Learning-Enhanced Finite Volume Methods for Nonlinear Convection-Diffusion Problems

Mustafa Abbas Majthoob[1]

1 Applied Mathematics, Islamic Azad University,Tahran Iran,14179-36531

1 *mustaahmed59@gmail.com*

Abstract

Nonlinear convection-diffusion-reaction equations are fundamental to CFD, with applications ranging from environmental modeling to reactive flows. While the Finite Volume Method (FVM) is prized for its conservation properties and low cost, it struggles with steep gradients and discontinuities. Conversely, Physics-Informed Neural Networks (PINNs) offer mesh-free flexibility but suffer from slow convergence. This study introduces Learning-Enhanced Finite Volume Methods (LE-FVM), a hybrid framework that integrates FVM with state-of-the-art deep learning paradigms: PINNs, Finite Volume Graph Networks (FVGN), and Kolmogorov-Arnold Networks (KANs). By embedding ML models directly into the FVM workflow—handling flux calculations, adaptive mesh refinement, or acting as end-to-end solvers—the framework ensures solutions are strictly constrained by integral conservation laws. Key innovations include a "twice-message aggregation" mechanism for unstructured grids and an adaptive loss-weighting scheme based on the Neural Tangent Kernel (NTK) to balance PDE residuals and boundary conditions. Benchmark evaluations on 1D and 2D systems (Burgers', Fisher', Burgers–Huxley) demonstrated that LE-FVM provides high-fidelity solutions over 95% of the time. For the 1D Burgers' equation (Re=1), LE-FVM reduced maximum absolute error by 173x compared to the Galerkin Finite Element Method (GFEM). In unsteady 2D flows, FVGN achieved 77% higher accuracy and 56% lower training time than purely data-driven models. Furthermore, the framework exhibited exceptional generalization to novel geometries (airfoils and ellipses) without retraining. By blending deterministic conservation laws with stochastic learning, LE-FVM offers an order-of-magnitude speedup for parametric studies, setting a new bar for solving complex nonlinear PDEs in science and engineering.

Keywords: *Nonlinear convection-diffusion, Hybrid numerical methods, Finite Volume Method, Physics-Informed Neural Networks, Deep Learning,*

طرق الحجم المحدود المعززة بالتعلم لمسائل الحمل والانتشار غير الخطية

مصطفى عباس مجذوب[1]

المستخلص

تُعد معادلات الحمل والانتشار والتفاعل غير الخطية نموذجاً أصيلاً في ميكانيكا الموائع الحسابية (CFD) والرياضيات التطبيقية. ورغم شهرة طريقة الحجوم المحدودة (FVM) بخصائص الحفظ والاستقرار، إلا أنها تواجه صعوبة في تمثيل التدرجات الحادة والانقطاعات بدقة. وفي المقابل، توفر الشبكات العصبية الموجهة بالفيزياء (PINNs) بديلاً مرناً، لكنها تعاني من بطء التقارب وضعف التعميم. يقدم هذا البحث مفهوم طريقة الحجوم المحدودة المعززة بالتعلم(LE-FVM) ، وهو إطار هجين يدمج (FVM) مع نماذج التعلم العميق المتطورة مثل(PINNs) ، وشبكات الحجوم المحدودة الرسومية (FVGN)، وشبكات كولموغوروف-أرنولد .(KANs) يعمل الإطار على دمج نماذج التعلم الآلي ضمن سير عمل (FVM) لحساب التدفقات، أو تحسين الشبكات التكيفية، أو كحلول طرفية مقيدة بقوانين الحفظ التكاملية. ويتضمن البحث ابتكارات جوهرية مثل آلية "تجميع الرسائل المزدوج" للشبكات غير المنتظمة، ووزن الخسارة التكيفي بناءً على "نواة المماس العصبي (NTK) "لموازنة متبقيات المعادلات التفاضلية والشروط الحدية. أظهر التقييم الشامل على نماذج (Burgers, Fisher, Burgers–Huxley) أن إطار (LE-FVM) قدم حلولاً عالية الدقة في أكثر من 95% من الحالات. ففي معادلة (Burgers) أحادية البعد، خفض الإطار الخطأ المطلق بمقدار 173 ضعفاً مقارنة بطريقة .(GFEM) وفي التدفقات ثنائية البعد غير المستقرة، حققت شبكة (FVGN) دقة أعلى بنسبة 77% وزمن تدريب أقل بنسبة 56% مقارنة بالشبكات المعتمدة على البيانات فقط. كما أظهر الإطار قدرة فائقة على التعميم للتنبؤ بالتدفق حول هندسات جديدة (مثل الأجنحة الانسيابية) دون إعادة تدريب. من خلال دمج قوانين الحفظ الحتمية مع التعلم الاحتمالي، توفر (LE-FVM) سرعة أكبر برتبة مقدار كاملة، مما يرفع سقف التوقعات في حل المعادلات التفاضلية الجزئية المعقدة.

الكلمات المفتاحية : الحمل والانتشار غير الخطي؛ الطرق العددية الهجينة؛ طريقة الحجم المحدود؛ الشبكات العصبية المستنيرة بالفيزياء؛ التعلم العميق.

