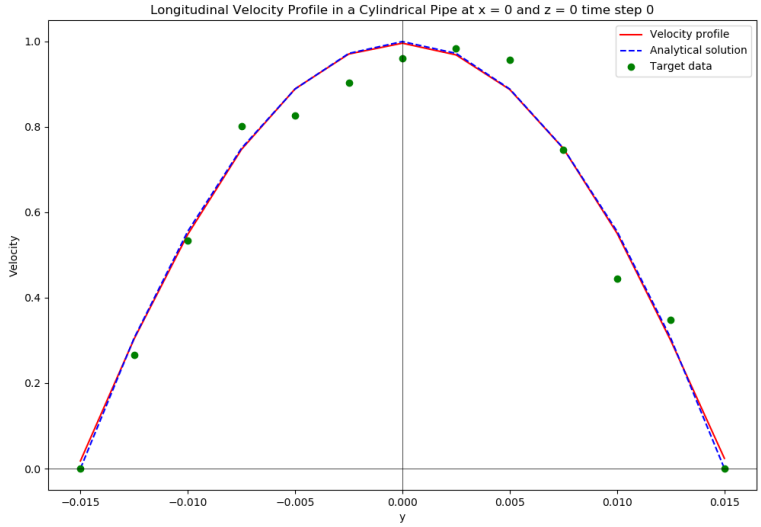

Figure 5.2: Comparison of Predicted and Analytical Velocity Profiles with Noisy Data in Poiseuille Flow

The close alignment of the network prediction (red line) with the analytical solution (blue dashed line) indicates the model's effectiveness in capturing the true flow behavior, even in the presence of noisy input data. The scattered green points illustrate how real data might deviate from the ideal solution due to measurement errors or natural fluctuations, yet the network successfully filters out this noise to produce a smooth, physically accurate velocity profile. This visualization effectively demonstrates the neural network's capability to reconstruct accurate flow profiles from noisy data in a Poiseuille flow scenario, which is a crucial step in validating the model for more complex flow situations.

5.1.3 Half Pipe

In this section, we present the results of a critical test designed to evaluate the generalization capabilities of our neural network model in the context of fluid flow prediction. Specifically, we examine the performance of the network when applied to a half-pipe scenario. The unique aspect of this test lies in our approach to data provision: the network is intentionally fed data from only one side of the pipe. This deliberate limitation serves as a subtle test of the model's ability to extrapolate and accurately predict flow patterns in unseen regions. By restricting the input to

half of the pipe's cross-section, we challenge the network to leverage its learned physics-informed principles and the partial flow information to reconstruct the complete flow profile. This test is particularly significant as it assesses not just the network's ability to interpolate within known data points, but its capacity to generalize and predict flow behavior in entirely unseen sections of the pipe. The results of this half-pipe test provide crucial insights into the robustness and versatility of our model, especially in scenarios where complete data coverage may not be available or practical to obtain

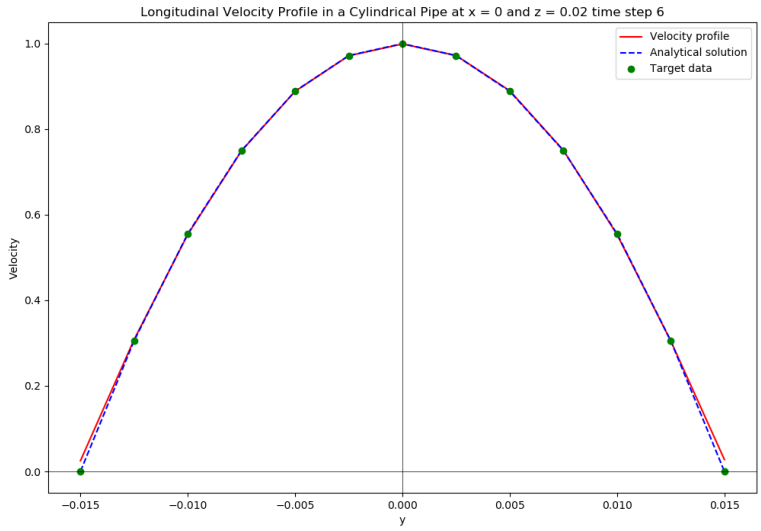

Figure 5.3: Comparison of Predicted and Analytical Velocity Profiles with Noisy Data in Poiseuille Flow in half Pipe case

5.1.4 Quantitative Results

We will present the performance of the algorithm with respect to the metrics previously showcased:

	RMS div	vNRMSE	sNRMSE	ASI	MSI	SI
Straight Pipe	0.0042	0.0034	0.0058	0.9999	0.9961	0.99604
Straight Pipe + 5% noise	0.0033	0.0038	0.0063	0.9999	0.9964	0.9964
Half Straight Pipe + 5% noise	0.0013	0.0046	0.0045	0.9999	0.9966	0.9966

Figure 5.4: Comparison of Predicted and Analytical Velocity Profiles with Noisy Data in Poiseuille Flow in half Pipe case

