Title

Alejandro Salazar Arango[1]

Advisor(s):
Cristhian David Zambrano Mora[2]

Research practice 3
Research proposal
Mathematical Engineering
School of Applied Sciences and Engineering
Universidad EAFIT

August 2023

[1]Student information (email)
[2]Affiliation, email

1 Introduction

Partial differential equations (PDEs) are one of the most powerful tools for the study of physical and biological phenomena. They allow us to represent in a simplified way, the dynamics between the spatial and temporal variables of any phenomenon and to take this dynamics to a mathematical model, whose solution describes in a great way the behavior of the studied system[1]. This has led partial differential equations to become a tool of great use in areas such as fluid dynamics, electromagnetism, optics, magnetism, among others[2]; these areas being dominated by PDEs such as Maxwell's Equations and Navier-Stokes Equations, the latter being the object of study of the present work.

However, PDEs, given their complex structure, are usually models whose analytical solutions are difficult or impossible to find, being even the solutions already found for some PDEs so complex that it is preferred to use other methods to solve the problem[3]. This is why, in recent years, the development of multiple numerical methods to solve this type of equations has been one of the major topics of study of the scientific community, being this an important issue to study, and that although there are already multiple methods developed, if these are not applied correctly can lead to solutions far from the real solution of the problem and therefore lead to erroneous conclusions about the behavior of the system under study.

One of the most commontly used numerical method for solving Partial Differential Equations is the Finite Element Method (FEM), a numerical method consisting on the discretization of the problem's domain, to later solve a variational formulation of the problem with determined test functions, to obtain an algebraic system of equations, which, when solved, gives the approximated solution of the problem. However, due to the large amount of degrees of freedom it needs in order to correctly solve the PDE, it remains as an extremely expensive method both in terms of CPU and memory demand, therefore not being too useful for real-time contexts[4][5]. As a consequence, models such as reduced basis models (RBMs) and physically informed neural networks (PINNs) have been developed and have become important methods in the study of PDEs.

This project focuses on the implementation of artificial intelligence models for the solution of PDEs, specifically Physical Informed Neural Networks (PINNs) and Neural Network asisted Reduced Basis Models (NN-RBM), seeking to obtain a reduced computational cost in a real-time simulation context. The Navier-Stokes equations, and in particular, a simulation problem of blood flow through the carotid artery, will be used as an object of study, to observe and compare the performance of these methods versus the FEM in an on-line simulation process, aiming for a deeper understanding of the advantages and limitations of NN-RBMs, PINNs and FEM in the context of real-time simulations.

2 Statement of the problem

2.1 Statement of the problem

For solving fluid dynamics problems, Computational Fluid Dynamics (CFD) mainly uses three methods: The Finite Difference Method, The Finite Volumes Method, and the Finite Element Method. In the finite volume method, the discretization is performed by partitioning the spatial domain into a mesh, in which each of its elements will be called a control volume. The idea is to perform the integration of the dominant equations of the problem through each control volume, balancing the fluxes across the boundaries of the individual volumes and getting what is called a balance equation. Then the set of balance equations will be discretized with respect to a set of discrete unknowns to finally get the solution to the problem[6]. In a topologically regular mesh, these calculations are done quite easily, but when working with irregular meshes, this calculation will result in an unbearable amount of fluxes and a significant accounting effort to ensure that all fluxes have been calculated correctly[7].

In the finite difference method, the derivatives of the problem are usually replaced by truncated expansions of the Taylor series, this method is straightforward for regular geometries, but when working with irregular geometries it is necessary to perform transformations to the equations before the expansion is carried out[7]. Finally, the finite element method works by dividing the initial domain of the same (a continuous element) into a large number of non-intersecting subdomains, called finite elements. At last, the formulation of a boundary value problem ends up giving a system of algebraic equations, which are then solved using variational methods in order to approximate a solution, minimizing an associated error function[8]. Nevertheless, these approaches come with a significant drawback. Due to their extensive equation systems, they demand a substantial number of degrees of freedom to accurately approximate the model solution.

This becomes a problem because as the complexity of the problem increases, the number of degrees of freedom will also increase and this, in turn, means that for the model to be solved correctly, a greater use of computational resources such as memory, storage and processing capacity will be necessary. This implies that the problem takes more and more time to solve, reaching a point where even the available resources may be insufficient to solve the problem properly. As a consequence of the above, the commonly used models do not excel as efficient choices for a real-time context, and given the importance of a fast response rate in some of the applications of PDEs[9], it becomes important to develop new agile methods for the solution of PDEs.

In this regard, one of the main solutions proposed in the literature are the Reduced Base Models (RBM), which base their solution process on the initial obtaining, in a first offline phase, of a group of snapshots representative of the model's parameter space and from this they build a reduced base of the space, to later, in a second online phase, project the complete order system on this reduced base, and thus obtain an approximate solution of the problem as a linear combination of the elements of the base. However, this method does not represent much computational gain when the problems are very complex, since this leads to the projection process being equally complex and therefore does not allow the correct solution of the PDE in the expected time.