1. Introduction

Convection-diffusion phenomena are central to a wide variety of science and engineering applications, including fluid dynamics, heat transfer, environmental modelling for pollutants dispersion, and mass transport in porous media [1], [2]. Nonlinear partial differential equations (PDEs) are especially able to mathematically characterise the stochasticity of these processes via the convective transport of a scalar quantity (e.g., concentration, temperature, or momentum) due to bulk fluid motion (convection) and its dispersal due to gradients (diffusion) [3]. In general, the governing equation of the scalar field can be written as Eq.(1):

$$\frac{\partial u}{\partial t} + \mathrm{A} \cdot (\mathrm{v}(u)u) = \mathrm{A} \cdot (D(u)\mathrm{A}u) + S(u, \mathrm{x}, t) \qquad (1)\ ,$$

where v is a (possibly nonlinear) velocity field, D is a (possibly nonlinear) diffusion coefficient, and Q represents source or sink terms [4]. The inherent non-linearity, mostly caused by the variable coefficients or reaction term, and the likelihood of obtaining sharp gradients, discontinuities or even multi-scale behavior makes these equations analytically intractable for all practical situations [5]. This has led to few good numerical methods for approximate solution, which has been a mainstay to the scientific and engineering communities for generations. Finite Volume Method (FVM) has especially brought toward itself a cornerstone in the area of Computational Fluid Dynamics (CFD) [6]. This feature endows it very well as by construction it conserves mass, momentum and energy over discrete control volumes (CV) and is favorable to shock/discontinuity problems [7]. Standard FVM is powerful but not perfect, so to be clear here. In practice, this leads to numerical diffusion (smearing of strong gradients) or dispersion (non-physical oscillations) [8] as well: Claiming stability, the elaborate reconstruction schemes to capture strong nonlinearities (or large Péclet number fronts) use some higher-order combinations. In particular, these problems have very high computational costs, especially for multi-dimensional or parametric problems [9], resulting in having to resolve very fine mesh and very small time steps. First it got a great start with the huge success of natural language processing and image recognition improvement on several disciplines and some recent evolution, while machine learning (ML), and first of all deep learning (DL), has emerged as one of the hottest new engine of technological progress in scientific computing with the hope of new ways to enhance, or in some cases erasing numerical methods[10]. By Raissi et al. This essential demand for data manifests itself as a new conceptual framework for computational modelling that incorporates this paradigm embodied as Physics-Informed Neural Networks (PINNs) [11]- where goads imposed in the shape of a loss function compel a neural network model to learn solutions that satisfy their governing PDEs, over sparse or noisy data. This approachavoids discretizing a large domain, and can easily adjust to more complex shapes [12].Recent advances such as operator learning employing DeepONets[13] and the new Kolmogorov-Arnold Networks(KANs)[11] have demonstrated even greater precision and efficiency on low-dimensional parametric PDEs. The joint use of these two paradigms constitute a paradigm shift among a recognized class of methods: the so-called Learning-Enhanced Finite Volume Methods (LE-FVM). LE-FVM is not meant as an alternative to FVM but as a complement to it with focus on ML/DL approaches that can help reduce the classical bottlenecks in FVM. Examples of the use of ML models are to learn the best flux limiters, solve adaptive meshes which determine over time according to the solution features, speedup the solve linear systems, or even learn sub-grid scale model that simulates the physics which cannot be resolved in coarse grid [12], [13]. However the aim is beyond just improvement, it is a fundamental step change in the computational power of physics engines - both with respect to the delivery of the anticipated speed, but also with regards to the accuracy at which solutions can then be solved in a reasonable time frame, this should allow to simulate problems previously considered far too complex in quasi real time or with orders-of-magnitude improved precision. We propose a systematic investigation into LE-FVM for nonlinear convection-diffusion problems. It begins from a fundamental understanding of classical FVM, describing its discretization schemes, stability analysis, and corresponding constraints in Section 2. We discover the ML/DL approach used for enhancement (e.g., PINNs, operator learning, and hybrid architectures) in Section 3. The fourth section forms the crux of the paper, where we introduce our unified framework for integrating both learning methodologies with FVM, as well as multiple pathways of integration between them e.g., learning-based flux correction and ML-led adaptive mesh refinement. Section 5 includes a comparison with the recent literature [8],[9] in terms of accuracy and efficiency, and demonstrates the benefit of the LE-FVM over classical methods. In Section 6 we summarize the results and challenges related to data needs, software integration and explainability of the models and in Section 7 we give some perspectives for future work, like the extension to multi-physics problems and meta-learning strategies for fast parametric studies.

The use of learning driven approaches in FVM is not just an optimization in engineering sense but a mathematical breakthrough in the realm of numerical methods. As previously mentioned, the universal approximation property of neural networks when learning mappings that must satisfy complicated physical effects enable LE-FVM to offer simulations that are not only more and accuracy intensive and robust but also more a exible and able to overcome the computation limits related with modeling real-world systems.

2. Theoretical Background on Traditional Finite Volume Methods

2.1. Governing Equations for Nonlinear Convection-Diffusion

The mathematical description of nonlinear convection-diffusion phenomena is encapsulated in partial differential equations (PDEs) that govern the transport of a scalar quantity $u(\mathrm{x}, t)$, such as concentration, temperature, or momentum, within a spatial domain $\mathcal{D} \subset \mathbb{R}^d$ over a time interval $[0, T]$. The general form of the nonlinear, time- dependent convection-diffusion-reaction equation is given by Eq.(2):

$$\frac{\partial u}{\partial t} + \mathrm{A} \cdot ((u)) = \mathrm{A} \cdot (\mathbf{F}_d(u, \mathrm{A}u)) + S(u, \mathrm{x}, t), \qquad (2)$$

where:

- $\mathrm{F}(u) = \mathrm{v}(u)u$ the convective flux, which stands for the transport due to the bulk motion of the fluid. Now, the function representing the velocity field $\mathrm{v}(u)$, can be in fact a function of the solution u and this gives an introduction to nonlinearity.

- $\mathrm{F}(u, \mathrm{A}u) = D(u)\mathrm{A}u$ the diffusive flux: the spreading of u caused by gradients Further, the diffusion coefficient $D(u)$ is allowed to be nonlinear in u.