The results demonstrate the robustness and effectiveness of our Physics-Informed Neural Network (PINN) approach across various flow scenarios. In the straight pipe case, the model achieves excellent performance with very low error rates (RMS div: 0.0042, vNRMSE: 0.0034, sNRMSE: 0.0058) and near-perfect similarity indices (ASI: 0.9999, MSI: 0.9961, SI: 0.99604). Notably, when 5% noise is introduced to the straight pipe scenario, the model maintains its high performance, with only slight increases in error rates and consistently high similarity indices. This indicates strong noise resilience. The half straight pipe case with 5% noise showcases the model's ability to generalize, as it maintains low error rates and high similarity indices even when predicting flow in unseen regions. The consistently high ASI (0.9999) across all scenarios underscores the model's accuracy in predicting flow directions. Overall, these results validate the PINN's capability to accurately reconstruct flow fields, even in the presence of noise and partial data, which is crucial for enhancing 4D Flow MRI data in clinical applications. In summary the algorithm achieved less than 1% noise and near perfect score in similarity indices in all Poiseuille flow test.

Chapter 6

Conclusion & Discussion

This study has implemented and evaluated a super-resolution and denoising method for 4D Flow MRI based on Physics-Informed Neural Networks (PINNs), as proposed by Fathi et al. [6]. Our research has yielded several key findings we successfully implemented the PINN framework for fluid dynamics problems, incorporating the Navier-Stokes equations as physical constraints. The model demonstrated the ability to predict flow patterns accurately, particularly in the case of Poiseuille flow. Our implementation of Gaussian Quadrature for simulating MRI signal averaging proved effective, allowing for a more accurate representation of the MRI acquisition process within the PINN framework. The method of representing MRI signals as complex images, with magnitude representing signal strength and phase encoding velocity information, was successfully implemented and validated. The PINN-based approach showed promising results in handling noisy input data, demonstrating its potential for denoising 4D Flow MRI. The model was tested on various pipe geometries (straight, tilted, and half-pipe), showing good generalization capabilities, particularly in the half-pipe scenario where it accurately predicted flow in unseen regions. In the course of developing this work, we faced numerous difficulties. First, it's hard to find the weights that balance the fidelity loss term and the Navier-Stokes loss term, which tend to steer the learning towards a solution that satisfies the Navier-Stokes equations but does not closely match the data. PINNs are also sensitive to learning rate choices. Additionally, there was no clear method to test the implementation of the Navier-Stokes equations. Thanks to Mr. Sebastien's brilliant idea to use only points on the surface of the pipe and observe if the network converges to Poiseuille flow inside the pipe, guided only by the Navier-Stokes loss term, we were able to validate our implementation. Another problem is that while computing integrals by Gaussian

quadrature is efficient when dealing with simple functions, it becomes challenging for complex functions. In these cases, you need to use more Legendre polynomials in each dimension to accurately capture the complex patterns in the function's behavior. If n is the number of Legendre polynomials used for each dimension, you will need 2^n weights for each voxel. For example, if you choose 2 Legendre polynomials to be used for each dimension in a 4-dimensional space, you have $2^4 = 16$ weights for each voxel which exponentially increases computation.

6.1 Discussion of Results

6.1.1 Effectiveness of PINN for 4D Flow MRI

The successful implementation and validation of the PINN-based approach for 4D Flow MRI super-resolution and denoising demonstrate the potential of this method in improving medical imaging techniques. The ability of PINNs to incorporate physical laws (in this case, the Navier-Stokes equations) provides a significant advantage over traditional deep learning approaches, especially in scenarios where data might be limited or noisy.

6.1.2 Implications for Clinical Applications

The improved resolution and reduced noise in 4D Flow MRI data could have substantial clinical implications. Enhanced image quality could lead to more accurate diagnoses of cardiovascular conditions, better treatment planning, and improved monitoring of disease progression. The ability to generate high-resolution flow fields from lower-resolution data could also potentially reduce MRI acquisition times, improving patient comfort and reducing healthcare costs.

6.1.3 Challenges and Limitations

Despite the promising results, several challenges and limitations were identified, the PINN approach, while effective, is computationally intensive, particularly for complex geometries and high-resolution outputs. Finding the optimal balance between data fidelity and physical constraints in the loss function required careful tuning. While the method performed well for Poiseuille flow and showed promise for aortic flow, its effectiveness for more complex, turbulent flows remains to be fully validated. Our study primarily used simulated data. The performance of the

method on real-world 4D Flow MRI data, with all its inherent complexities and artifacts, needs further investigation.

6.2 Future Research Directions

Based on our findings and the limitations identified, we propose several directions for future research, explore more sophisticated PINN architectures, such as those incorporating attention mechanisms or adaptive activation functions, to improve performance and reduce the need for parameter tuning, appeal to Appendix D.1.5 for a deep dive into the flaws of PINN and the proposed solutions. Also investigation of transfer learning techniques to adapt models trained on simulated data to real-world 4D Flow MRI data is needed. Another interesting idea is developing methods to quantify uncertainties in the super-resolved and denoised outputs, which is crucial for clinical decision-making. Conducting comprehensive comparisons with other super-resolution and denoising techniques, including both traditional and deep learning-based methods is also needed.