2.2 Formalization of the problem

In a domain $\Omega \subset \mathbb{R}^2$ Quateroni[10] define incompressible Navier-Stokes Equations as follows

$$\begin{cases} \nabla \cdot \mathbf{u} = 0 & \mathbf{x} \in \Omega, t > 0 \\ \mathbf{u}_t - \nabla \cdot [\nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{f} & \mathbf{x} \in \Omega, t > 0 \end{cases} \tag{1}$$

When ν is constant, we obtain

$$\nabla \cdot [\nu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] = \nu(\Delta \mathbf{u} + \nabla(\nabla \cdot \mathbf{u})) = \nu \Delta \mathbf{u}$$

and therefore the equations are written as

$$\begin{cases} \nabla \cdot \mathbf{u} = 0 & \mathbf{x} \in \Omega, t > 0 \\ \mathbf{u}_t - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{f} & \mathbf{x} \in \Omega, t > 0 \end{cases} \tag{2}$$

Being $\mathbf{u}$ the fluid velocity field, p the pressure divided by the density ρ, ν the kinematic viscosity define as $\frac{\mu}{\rho}$ with μ being the dynamic viscosity and $\mathbf{f}$ a forcing term per unit mass.

For the purpose of this work the domain Ω must also be bounded, and it is therefore necessary to assign initial and suitable boundary conditions for the problem to be well posed.

$$\begin{cases} \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) & \forall \mathbf{x} \in \Omega \\ \mathbf{u}(\mathbf{x}, t) = \varphi(\mathbf{x}, t) & \forall \mathbf{x} \in \mathit{\Gamma}_D, t > 0 \\ \left(\nu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} - p\mathbf{n}\right)(\mathbf{x}, t) = \psi(\mathbf{x}, t) & \forall \mathbf{x} \in \mathit{\Gamma}_N, t > 0 \end{cases}$$

Where $\mathbf{u}_0$ is a divergence free vector field, φ and ψ are given vector functions, $\mathit{\Gamma}_D$ and $\mathit{\Gamma}_N$ provide a partition of the boundary $\partial\Omega$ and $\mathbf{n}$, as usual, represents the outward unit normal vector to $\partial\Omega$.

3 Objectives

3.1 General objective

Compare the performance of Neural Network asisted Reduced Basis Models (NN-RBMs) and Physical Informed Neural Networks (PINNs) agains the Finite Element Method on a real-time context, by performing multiple simulations of the blood flow through the carotid arthery.

3.2 Specific objectives

- Identify the principal differences on the implementation of NN-RBMs and PINN like models and FEM in a fluid mechanic context.
- Assess the ability of NN-RBMs and PINNs to adapt to changing boundary conditions or parameters, compared to the FEM approach.
- Study the generalization capabilities of NN-RBMs and PINNs when trained on a specific set of data and tested on different scenarios, compared to FEM.
- Analyze the dependency of NN-RBMs to the Snapshots used for the training on the model

4 Justification

Finally, this project aims to contribute to a deeper understanding of the advantages and limitations of neural NN-RBMs, PINNs and FEM in the context of real-time simulations of the Navier-Stokes equations. Guiding researchers and practitioners in selecting the most appropriate method for specific applications based on accuracy and computational efficiency considerations.

5 Scope

Now then, it is clear that for this kind of project one of the biggest obstacles that it is possible to encounter when researching and implementing the model is the impossibility of carrying out analytical validation of the solution obtained by the method, since as it is known, the Navier-Stokes equations, due to their high complexity, do not have an analytical solution against which to compare the numerical solution. However, since multiple models are implemented, and having the FEM as a commontly accepted method for the solution of Parameterized PDEs, it's solution will be used as an accepted solution for validating the implemented models.

6 State of the art

7 Proposed methodology

As a means to achieve the objectives set for this project, a 4-stage methodology is proposed in which various mathematical methods will be worked. In the first stage, the interpretation of angiograms obtained on the web will be performed and then a two-dimensional geometry of the carotid artery will be recreated to be used as the spatial domain of the simulations to be performed. In the second stage, a computational implementation of the finite element method for obtaining the data and snapshots needed for the implementation and validation of the other models. The third stage involves the implementation of NN-RBMs and PINNs using the results obtained before as a basis for this models and the simulation fo multiple study cases. And finally, in the fourth stage, a comparison of the results obtained for each implmented model, as well as their simulation-time and memory usage is performed, to analyze the performance of each model. Let's dive now into the pricipal numerical methods used for this project.