- (u, x, t) is a source/sink-term, kinetic behaviour or external forcing, which can be a nonlinear function of u.

As a result of the intrinsic nonlinearity in $\mathrm{v}(u)$, $D(u)$ or $S(u)$, analytical solutions are unmanageable in a wide range of relevant cases, and thus robust numerical approximation techniques are required [1], [3]. The relationship between convection versus diffusion is quantifed by the Péclet number (Pe), which is the ratio of rates of convective to diffusive transport. Flows dominated by convection tend to have high Pe numbers, and these flows can give rise to sharp gradients and discontinuities that are difficult for traditional numerical methods to capture accurately without introducing spurious oscillations or excessive numerical diffusion [8].

To close the problem, Equation (1) must be supplemented with appropriate initial and boundary conditions:

- **Initial Condition (IC):** $u(\mathrm{x}, 0) = u_0(\mathrm{x})$ for all $\mathrm{x} \in \mathcal{D}$.

- **Boundary Conditions (BCs):** These can be of Dirichlet type (prescribed u on $\partial\mathcal{D}$), Neumann type (prescribed normal flux $\mathrm{A}u \cdot \mathrm{n}$ on $\partial\mathcal{D}$), or Robin type (a combination of both).

This class of equations is particularly well suited to the Finite Volume Method (FVM) There is two general approaches for discretizing the governing equations: conservation and non-conservation Because FVM is based on conservation principle, then the governing laws at the integral form [6], [7].

2.2. Discretization Techniques

The core principle of the Finite Volume Method is to partition the computational domain $\mathcal{D}$ into a finite set of non-overlapping control volumes (CVs) or cells, denoted as V_i, where

$i = 1,2, \ldots, N$. The union of these CVs approximates the domain, $\mathcal{D} \approx \cup$. The solution u

is typically represented by its cell-averaged value, $U_i(t)$, defined as Eq(3):

1

$(t) = |V| \int u\,(\mathrm{x}, t)d\mathrm{x},$ (3)

i Vi

where $|V_i|$ is the volume (or area in 2D) of cell V_i.

The governing PDE (Equation 1) is then integrated over each control volume V_i and over a finite time step $\Delta t = t^{n+1} - t^n$. Applying the divergence theorem to the flux terms transforms the volume integrals of the divergence into surface integrals over the cell boundaries Eq(4):

$$|V_i| \frac{dU_i}{dt} + \phi \; (u) \cdot \mathrm{n} dA = \phi \; \mathbf{F}_d\,(u, \mathrm{A}u) \cdot \mathrm{n} dA + \int S\,(u, \mathrm{x}, t)d\mathrm{x} \quad (4).$$

∂Vi ∂Vi Vi

This integral form is the foundation of FVM and ensures that the numerical scheme inherently respects the conservation law for the quantity u over each discrete control volume [6], [7].

The discretization process involves approximating the surface and volume integrals in Equation (3):

1. Flux Approximation: The key challenge lies in evaluating the convective and diffusive fluxes, F_c and F_d, at the cell faces (or edges in 2D). Since the solution is known only as a cell average, interpolation or reconstruction techniques are used to estimate the solution values and its gradients at the cell interfaces. For convection, upwind schemes are commonly employed for stability, while central differencing is often used for diffusion. High-resolution schemes (e.g., MUSCL, QUICK) are used to achieve higher-order accuracy while controlling oscillations [6].

2. Source Term Integration: The source term integral is typically approximated using a quadrature rule, often a simple evaluation at the cell centroid.

3. Temporal Discretization: The time derivative dU_i/dt is discretized using a finite difference scheme. Common choices include:

    - Explicit Schemes (e.g., Forward Euler): $U^{n+1} = U^n + \Delta t \cdot R(U^n)$. These are

i i i

simple to implement but are subject to a stability condition (e.g., CFL

condition) that can severely restrict the allowable time step Δt, especially for stiff problems or fine meshes [9].

Implicit Schemes (e.g., Backward Euler, Crank-Nicolson): $U^{n+1} = U^n + \Delta t \cdot$

i i

$R(U^{n+1})$. These schemes are unconditionally stable for linear problems, allowing for larger time steps, but require solving a system of (often nonlinear) algebraic equations at each time step, which can be computationally expensive [9].

The choice of spatial and temporal discretization schemes directly impacts the accuracy, stability, and computational efficiency of the FVM solver.

2.3. Stability and Convergence Analysis

For a numerical method to be reliable, it must be both stable and convergent. Stability ensures that errors introduced at any stage of the computation (e.g., round-off errors, initial condition errors) do not grow uncontrollably as the simulation progresses. Convergence guarantees that as the discretization is refined (i.e., as the mesh size Δx and time step Δt approach zero), the numerical solution U^n approaches the exact solution

i

(x_i, t^n) of the continuous PDE.

- **Stability Analysis:** In FVM, stability analysis usually focuses on the discrete maximum principle or linearized von Neumann stability analysis on orthogonal grids [9]. As an example, if we use upwind differencing for convection, it provides stability by ensuring that information marches and propagates in the right order. Nevertheless, in extremely nonlinear problems and for sophisticated unstructured grids, stability is hard to theoretically express and is usually checked empirically [8]. In high-resolution schemes, monotonicity preserving techniques such as slope limiters are used to maintain boundedness of systems by preventing new extrema from being created [6].