6.3 Concluding Remarks

This research has demonstrated the potential of Physics-Informed Neural Networks in enhancing 4D Flow MRI through super-resolution and denoising. By leveraging the power of deep learning while respecting fundamental physical principles, this approach offers a promising path towards more accurate and detailed cardiovascular imaging. As we continue to refine these methods and address the identified challenges, we anticipate significant advancements in both the technical capabilities of medical imaging and its clinical applications. The integration of physics-based modeling with data-driven approaches represents a frontier in medical image analysis, holding the promise of improved patient care and deeper insights into cardiovascular health.

Appendix A

Magnetic Resonance Imaging

A.1 Basic Principle of MRI

MRI machines utilize the quantum properties of hydrogen atoms, particularly their spin, which acts like a tiny bar magnet. Normally, the magnetic fields of these hydrogen atoms cancel each other out. However, when placed in an MRI machine's strong magnetic field, these atoms align along with the direction of this field.

A.2 Activation of Hydrogen Atoms

The MRI machine generates a brief magnetic pulse using coils that carry alternating currents. This pulse slightly misaligns the hydrogen atoms from their aligned position, causing them to shift their magnetic fields perpendicular to the machine's main magnetic field.

A.3 Signal Generation

After being nudged, the hydrogen atoms want to return to their initial aligned state. As they return, they don't immediately fall back into alignment but instead spiral back, which causes a change in the magnetic field. This change induces an electrical current in nearby coils, which is then captured as a signal by the MRI machine.

$$\oint_{\partial\Sigma} \vec{B}.\vec{dl} = \mu_0(\epsilon_0 \frac{d}{dt} \iint_{\sigma} \vec{E}.\vec{ds} + \iint_{\sigma} \vec{J}.\vec{ds})$$

A.4 Coils in MRI Machines

MRI machines might use the same set of coils for sending magnetic pulses and reading the returning signals. Some advanced machines use separate coils for transmitting and receiving to improve the image's contrast and resolution.

A.5 Importance of Magnetic Field Strength

The strength of the MRI's magnetic field can significantly impact the quality of the image. Common field strengths for clinical MRIs range from 1.5 to 3 Tesla, but research MRIs may use fields up to 20 Tesla. Higher magnetic fields provide a stronger signal and better image quality.

$$w = -\gamma B$$

w : Rotational Frequency γ : Gyromagnetic Ratio B : Magnetic Field strength

A.6 Superconducting Coils

To create such high magnetic fields, MRI machines use superconducting coils made from materials like Niobium-Titanium. These coils require cooling with liquid helium within a vacuum-sealed chamber to maintain superconductivity without overheating.

A.7 Energy Consumption

The main energy requirement for an MRI machine is not from generating the magnetic field but from keeping the superconducting coil cool enough to maintain its superconductivity. The energy used to operate an MRI machine for a year is similar to the amount used by 25 average households.

A.8 MRI Imaging Technique

Instead of capturing individual points or pixels, MRI scans sample patterns of signals from hydrogen atoms in the body. By varying the timing and intensity of magnetic gradients, MRIs create and capture different striped patterns across various directions and frequencies. These patterns are combined to form a detailed image of a slice of the body.

A.9 Image Formation and Slices

MRI technology collects signals in slices, layer by layer. Each image slice is built from a complex overlay of various striped patterns, revealing different tissues based on their unique characteristics like T1 and T2 relaxation times. These times indicate how quickly hydrogen atoms in different types of tissues realign and lose uniformity after a magnetic pulse, which affects how the signal decays and is ultimately visualized.

In the end, for more information i strongly suggest MRI in Practice Catherine Westbrook 4th edition [25] and specially the first 3 chapters.

Appendix B

Fluid Mechanics

B.1 Definitions

Fluid

Fluid is a substance that deforms continuously under the application of shear stress.

Fluid Continuum

In fluids mechanics, their is a minimal volume that under it our calculations are no more valid and quantum mechanics need to be invoked. For example mass density in a point is defined as follow :

$$\rho = \lim_{\Delta V \to \Delta V'} \frac{\Delta m}{\Delta V}$$

where $\Delta V'$ is the minimal volume in which the classical formulation of physics are still valid

Fluid Flow

Assuming the continuum hypothesis ,Fluid flow is a vector Field that maps position and time to velocity vector of the fluid in that position in time ,i.e

$$\vec{V} = \vec{v}(x, y, z, t)$$

Note: for simplicity we assume Cartesian coordinates

B.1.1 Laminar and Turbulent Flow

The Reynolds number (Re) is a dimensionless quantity in fluid mechanics that predicts flow patterns in different fluid flow situations. It is defined as the ratio of inertial forces to viscous forces within a fluid and is crucial in determining whether flow will be laminar or turbulent. A low Reynolds number (typically Re $<$ 2300 for pipe flow) indicates laminar flow, characterized by smooth, predictable fluid motion. Higher Reynolds numbers (Re $>$ 4000 for pipe flow) signify turbulent flow, marked by chaotic eddies and irregular fluctuations. The Reynolds number is calculated using the fluid's density, viscosity, velocity, and a characteristic linear dimension of the system. It finds widespread application in various fields, including aerodynamics, chemical engineering, and civil engineering, where it helps in designing pipes, aircraft, and other structures that interact with fluids.

Reynolds Number

$$Re = \frac{\rho UL}{\mu}$$

B.2 Navier-Stokes Equation

Navier-Stokes equation is a system of non-linear partial differential equations that describe how any fluid evolve through time.