The finite element method is a numerical method highly used in computational fluid dynamics to solve problems whose domain is defined by irregular geometries. The method makes it possible to find the numerical solution to these problems by dividing the initial domain of the same (a continuous element) into a large number of non-intersecting subdomains, called finite elements. Each element will have a set of representative points called nodes, which will make up the complete grid of the problem, on which the calculations will be performed. At last, the formulation of a boundary value problem ends up giving a system of algebraic equations, which are then solved using variational methods in order to approximate a solution, minimizing an associated error function[8]. Nevertheless, these method have a great disadvantage, since, being systems of such large equations, they require a high number of degrees of freedom to achieve the correct estimation of the model solution, and therefore, they are not very efficient methods for an online context, in which a higher response speed is needed. It is here where RBMs and PINNs become important, since they allow giving answers to PDE problems in a much more agile way.

RBMs involve constructing a low-dimensional approximation space, known as the *reduced basis*, by performing some dimensional reduction technique over a group of representative snapshots computed for a set of parameter values, ending up with an space containing the principal characteristics of the expected PDE's Solution[11]. By projecting the PDE onto the reduced basis, RBMs effectively transform the high-dimensional problem into a lower-dimensional one, facilitating real-time simulations, design optimization, and uncertainty quantification, offering a balance between accuracy and computational efficiency. RBMs are particularly advantageous for parametric PDEs, where the solution depends on a set of input parameters, as they allow for rapid evaluations of solutions across parameter variations.[12]. However, since for the online section it is needed to perform a Garlekin-like process to project the full order system into the reduced basis, for Complex no-linear problems RBM do not provide any computational Gain with respect to the original approach, therefore, some non-intrusive methods had been developed in which the projectiion is made by an interpolation, beong one of the most interesting approches, neural network-assisted RBMs (NN-RBMs) which incorporate neural networks in the projection process, by helping interpolating over the parameter space[4].

On the other hand, physics-informed neural networks (PINNs) have emerged as a promising approach to solve PDEs with remarkable efficiency and accuracy. These combine the power of neural networks with the governing equations of physical systems, offering a seamless integration of data-driven learning and domain-specific physics. They work by incorporating the residual error of the governing equations into the neural network training process, gaining a unique ability to capture the underlying physics of the model while learning from sparse and noisy data. [9] Unlike traditional methods, PINNs avoid the need for explicit mesh discretization and provide continuous solutions throughout the domain, dynamically adapting to complex geometries. Their versatility, combined with the ability to leverage existing deep learning frameworks, positions PINNs as a transformative tool for advancing EDP-based modeling and simulation in interdisciplinary domains.

8 Schedule, commitments and deliverables

Table 1 presents the proposed schedule of the project

Table 1: Schedule

Activity	*Weeks*																	
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*
Literature Review	■	■	■	■	■													
Research proposal writting					■	■												
FEM Implementation							■	■	■									
PINN and NN-RBM Implementation									■	■	■	■						
Results Analysis and comparison													■	■	■	■		
Final Report writting																■	■	■

9 Intellectual property

According to the internal regulation on intellectual property within Universidad EAFIT, the results of this research practice are product of *Alejandro Salazar Arango* and *Cristhian David Zambrano Mora*.

In case further products, beside academic articles, that could be generated from this work, the intellectual property distribution related to them will be directed under the current regulation of this matter determined by [13].

References

[1] Logan JD. Applied partial differential equations. Springer; 2014.

[2] Farlow SJ. Partial differential equations for scientists and engineers. Courier Corporation; 1993.

[3] Strauss WA. Partial differential equations: An introduction. John Wiley & Sons; 2007.

[4] Hesthaven JS, Ubbiali S. Non-intrusive reduced order modeling of nonlinear problems using neural networks. Journal of Computational Physics. 2018;363:55-78.

[5] Quarteroni A. Physics-Based and Data-Driven-Based Algorithms for the Simulation of the Heart Function; 2023. .

[6] Eymard R, Herbin R, Gallouët T. Finite volume methods. Handbook of Numerical Analysis. 2002;7:713-1020.

[7] AutoDesk. Finite Element vs Finite Volume; 2019. `https://knowledge.autodesk.com/support/cfd/learn-explore/caas/CloudHelp/cloudhelp/2014/ENU/SimCFD/files/GUID-12A9AED8-2047-4D3A-BC80-82BE9CF47517-htm.html`.

[8] Zienkiewicz OC, Taylor RL, Nithiarasu P. The finite element method for fluid dynamics. Butterworth-Heinemann; 2005.

[9] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational physics. 2019;378:686-707.

[10] Quarteroni A, Quarteroni S. Numerical models for differential problems. vol. 2. Springer; 2009.

[11] Benner P, Ohlberger M, Cohen A, Willcox K. Model reduction and approximation: theory and algorithms. SIAM; 2017.

[12] Quarteroni A, Manzoni A, Negri F. Reduced basis methods for partial differential equations: an introduction. vol. 92. Springer; 2015.

[13] Universidad EAFIT. Reglamento de propiedad intelectual; 2017.