- **Convergence Analysis:** Convergence is usually proved by showing the numerical scheme to be consistent (the truncation error, which is the residual after substituting the exact solution into the discrete equations, vanishes as $\rightarrow 0$) and stable. Lyapunov type global stability properties for finite difference schemes play a significant role in numerical analysis of PDEs (partial differential equations), since the celebrated Lax Equivalence Theorem postulates that stability is necessary and sufficient for convergence [2], holding under the assumption of a consistent finite difference scheme applied to a well-posed linear initial value problem. This theorem establishes a basis for our approach, but although the rigorous convergence proofs can be extended for nonlinear PDEs solved with FVM on unstructured grids, it has remained as quite a theoretical challenge. In reality, convergence is usually proven through numerical experiments, e.g., to show that the solution error decays at the correct rate when the mesh is refined.

3. Introduction to Learning-Enhanced Techniques

3.1. Machine Learning & Deep Learning Approaches–A Summary

ML/DL provides new perspectives to infer solutions of PDEs[1], [4]. Conventional numerical methods can produce competitive accurate results, but may be hindered by factors such as steep gradients, discontinuities, or high dimensional parameter spaces that worsen the trade-off among accuracy, stability, and computational cost [1], [2]. Machine learning brings a parallel, and sometimes disruptive, paradigm to translate the extremely nonlinear mappings wiring experimental observations—directly from data or constrained by physical laws [7]—through the well-known universal approximation capabilities of neural networks [6].

Actions Create Opportunities Notability. 4BACHC Learning-driven approaches do not aim to simply replace classical solvers, but to complement them into hybrid methods that can take advantage of both paradigms. Among those perfect interactions is Physics-Informed Neural Networks (PINNs) [1]. The PINNs are trained with a composite loss function which considers the difference from available data (if any) as well as the residual of the governing PDE and its boundary/initial conditions[1], [5]. This is achieved by incorporating the physical laws into the training and optimization process thereby ensuring that the resultant learned solution is not just a data interpolant but physically consistent, even in the case of sparse or zero data regimes [5]. This is in stark contrast with data-driven models, which can make physically non-consistent predictions outside of the training distribution [9].

In addition to PINNs, there are also other state-of-the-art architectures which are previously developed. DeepONets (Deep Operator Networks) which learn mappings between infinite dimensional function spaces to generate a parametric PDE solution for any number of initial or boundary conditions in exchange for a single training step [7], [4]. This is especially useful in problems where many simulations under various conditions are required, e.g. uncertainty quantification or optimization. Recently Kolmogorov-Arnold Networks (KANs) [6] were proposed as an effective substitute for classical multilayer perceptron(MLP) regarding this task. Then, using a version of the Kolmogorov-Arnold representation theorem, they define a kind of KAN (replace fixed activation functions on nodes with learnable univariate functions on edges) While KANs do not provide strong performance on standard supervised problems yet, they have (in preliminary studies) equal or better performance than MLPs with fewer parameters and stronger robustness to noise, making it a compelling candidate for physics-informed learning[12], [9].

Note that these techniques are not limited to approximate end-to-end solutions. Machine Learning (ML) models today are soon to be intelligent replacements for specific, costly to compute parts of the standard finite volume workflow. To that end, they can either be trained to predict turbulent closure models [13], flux limiters [12], or even the optimal time-step sizes, to both accelerate the ML procedures (13) or the global simulation [14], while respecting the conservation equations complied by the core of the VFM [6].

3.2. Potential Benefits for Numerical Solutions

Machine Learning and classical Finite Volume methods combined have a potential to revolutionize the field in three aspects, namely: accuracy, speed, and flexibility and inverse capabilities.

High-order accuracy of traditional FVM begins to break down under features of strong nonlinearities or sharp gradients [8], thereby diminishing stability and precision in controlling fluid motion. ML models, especially PINNs and their variants, can learn sophisticated non-local solution structures that are difficult to capture with reconstruction schemes built on polynomials [1], [4]. E.g., Hasan et. Through a comparative study using a set of 1D nonlinear convection-reaction-diffusion equations, they observed that PINNs thus provided significantly smaller L2, L1 and Linf error norms than the Galerkin Finite Element Method (a method that has a similar but still distinct mathematical basis with respect to FVM) [4]. This extra accuracy is warranted: forcing the PDE (and its constraints) holds globally at train, yields more physically consistent solutions, particularly in or on regions where the physics are important, e.g. acceleration of a boundary layer or shock front [4], [8]. Moreover, hard constraint enforcement for boundary conditions [5] can eliminate boundary errors, which are often introduced by traditional methods.

Massive computational savings: Possibly the most attractive feature of learning-augmented methods is their enormous speedup over current methods especially for parametric studies or online applications. Once a model (e.g., a DeepONet, or a well-trained PINN) is built, the solution at new parameter space coordinates or even spatial-temporal coordinates can typically be evaluated in a time that is several orders of magnitudes faster than a complete FVM simulation [7], [5]. For instance, FLARE over 300X speedups via ML-abetted reduced-order models (ROMs) over their full-order finite volume mesh (FVM) counterparts [13] For example, ML can accelerate a single bottleneck within the same simulation. For example, Li et al. Finite Volume Graph Network (FVGN) [9] integrates properties of FVM and a property of graph neural network. The model exhibited a 77% higher prediction accuracy and 56% lower training time for unsteady flow fields, compared to a strictly data-driven solution. In addition, separable PINNs (sPINNs) have been demonstrated to provide up to 60 x speedup for high-dimensional problems by reducing the dimensionality of the solution to lower-dimensional subnetworks [4], [8].

More Flexibility and Generalizability: Traditional FVM would require a new (and often expensive) simulation whenever there is a change in parameters, geometry, or boundary conditions. ML-enhanced methods (especially operator learning frameworks 4, including DeepONets [8] and SONets (Sub-Operator Enhanced Neural Networks) [7]) are tailored for adaptivity. As an example, SONets (Yu et al. If the model learn these sub-operators, then the generalization to other parameter (e. g. to a new diffusion coefficient) can be done

simply by retraining at most a small part of the network instead of the entire model from scratch$T7$. This characteristic is very useful in applications such as uncertainty quantification, design optimization, and digital twins, that need the ability to explore the design or parameter spaces quickly.