B.2.1 Derivation Navier-Stokes Equation

Continuity equation

Continuity equation is an equation which describes the change an intensive property.Intensive property is a property that is independent of the amount of material e.g Temperature.

$$\frac{d}{dt}\int_{\Omega} L\,dv = -\int_{\partial\Omega} L\vec{v}.\vec{n}\,dS - \int_{\Omega} Q\,dv$$

such that $-\int_{\partial\Omega} L\vec{v}.\vec{n}\,dS$ is how much of property L is leaving the volume by flowing over boundary $\partial\Omega$, and $-\int_{\Omega} Q\,dv$ how much of the property is leaving the volume to sinks,informally speaking ,the change in the property L is due to

how much flows out through the volume boundary as well as how much is lost or gained through sources or sinks inside the boundary

Divergence Theorem

$$\int_{\partial\Omega} L\vec{v}.\vec{n}\, dS = \int_{\Omega} \nabla.(L\vec{v})\, dV$$

D.T allows the flux term to be expressed as a volume integral
the previous equation become

$$\frac{d}{dt}\int_{\Omega} L\, dv = -\int_{\Omega} (\nabla.(L\vec{v}) + Q)\, dV$$

Leibniz Rule

$$\frac{d}{dx}\int_{a}^{b} f(x,y)\, dy = \int_{a}^{b} \frac{d}{dx} f(x,y)\, dy$$

$$\int_{\Omega} (\frac{d}{dt}L + \nabla.(L\vec{v}) + Q)\, dV = 0$$

because we want the equation to lead the same result for any arbitrary volume we got :

$$\frac{d}{dt}L + \nabla.(L\vec{v}) + Q = 0$$

applying the continuity equation to density :

$$\frac{d}{dt}\rho + \nabla.(\rho\vec{v}) + Q = 0$$

if we have no sources or sinks $Q = 0$ we got :

$$\frac{d}{dt}\rho + \nabla.(\rho\vec{v}) = 0$$

for incompressible fluids , the density is constant .so

$$\nabla.\vec{v} = 0$$

Material Derivative

we introduce the following notation to simplify the equation

$$\frac{D}{Dt}u = \frac{d}{dt}u + (\vec{v}.\nabla)u$$

so the material derivative is the sum of normal rate of change w.r.t time and the directional derivative in direction of $\vec{v}$

B.2.2 Actual derivation of Navier-Stokes equation

Navier-Stokes general form

Now we have all the tools to derive the equation ,the Navier-Stokes equation is basically the second law of Newton ,substituting the mass by density , we get :

$$\vec{b} = \rho.\frac{d}{dt}\vec{v}(x, y, z, t)$$

using chain rule we get

$$\vec{b} = \rho.(\frac{\partial\vec{v}}{\partial t} + \frac{\partial\vec{v}}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial\vec{v}}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial\vec{v}}{\partial z}\frac{\partial z}{\partial t})$$

which is equivalent to

$$\vec{b} = \rho.(\frac{\partial\vec{v}}{\partial t} + \vec{v}.\nabla\vec{v})$$

which is exactly the definition of material derivative

$$\vec{b} = \rho.\frac{D\vec{v}}{Dt}$$

the body force on the fluid consists of two components fluid stresses and external forces

$$\vec{b} = \nabla.\sigma + \rho.\vec{f}$$

we have the fluid stresses σ consists of pressure P and stress T so

$$\sigma = -P.I + T$$

(I is the identity matrix)

note that we multiply by the I because P is value field .

substituting the body force by its corresponding expression we get the general form of Navier-Stokes Equation

$$\rho.\frac{D\vec{v}}{Dt} = -\nabla P + \nabla.T + \rho.\vec{f}$$

Newtonian Fluids

the stress is proportional to the rate of deformation ,i.e the change in velocity is direction of the stress

$$\nabla.T = \mu.\nabla^2 v + \rho.\vec{f}$$

In-compressible Newtonian fluid we get :

$$\rho.\frac{D\vec{v}}{Dt} = -\nabla P + \mu.\nabla^2 v + \rho.\vec{f}$$

where μ is the viscosity, the conservation of density and

$$\nabla.\vec{v} = 0$$

For more information on fluid mechanics, i strongly suggest Fundamentals of Fluid Mechanics 7 edition by Bruce R. Munson [15].

Appendix C

Gaussian Quadratures

Introduction :

Gaussian Qudrature is a numerical integration technique to approximate definite integrals of the form $\int_{-1}^{1} f(x)dx$, while there exists other simpler methods that shares the same aim as Gaussian Quadratures like the Trapezoidal Rule, Gaussian Quadrature offers a substantial improvements in accuracy. While Trapezoidal Rule is more intuitive, Gaussian Quadrature offers superior performance for many types of integrals.
In the next sections we will discuss how the Trapezoidal Rule works to develop intuition to better understand the Gaussian Qudratures, after that we will discuss the Gaussian quadratures for 1 dimensional definite integral, after that we will discuss how the method is generalized to tackel more complex intergrals, those of multiple dimensions, we will take 4D as use case, in the end we will discuss how it fits in th whole picture of our work.