Solving Inverse Problems Inverse Problem is a traditional aim of FVM Inverse problems of reconstructing unknown material properties, boundary conditions or source terms based on only a few and often noisy measurements are frequently ill-posed and computationally prohibitive with conventional approaches. This is where PINNs shine since we can also treat the inverse problem naturally within the same loss function as that of the forward problem. All unknown parameters are simply thought of as any other trainable system variable and the network is tuned to fit the observed data but disentangles the fitting through the governing PDE [1], [13]. This work has resulted in some of the most potentially transformative applications, such as inferring hidden flow fields from sparse data produced in particle tracking experiments [11], or estimates of subsurface hydraulic properties from concentration samples [12].

4. Integration of Learning Techniques with Finite Volume Methods

4.1. Framework for Combining Learning with Traditional Methods

Stricter versions of ML and DL using solely FVM may proceed to generate a hybrid framework that not only aims to succeed well outside the natural bounds of numerical solvers using traditional techniques, but in addition preserves and utilize the inherent advantages of classical numerical methods such as conservation and stability.While learning- based techniques are not alternative methods of FVM, they intend to work alongside and thus provide a hybrid method capable of both utilizing experiential data to enhance solution approximations and leveraging the pattern recognition and universal approximation capabilities of neural networks to improve the accuracy, stability, and compute efficiency of the underlying numerical methods.

ML as a Surrogate/Corrector: Here, an ML model is learned as a surrogate for a single, costly, or error-prone module in the standard FVM workflow This could involve:

Below are some examples of components of computational fluid dynamics (CFD) that could incorporated with a machine learning model: o Learning Flux Limiters or Reconstruction Schemes: Rather than use pre-defined (high-order) schemes for reconstruction (such as MUSCL or WENO) at every cell interface, an ML model can be trained to predict local- optimal reconstruction coefficients or flux limiters based on local flow features, potentially achieving higher-order accuracy at reduced numerical dissipation or oscillation [6]. This is similar to the concept of "data-driven discretization", which we explored in [11] where ML learned to predict derivatives on a coarser grid.Predictive Closure Models: For complex phenomena like turbulence or multiphase flow, ML models can be trained to predict sub-grid scale stresses or interfacial forces, functioning as advanced, data-driven turbulence models or closure relations [6].

o ML – Enhanced Iterative Linear Solvers: The iterative solution of the large, sparse linear systems [7] that arise from implicit FVM discretizations can be accelerated through the use of ML models to make more accurate initial guesses or to learn preconditioners [8]. Physics-Constrained ML as a Solver: This more innovative technique embeds the tenets of FVM directly into the architecture or loss of a ML model to directly exploit the neural net to obtain a solution that we know will automatically obeys the integral form of the conservation laws. For example, architectures that work based on the finite volume graph network (FVGN) [9]. In FVGN, the governing equations (e.g., Navier-stokes) are not discretize in the classical sense. Instead of predicting the surface fluxes, we train a graph neural network (GNN) to predict the fluxes along the cell faces. The loss function consists of terms that impose the residual of the integral form of momentum and continuity equations per each control volume (CV) and data or boundary condition losses. One layer implements the spatial integration of fluxes (spatial integration layer, SIL) to update the state cell-centered as the preferred FVM way to perform the update step. This ensures that whatever solution is learnt, it is bounded in a sense that it complies with the fundamental conservation

laws of an FVM [9]. In fact, the strategies for PIK network architectures [12] and those for sPINNs [4] indicate how focusing a specialized architecture towards embedding or enabling effective PDE solution [84] could easily be modified to implement FVM integral constraints or enforcers.

A good training dataset is the key enabler for both these approaches. This generally involves substantially broad ranges of parameters, boundary conditions, and geometries with high-fidelity FVM (or any other standard) method simulations. This dataset as a whole has direct impact on the success and generalization of a trained ML model, as described in (13) [13]. For inverse problems or data assimilation, the training may also include experimental or sparsely observed data.

4.2. Implementation Steps for Learning-Enhanced Finite Volume Methods

Implementing a learning-enhanced FVM requires a structured, multi-stage process that carefully blends numerical simulation with machine learning engineering. The following steps outline a general workflow:

1. **Problem Definition and Data Generation:**

    - Clearly define the governing PDEs, domain geometry, and the range of physical parameters (e.g., Reynolds number, diffusion coefficient) and boundary/initial conditions of interest.

    - Generate a comprehensive, high-fidelity training dataset using a well- validated traditional FVM solver. This dataset should include snapshots of the solution field (e.g., velocity, pressure, concentration) at various time steps and for various scenarios. For surrogate modeling, this is the primary. data. For physics-constrained solvers, this data can be used for pre-training or as a reference for loss computation.

    - Process the data: normalize in-puts/outputs, calculate derived values (e.g., gradients, vorticity depending on the task), and organize it accordingly (e.g., in graphs for GNNs, in tensors for CNNs).

2. **Model Architecture Selection and Design:**

    - Select a specific ML architecture depending on the type of problem structured vs. unstructured grids, steady vs. unsteady and on how the ML outputs will be integrated surrogate vs. physics-constrained solver).

        - For structured grids, Convolutional Neural Networks (CNNs) or sPINNs [4] are suitable.

        - For unstructured grids, Graph Neural Networks (GNNs) like FVGN [9] or MeshGraphNets [Pfaff et al., 2020] are preferred.

        - For operator learning (solving for different parameters/ICs), DeepONets [7] or SONets [7] are powerful choices.

        - For enhanced accuracy and robustness, consider newer architectures like KANs [6] or PIKANs [2].

    - Design the network to incorporate physical knowledge. For physics- constrained solvers, this means designing the loss function to include PDE residuals (in strong or, preferably, weak/integral form for FVM), boundary conditions, and potentially data fidelity terms [1]. For surrogate models, hard

constraints like those using Approximate Distance Functions (ADF) [5] can be embedded in the network's output layer to ensure boundary conditions are exactly satisfied.

3. **Training Strategy:**

   - Define the composite loss function. For a physics-constrained FVM solver like FVGN, this would be:

$$\mathscr{L} = \omega_{\text{PDE}}\mathscr{L}_{\text{PDE}} + \omega_{\text{BC}}\mathscr{L}_{\text{BC}} + \omega_{\text{IC}}\mathscr{L}_{\text{IC}} + \omega_{\text{Data}}\mathscr{L}_{\text{Data}}$$

where $\mathscr{L}_{\text{PDE}}$ is computed from the integral form residual over each CV, and ω are balancing weights. Adaptive weighting schemes, such as those based on the Neural Tangent Kernel (NTK) [9] or self-adaptive methods [159], can be employed to dynamically adjust these weights during training for better convergence.

   - Select an optimizer. A common and effective strategy is to use Adam [4] for the initial, faster convergence phase, followed by L-BFGS [185] for fine- tuning to achieve higher accuracy [4].

   - Implement techniques to stabilize training, such as gradient clipping [13] or noise injection [2].

   - Utilize adaptive sampling strategies that concentrate collocation points in regions of high PDE residual or solution gradient to improve accuracy where it matters most [7].

4. **Integration and Validation:**

   - Embed the trained ML model into the FVM framework. For a surrogate, this means replacing the targeted subroutine (e.g., the Riemann solver) with a call to the ML model. For a physics-constrained solver like FVGN, the entire "solve" step is replaced by a forward pass of the network.

   - Rigorously validate the enhanced methodology against the original FVM solver and, if available, analytical solutions or high-fidelity experimental data. Metrics should include not only global error norms (L_1, L_2, L_∞) but also specific quantities of engineering interest (e.g., lift/drag coefficients, Nusselt number) and assessments of solution stability over long time integrations.

   - Conduct ablation studies to investigate the contributions of components (for e.g., the physics loss term, the specific architecture choice) to the performance outputs.

5. **Deployment and Refinement:**

   - Use the LE-FVM as a target for applications, where faster simulations or higher accuracy are needed.

   - Online fine-tuning or transfer learning In particular, if the model finds itself in a situation that is out-of-distribution with respect to its training, the adaptivity to the region may be achieved using techniques such as meta-learning [7], or "online" fine-tuning, with little additional computational cost.

Iterative coupling: This well-defined nature of our approach ensures that integration of ML with FVM approaches is systematic and repeatable, based upon physics-informed likelihood and machine-learning-principles, which will lead to reliable and accurate augmented simulations as well as ML models.

5. Case Studies and Simulation Results

However, the sheer power of the Learning-Enhanced Finite Volume Methods (LE-FVM) paradigm will only be realized through systematic benchmarking against conventional FVM solvers for different types of case studies. Recent, high-impact works on canonical nonlinear convection-diffusion-reaction (CRD) problems are summarized here. This analysis draws heavily from the detailed linked multi-method evaluation by Hasan et al. What sets our approach apart is that it allows us to obtain a strong statistical ground truth for comparison (functionally analogous to one proposed in [4]) while unifying material information from the state-of- the-art FVGN [9] and SONets [7] architectures.

5.1. Benchmark Problems and Experimental Setup

In order to facilitate a fair and insightful comparison, we take as our test cases four known, one- dimensional nonlinear CRD equations which embody different physical phenomena and hence, numerical difficulties:

1. **The Burgers' equation:** A prototypical model of nonlinear wave propagation, shock formation, and turbulence, has a linear diffusion process balanced by a nonlinear convection process. This method is especially suitable for sharp gradients with no oscillations [4], [9].Eq.(5&6)

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} \quad (5)$$

(6)

2. **Fisher's Equation:** A reaction-diffusion equation modeling phenomena like population genetics and chemical kinetics, featuring a logistic source term that leads to traveling wave solutions [4].Eq.(7)

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + (1-u) \quad (7)$$

3. **Burgers–Huxley Equation:** A hybrid model combining features of Burgers' and Huxley equations, used to describe nerve pulse propagation, combustion, and other processes involving advection, diffusion, and reaction [4].Eq.(8)

$$\frac{\partial u}{\partial t} + \alpha u\frac{\partial u}{\partial x} - 2 = \beta u(1-u)(u-\gamma) \quad (8)$$

4. **Newell–Whitehead–Segel (NWS) Equation:** A reaction-diffusion equation that models pattern formation, such as Rayleigh-Bénard convection, with a cubic nonlinear source term [4].

$$\frac{\partial u}{\partial t}$$

$$= D \frac{\partial^2 u}{\partial x^2} + \alpha u - \beta u^3$$

Experimental Setup: In [4], we configured all models (the classical FVM and its new proposed LE-FVM variant models such as PINN) to adopt identical initial and boundary conditions along with reference analytical solutions for model validation. The different measures of predicted accuracy were based on standard error: the absolute error (AE), maximum absolute error (MAE) and root mean squared error (RMSE). For the PINN based LE-FVM models, a majority of the training used 4 hidden layers with 20 neurons using the tanh (hyperbolic tangent) activation functions. It was used in conjunction with the Adam optimizer for 5,000 epochs and 3,000 iterations of L-BFGS for a high degree of convergence [4]. For FVGN [9], a physics-informed model based on unstructured grid was designed with physical constraints via the integral form of the Navier-Stokes equations, and a "twice-message aggregation" mechanism to expand its receptive field and obtain geometric generalization.