C.1 Trapezoidal Rule

As discussed in the introduction Gaussian Quadratures are more accurate than Trapezoidal Rule, nevertheless Trapezoidal Rule is good step to understand the the fundamentals of G.Q. // the idea is to approximate $\int_a^b f(x)dx$ by a weigted sum of f evaluated in the ends of the integral , mathematically:

$$\int_a^b f(x)dx \approx w_a f(a) + w_b f(b)$$

the weight w_a, w_b are chosen to be exact for polynomials,so

$$w_a = w_b = \frac{b-a}{2}$$

$$\int_b^a f(x)\mathrm{d}x = \frac{b-a}{2}f(a) + \frac{b-a}{2}f(b)$$

C.2 Gaussian Quadrature 1D

Gaussian quadrature method approximate an integral in the interval $[-1, 1]$

$$\int_{-1}^{1} f(x)dx \approx \Sigma_{i=1}^{s} w_i f(x_i)$$

nodes $x_i's$ are chosen to be the s first roots of the Legendre polynomials,and the weights $w_i's$ are calculated to be exact for polynomials of $2n + 1$ degree,

$$w_i = \frac{2}{(1-x_i^2)[P_n'(x_i)]^2}$$

C.3 Gaussian Quadratues 4D

given a function f defined $f : \mathbb{R}^4 \to \mathbb{R}$ the integral is approximated as follow :

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f(t,x,y,z)\mathrm{d}t\mathrm{d}x\mathrm{d}y\mathrm{d}z \approx \Sigma_{i=0}^{s}\Sigma_{j=0}^{n}\Sigma_{k=0}^{m}\Sigma_{l=0}^{p} w_i w_j w_k w_l f(t_i, x_j, y_k, z_l)$$

where t_i, x_j, y_k, z_l are the nodes for the quadrature in each dimension,and w_i, w_j, w_k, w_l are the corresponding weights.
the steps for approximation such an integral is as follow :

1. choose the number of nodes in each dimension (it could be the same for every dimension)
2. compute the nodes for each dimension
3. form a grid for all combinations of weights and nodes

4. calculate the for all combinations the product of the weights and f evaluated at that combinations of nodes

5. sum for all the weighted evaluations

C.4 PINN-MRI Alignment via Gaussian Quadrature

MRI data is intrinsically averaged , but the neural network u_{nn} used does point wise prediction, meaning that given (t, x, y, z) the network predict the velocity vector at that specific point, so we Need to reconcile point-wise predictions with volume-averaged measurements i.e MRI data, this were Gaussian quadrature comes to play. The idea is to define the size of a voxel i.e 3D spatial volume over time, we average the velocities inside that voxel, mathematically speaking:

$$\bar{u}_{nn} = \frac{1}{\Delta t\, \Delta x\, \Delta y\, \Delta z} \int_{\Delta t} \int_{\Delta x} \int_{\Delta y} \int_{\Delta z} u_{nn}\, \mathrm{d}t\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z$$

We use the Gaussian quadrature to approximate this integral and we got the following:

$$\bar{u}_{nn} \approx \frac{1}{\Delta t \Delta x \Delta y \Delta z} \Sigma_{i=0}^{s} \Sigma_{j=0}^{n} \Sigma_{k=0}^{m} \Sigma_{l=0}^{p} w_i w_j w_k w_l u_{nn}(t_i, x_j, y_k, z_l)$$

Appendix D

Diving into Problems of PINN

PINNs, despite their potential, face several critical challenges. Spectral bias causes PINNs to struggle with learning high-frequency components of solutions, leading to poor approximations of complex functions. The issue of imbalanced losses arises when different terms in the loss function (e.g., data fitting, PDE residuals) have disparate magnitudes, resulting in suboptimal training and solutions that may satisfy some constraints at the expense of others. Lastly, causality violation [21] occurs when PINNs fail to respect the causal structure inherent in time-dependent problems, potentially leading to physically inconsistent predictions. These flaws can significantly impact the accuracy and reliability of PINN solutions, particularly in complex, multi-scale, or time-dependent physical systems, which the case in fluid mechanics. In summary, PINN suffers from the following problems:

1. Spectral Bias
2. Imbalanced Losses
3. Causality Violation

To address the inherent challenges in PINNs, several advanced techniques have been developed. Non-dimensionalization helps normalize the scales of different variables and equations, mitigating issues related to imbalanced losses and improving overall training stability. Fourier feature embeddings combat spectral bias by explicitly incorporating high-frequency components into the network's input, enabling better approximation of complex functions. Various loss weighting schemes, such as adaptive or annealing-based approaches, dynamically adjust the relative importance of different loss terms during training, addressing the problem

of imbalanced losses and ensuring a more balanced satisfaction of all constraints. Lastly, causal training strategies, which include techniques like time-marching schemes or causal encoders, enforce the temporal causality in time-dependent problems, preventing causality violations and ensuring physically consistent predictions. These enhancements collectively improve the accuracy, reliability, and physical consistency of PINN solutions across a wide range of applications. In summary, the following techniques mitigate the inherent flaws mentioned earlier:

1. Non-dimensionalization
2. Fourier feature embeddings
3. Loss Weigheting Schemes
4. Causal Training Strategies

Also the following techniques can be used to enhance the performance of PINNs:

1. Modified multilayer perceptron(MLP) architecture
2. Random Weight Factorization
3. Curriculum Training

D.1 The Right Pipeline to Solve PINN problems

Pipeline proposed consisted of three-step approach optimizes Physics-Informed Neural Networks. It begins with non-dimensionalization, transforming varied physical quantities into comparable dimensionless parameters. Next, it employs advanced network architecture techniques to better capture complex physical behaviors. The final step introduces innovative training optimization strategies, balancing loss components and respecting physical causality. This comprehensive pipeline significantly improves PINN accuracy and consistency in modeling diverse scientific and engineering applications.

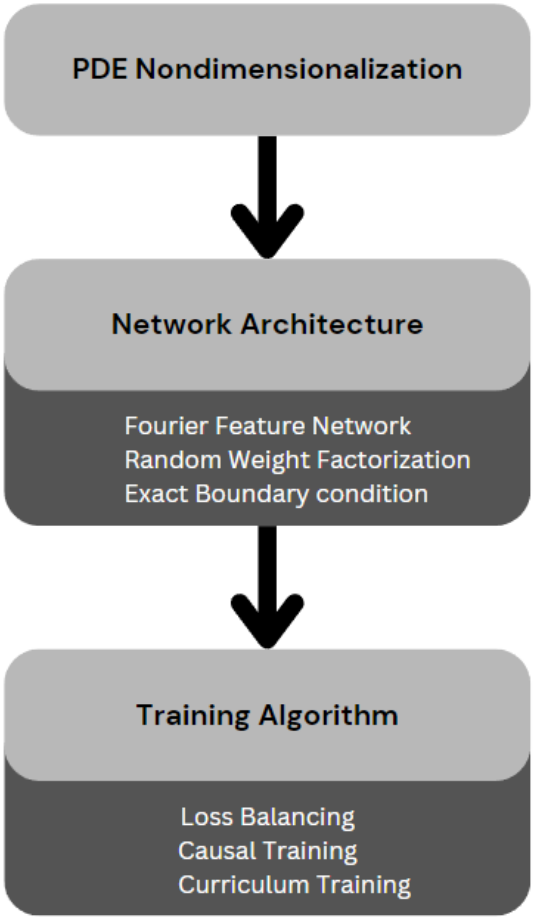

Figure D.1: Proposed PINN Enhancement Pipeline

D.1.1 Non-dimensionlization

Non-dimensionalization in PINNs transforms complex systems into dimensionless forms, offering two key benefits. First, it mitigates differences in variable scales, allowing the neural network to handle diverse physical quantities more effectively. Second, it promotes faster convergence and improved performance during training. By normalizing variables to similar ranges, the optimization process becomes more stable and efficient, enabling PINNs to learn underlying physical relationships more accurately and rapidly across various scientific and

engineering applications.

D.1.2 Architecture

Random Fourier Features

Random Fourier Features is proposed [20] to solve the problem of spectral bias, which is the tendency of MLP to prefer learning low frequency functions. To mitigate the spectral bias, Random Fourier Embedding is suggested:

$$\gamma : \mathbb{R}^n \rightarrow \mathbb{R}^m \tag{D.1}$$

$$\gamma(x) = [B_1 \cos(x_1), B_2 \sin(x_1), ...] \tag{D.2}$$

where B_i is sampled from a normal distribution $B_i \sim \mathcal{N}(0, \sigma^2)$ where σ is an hyper-parameter

Random Weight Factorization

Random Weight Factorization is an effective technique that improves the performance of the PINN, the idea is to add a trainable parameter which is a scale for every neuron in the network. RWF implementation is straight forward:
(a) Initialize the weights of the network $\{W_i, b_i\}_{i=1}^{L}$
(b) For each layer l sample each of the entries of s^l from a normal distribution $s^l \sim \mathcal{N}(0, \sigma^2)$
(c) Thus $W^{(l)} = diag(\exp\left(s^{(l)}\right)).V^{(l)}$
(d) The new network parameters become $\{s^i, V^i, b^i\}_{i=1}^{L}$
The authors recommend $\mu = 1.0$ and $\sigma = 0.1$

D.1.3 Training

Temporal Causality

PINNs tends to minimize the residual without taking into account the chronological order, meaning trying to minimizing the residuals at later times before getting correct solutions for earlier times. The idea is to split the time range into M equal parts, so the residual loss will be defined as follow:

$$\mathcal{L}_r(\theta) = \frac{1}{M}\Sigma_{i=1}^{M} w_i \mathcal{L}_r^i(\theta) \tag{D.3}$$

Such that

$$w_i = exp(-\epsilon \Sigma_{j=1}^{i} \mathcal{L}_r^i(\theta)) \tag{D.4}$$

Where ϵ is a hyper-parameter

Loss Balancing

Loss Balancing is a technique implemented to prevent the gradient of some loss term to steer the learning in its direction too much, which may affects the learning badly. The weights are first initialized to one, after some specified number of epochs the the weights are updated as follow:
(a) Initialize the weights $\lambda_{ic} = \lambda_{bc} = \lambda_r = 1$
(b) For every f epochs :

$$\hat{\lambda}_{ic} = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_{ic}||}$$
$$\hat{\lambda}_{bc} = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_{bc}||}$$
$$\hat{\lambda}_{r} = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_r||}$$