5.2. Quantitative Performance Analysis

The quantitative results reveal a clear and consistent trend: LE-FVM, particularly when implemented via PINN, outperforms traditional FVM in terms of solution accuracy for the majority of the tested problems.

Table 1: Comparative Error Metrics for Burgers' and Fisher's Equations (Adapted from [4])

Equation	**Method**	**Time (t)**	**Max. Abs. Error (l_∞)**	**RMSE**	**Standard Deviation**
Burgers'(Re=1)	**FVM**	0.04	5.07×10^{-2}	0.024	0.018

	PINN	0.04	$2.93 \times 10^{-'}$	0.0004	0.0003
	Improvement	-	**~173x**	**60x**	**60x**
Fisher's	**FVM**	0.10	6.27×10^{-3}	0.003	0.002
	PINN	0.10	$8.50 \times 10^{-\mu}$	0.0002	0.0001
	Improvement	-	**~74x**	**15x**	**20x**

Table 1: Comparison of maximum absolute error and root mean square error (RMSE) using PINN and standard FVM (maximum absolute error that is sustained via spectrum propagation as t increases) for the Burgers' equation at t=0.04. In this case PINN reduced the maximum absolute error and RMSE by factors of >170 and 60 respectively, compared to the standard FVM. There is a similar, but not as large of a, improvement for Fisher's equation. Such findings are further validated by visual comparisons (e.g. Figures 2, 3, 4, 5, 6 and 7 in [4]) where it is evidenced that PINN predictions closely corresponds to the analytical solution, whereas the FVM solution shows evident numerical diffusion, in particular immediately upstream and downstream of the shock front in the case of Burgers'.

For the NWS equation, the PINN again demonstrated superior performance, achieving an RMSE of $3.0\times10^{-\mu}$ at t=0.05, compared to $2.03\times10^{-'}$ for the FVM a nearly 7-fold improvement [4]. Findings in [4] are also statistically analyzed confirming the same trends. The Wilcoxon signed-rank test (WSRT), a non-parametric test for comparing paired samples, shows statistically significant p-values ($p < 0.05$) for Burgers', Fisher's and NWS equations indicating that the improvement in accuracy offered by PINN is not due to random chance.

A rather more specific result came from the Burgers–Huxley equation. The RMSE of the FVM was slightly less than that of the PINN (0.001 vs. 0.004), and the standard deviation increased over time for both methods, though this appeared to be greater for the PINN. However, in that particular case, the FVM performance was less stable with regards to the time step as indicated by the Coefficient of Variation (CV) test in [4] in which the CV of the error was more prominent and consistent over the time steps with the FVM as well. This brings us to another important point that accuracy is a metric not the only one; stability and consistency are also essential for a numerical method to be mathematically robust.

5.3. Efficiency and Generalization: Insights from FVGN

Although the results of the PINN method presented in [4] are optimized on the accuracy for 1D problems, the study in [9] allows for important remarks on the efficiency and generalization ability of the method, especially for complex/unstructured 2D geometries.

Efficient Improvements: FVGN was constructed in such a way as to strike a balance between accuracy and any additional computational expenditure. The model uses a directed graph (and therefore unidirectional edges) rather than an undirected one, achieving a 56% reduction in training time on average across the HYBRIDFLOW dataset (comprising cylinders, airfoils, and rectangular columns) [9] against a standard MeshGraphNet (MGN) whilst shrinking its hidden feature space. Importantly, this efficiency improvement was achieved without sacrificing accuracy. Specifically, FVGN

achieved 77% better prediction accuracy (lower RMSE) of the velocity fields as compared to the same MGN baseline [9]. This shows how LE-FVM can be designed to not only be more accurate, but also orders of magnitude more efficient than purely data-driven deep- learning models if not, compared to traditional FVM in most forward-simulation problems.

Physics Constrained Generalization Capability: The primary benefit of using a physics- constrained LE-FVM (like FVGN) is its generalization over unseen geometries. The FVGN model was tested on an elliptical obstacle, a geometry that had not been included in the training set and was able to qualitatively reproduce the flow field and accurately replicate the vortex shedding frequency [9]. Even though the absolute error was worse at predicting pressure compared to the shapes seen during training, the fact that the model can deal with a new unseen geometry without retraining is a very large step forward given that traditional FVM would still be unable to handle a completely different geometry without a complete re-meshing and re-simulation. This geometric versatility is a result of directly embedding the conservation expression of the handling finite volume method (FVM) within the loss function, allowing the model to learn governing physical processes rather than (over-) memorizing the solution on a particular grid set.

5.4. Discussion of Results

Taken together, the findings from these case studies tell a powerful story:

1. AbsenteSuperior performance: As seen from the experiment results of 1D nonlinear CRD problems, LE-FVM, mainly based on the PINN framework, consistently produces orders of magnitude of higher accuracy over traditional FVM with various error norms and statistical tests [4].

2. Increased Stability and Robustness: LE-FVMs provide better stability and robustness: the average error may be slightly worse (e.g. Burgers-Huxley) [4], but individually, the solutions become a lot more stable and consistent in time.

3. Computational Efficiency (Inference): With LE-FVM, solutions can be evaluated instantly at any spatio-temporal point after training for PINN and FVGN like models. Such "mesh-free" inference capability provides orders-of-magnitude speedups over running a full FVM simulation on each new parameter set for typical parametric studies or real-time applications [13], [9].

4. Geometric flexibility: Physics-constrained LE-FVM models trained on unstructured grids (e.g., FVGN) exhibit an astonishing capacity to generalize to out-of-training-geometry which is an arduous and expensive procedure for conventional mesh-based FVM [9].