Updates the weights :

$$\lambda_{new} = \alpha \lambda_{old} + (1 - \alpha)\hat{\lambda}_{new}$$

Curriculum Training

The idea of curriculum training[12] is to decompose the optimization task into more manageable sub-tasks, this approach is suggested to be an effective way to solve the problem that requires high predictive accuracy. This step is better explained by an example, if our problem is to find a solution for navier-stokes equation with a high Renolds number, we start by training the model for low Renolds number, and we use that as a starting point for solving with respect to high Renolds number

D.1.4 Various Enhancing Methods

Optimizer and Learning rate

The Adam optimizer is the preferred choice for training Physics-Informed Neural Networks (PINNs) in forward problems, offering superior performance with minimal tuning requirements. Notably, weight decay, also known as L2 regularization, is not recommended for forward problems, as it may unnecessarily constrain the network's ability to capture intricate physical relationships. Research [22] indicates that an initial learning rate of 10^-3, coupled with an exponential decay schedule, consistently yields favorable results. This learning rate strategy allows the optimizer to make significant progress in the early stages of training while gradually refining the solution as training progresses. The combination of Adam optimizer and this specific learning rate approach enables PINNs to converge efficiently and accurately to solutions that satisfy both the governing equations and boundary conditions of the physical system being modeled.

Random Sampling

Random sampling of points for enforcing physics in PINNs is highly recommended due to its significant benefits in improving both performance and generalization capabilities. This approach involves selecting a diverse set of points throughout the domain where the governing equations are evaluated, rather than using a fixed grid. By randomly sampling these collocation points, PINNs are exposed to a more comprehensive representation of the problem space, which acts as an implicit form of regularization. This randomization helps prevent overfitting to specific regions and encourages the network to learn a more robust and generalized representation of the underlying physics. The stochastic nature of point selection also aids in breaking symmetries that might otherwise lead to suboptimal local minima during training. Furthermore, random sampling allows for adaptive refinement strategies, where more points can be dynamically allocated to regions of high complexity or error. This flexibility enhances the PINN's ability to capture fine-scale features and steep gradients in the solution, ultimately leading to more accurate and physically consistent predictions across the entire domain.

Modified MLP

A modified version of MLP is suggested [23] to enhance the capability of the PINNs for learn non-linear and complex solutions of PDE. The modified MLP is

presented in the following:

$$U(x) = \sigma(W_1 * x + b_1); V(x) = \sigma(W_2 * x = b_2)$$

Such that σ is a non linear activation

$$f^{(l)}(x) = W^{(l)}.g^{(l-l)}(x) + b^{(l)}$$

$$g^{(l)}(x) = \sigma(f^{(l)}(x)) \odot U(x) + (1 - \sigma(f^{(l)}(x))) \odot V(x)$$

Thus the parameters of the network are the following:

$$\theta = \{W_1, b_1, W_2, b_2, (W_l, b_l)_{l=1}^{L}\}$$

Choosing a small ϵ may fail to impose temporal causality, in contrast choosing a large ϵ may results in a more difficult optimization problem.

D.1.5 The Optimal Algorithm

1. Non-dimensionlize PDE
2. Represent the solution of the PDE by a modified MLP u_θ by Fourier Feature Embeddings and Random weights Factorization, use the tanh as an activation function and use the Glorot scheme
3. Formulate the Loss function :

$$\mathcal{L}(\theta) = \lambda_{ic}\mathcal{L}_{ic}(\theta) + \lambda_{bc}\mathcal{L}_{bc}(\theta) + \lambda_r\mathcal{L}_r(\theta) \tag{D.5}$$

   Such that we split the range of time into M equal parts, so:

$$\mathcal{L}_r(\theta) = \frac{1}{M}\Sigma_{i=1}^{M} w_i \mathcal{L}_r^i(\theta) \tag{D.6}$$

4. Set all the weights λ_{ic},λ_{bc}, λ_r and $\{w_i\}_{i=1}^{M}$

5. for $n = 1, ..., S$ do (a) Randomly sample points to evaluate the terms of the loss function (b) Update the Temporal weights w_i if $nmodf$ then (c) Compute the loss terms weights by:

$$\hat{\lambda}_{ic} = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_{ic}||} \tag{D.7}$$

$$\hat{\lambda}_{bc} = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_{bc}||}$$

$$\hat{\lambda}_r = \frac{||\Delta_\theta \mathcal{L}_{ic}|| + ||\Delta_\theta \mathcal{L}_{bc}|| + ||\Delta_\theta \mathcal{L}_r||}{||\Delta_\theta \mathcal{L}_r||}$$

(d) Update the weights of the loss terms

$$\lambda_{new} = \alpha \lambda_{old} + (1 - \alpha)\hat{\lambda}_{new} \tag{D.8}$$

end if

6. Update the parameters θ

$$\theta_{n+1} = \theta_n - \eta \Delta_\theta \mathcal{L}(\theta_n) \tag{D.9}$$

Bibliography

[1] Amirhossein Arzani, Jian-Xun Wang, and Roshan M D'Souza. Uncovering near-wall blood flow from sparse data with physics-informed neural networks. *Physics of Fluids*, 33(7), 2021.

[2] Jens Berg and Kaj Nyström. A unified deep artificial neural network approach to partial differential equations in complex geometries. *Neurocomputing*, 317:28–41, 2018.

[3] Ching-En Chiu, Aditi Roy, Sarah Cechnicka, Ashvin Gupta, Arieh Levy Pinto, Christoforos Galazis, Kim Christensen, Danilo Mandic, and Marta Varela. Physics-informed neural networks can accurately model cardiac electrophysiology in 3d geometries and fibrillatory conditions. *arXiv preprint arXiv:2409.12712*, 2024.

[4] Sourav Das and Solomon Tesfamariam. State-of-the-art review of design of experiments for physics-informed deep learning. *arXiv preprint arXiv:2202.06416*, 2022.

[5] Leon Ericsson, Adam Hjalmarsson, Muhammad Usman Akbar, Edward Ferdian, Mia Bonini, Brandon Hardy, Jonas Schollenberger, Maria Aristova, Patrick Winter, Nicholas Burris, et al. Generalized super-resolution 4d flow mri-using ensemble learning to extend across the cardiovascular system. *IEEE journal of biomedical and health informatics*, 2024.

[6] Mojtaba F Fathi, Isaac Perez-Raya, Ahmadreza Baghaie, Philipp Berg, Gabor Janiga, Amirhossein Arzani, and Roshan M D'Souza. Super-resolution and denoising of 4d-flow mri using physics-informed deep neural nets. *Computer Methods and Programs in Biomedicine*, 197:105729, 2020.

[7] Edward Ferdian, Avan Suinesiaputra, David J Dubowitz, Debbie Zhao, Alan Wang, Brett Cowan, and Alistair A Young. 4dflownet: super-resolution 4d

flow mri using deep learning and computational fluid dynamics. *Frontiers in Physics*, 8:138, 2020.

[8] Nathan Fulton and André Platzer. Safe reinforcement learning via formal methods: Toward safe control through proof and learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[9] Zhongkai Hao, Songming Liu, Yichi Zhang, Chengyang Ying, Yao Feng, Hang Su, and Jun Zhu. Physics-informed machine learning: A survey on problems, methods and applications. *arXiv preprint arXiv:2211.08064*, 2022.

[10] Xiuxiu He, Yang Lei, Yabo Fu, Hui Mao, Walter J Curran, Tian Liu, and Xiaofeng Yang. Super-resolution magnetic resonance imaging reconstruction using deep attention networks. In *Medical Imaging 2020: Image Processing*, volume 11313, pages 642–647. SPIE, 2020.

[11] Ameya D Jagtap, Kenji Kawaguchi, and George Em Karniadakis. Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *Journal of Computational Physics*, 404:109136, 2020.

[12] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021.

[13] Haitong Ma, Jianyu Chen, Shengbo Eben, Ziyu Lin, Yang Guan, Yangang Ren, and Sifa Zheng. Model-based constrained reinforcement learning using generalized control barrier function. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4552–4559. IEEE, 2021.

[14] Mohammadamin Mahmoudabadbozchelou, George Em Karniadakis, and Safa Jamali. nn-pinns: Non-newtonian physics-informed neural networks for complex fluid modeling. *Soft Matter*, 18(1):172–185, 2022.

[15] B.R. Munson, A.P. Rothmayer, and T.H. Okiishi. *Fundamentals of Fluid Mechanics, 7th Edition*. Blackwell handbooks in linguistics. Wiley, 2012.

[16] Maziar Raissi, Paris Perdikaris, and George 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*, 378:686–707, 2019.

[17] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

[18] Suprosanna Shit, Judith Zimmermann, Ivan Ezhov, Johannes C Paetzold, Augusto F Sanches, Carolin Pirkl, and Bjoern H Menze. Srflow: Deep learning based super-resolution of 4d-flow mri data. *Frontiers in Artificial Intelligence*, 5:928181, 2022.

[19] Fergus Shone, Nishant Ravikumar, Toni Lassila, Michael MacRaild, Yongxing Wang, Zeike A Taylor, Peter Jimack, Erica Dall'Armellina, and Alejandro F Frangi. Deep physics-informed super-resolution of cardiac 4d-flow mri. In *International Conference on Information Processing in Medical Imaging*, pages 511–522. Springer, 2023.

[20] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

[21] Sifan Wang, Shyam Sankaran, and Paris Perdikaris. Respecting causality is all you need for training physics-informed neural networks. *arXiv preprint arXiv:2203.07404*, 2022.

[22] Sifan Wang, Shyam Sankaran, Hanwen Wang, and Paris Perdikaris. An expert's guide to training physics-informed neural networks. *arXiv preprint arXiv:2308.08468*, 2023.

[23] Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

[24] Xinjie Wang, Siyuan Zhu, Yundong Guo, Peng Han, Yucheng Wang, Zhiqiang Wei, and Xiaogang Jin. Transflownet: A physics-constrained transformer framework for spatio-temporal super-resolution of flow simulations. *Journal of Computational Science*, 65:101906, 2022.

[25] C. Westbrook, C.K. Roth, and J. Talbot. *MRI in Practice*. Wiley, 2011.

[26] Zhiyuan Zhao, Xueying Ding, and B. Aditya Prakash. Pinnsformer: A transformer-based framework for physics-informed neural networks, 2024.