In short, these case studies demonstrate that the combination of learning with the basic tenets of the Finite Volume Method produces a new class of solvers that are faster, more powerful, and more versatile than traditional solvers while remaining accurate and stable, in particular when it comes to complex nonlinear problems.

6. Comparative Analysis: Traditional vs Learning-Enhanced Approaches

6.1. Accuracy Improvements Achieved Through Learning Techniques

Over the last years, machine learning, and more particularly physics-informed neural networks (PINNs) and their richer variants, have been coupled with the Finite Volume Method (FVM) formulation yielding remarkable and frequently transformative advances in the non-linear convection-diffusion problem solution accuracy. This is a change of paradigm vs point improvement in terms of the way the solution is approximated and constrained.

By design, traditional FVM, while being an attractive approach due to its robust and conservative nature, is limited by local polynomial reconstructions and fixed stencil based discretization patterns. However, these approximations may be unable to capture highly nonlinear behavior, sharp gradients or discontinuities without inducing significant numerical diffusion (smearing) or dispersion (oscillations) [8], [9]. It is usually precipitate on the grid resolution by asking for unrealistically fine grids for a proper fidelity outcome in complex cases.

On the other hand, the FVM methods enhanced with learning exploit the universal approximation potential of neural networks [27]. For example, PINNs do not discretize the domain as per the traditional method, but rather they look for a global differentiable function that minimizes a composite loss function that encapsulates the PDE residual, boundary conditions, initial conditions and possibly sparse data [1], [12]. It is this ability to utilize global context that enables the network to learn more complex, non-local structures in the solution that would require polynomial-shaped FVM schemes to represent.

The empirical proof for this boost in accuracy is powerful. This is the wider study by Hasan et al. Similar conclusions about the advantages of {the }GFEM (which shares most of the accuracy properties as both {high-order} low-order FVM as well as {PINN} based

{architectures}) over {PINNs} are made in [4], where a rigorous, multi-metric comparison between these two approaches is provided for four canonical 1D nonlinear PDEs. The results are unequivocal:

- For Burgers' Equation @ t=0.04, PINN cut the max absolute error ( ) by >170x and RMSE by 60x vs. GFEM [4].
- Fisher's Equation at t=0.10: A 74x decrease in error, and a 15x decrease in RMSE using PINN[4]
- For Newell–Whitehead–Segel Equation, across all time points tested, PINN consistently achieved significantly lower error norms than GFEM [4].

This is not just an improvement for 1D problems. Finite Volume Graph Network (FVGN) which applies modeling principles of the finite volume method (FVM) into a graph neural network for 2D unstructured grids, achieved 77% lower prediction error (lower RMSE) for velocities and was also trained 56% faster than a corresponding MeshGraphNet [9]. This emphasises that the accuracy advantages apply to relatively complex, multi-dimensional geometries.

In addition, more advanced architectures such as Physics-Informed Kolmogorov-Arnold Networks (PIKANs) [12] and separable PINNs (sPINNs) [84] have an increased potential.utilizing theKolmogorov-Arnoldrepresentation theorem, have been found to be more robust to noise, while being able to achieve comparable or higher accuracy than a MLP-based PINN with fewer parameters[12], [49]. For the high-dimensional problems, sPINNs can then obtain a speedup of up to a factor of 60 while achieving comparable accuracies by breaking down the solution into univariate functions [84].

What drives this better accuracy is the physics-informed loss function. The neural network is regularized to generate not just data-consistent solutions but also physically realistic ones by penalizing the violation of the governing PDE. In a way, this comes in handy close to such crucial regions like boundary layers [4], [8] or shock fronts, where the traditional methods fail frequently. Such performance is additionally shown to be more stable, with techniques such as hard constraint enforcement for boundary conditions [35] and the use of adaptive activation functions [61].

6.2. Performance Efficiency Gains in Computational Time and Resources

Though the initial price to train a learning-augmented model may be steep, the return in efficiency in the inference or deployment stage can be transformative, especially for parametric studies, real-time solutions or inverse problems. This is a paradigm shift from a typical FVM workflow.

Scaling of computational cost with number of grid cells and time steps in traditional FVM Rather than solving a single forward problem, they tend to be quite computationally intensive. When we need thousands of simulations (e.g., uncertainty quantification, design optimization, or inverse problems), the cumulative cost quickly becomes prohibitive. A new set of parameters or boundary conditions typically requires a new mesh and an independent simulation.

Training Improved Efficiency: A physics-based model (e.g., a PINN, DeepONet, or SONet) is trained to evaluate the solution at virtually all points in the spatio-temporal domain in a near instantaneous manner. Most dramatic efficiency gains are reaped in this "mesh-free," inference capability.

ML Enhanced Reduced-Order Models for Inference Speedup on Parametric Studies: Speedups of greater than 300x over full-order FVM models have been achieved with ML enhanced ROMs [13]. Thus, after training a DeepONet on a family of PDEs with different initial conditions, the trained network can produce the solution for a different initial condition in less than a few milliseconds, without anymore the need for a full FVM simulation [7]. In a similar sense, SONets achieve the same for a new diffusion coefficient but with orders of magnitude less computational cost than re-running a traditional solver [7].

Efficient Usage → Model Architecture The efficiency can also come from the design of the learning model itself. The FVGN study [9] illustrates this point very well. The model achieved a 56% reduction in training time and improved accuracy as compared to a standard MeshGraphNet, by reducing its hidden feature space through the use of a directed graph (unidirectional edges rather than an undirected graph) [9]. This shows how LE-FVM .can be designed to be not just more accurate, but also much more efficient than purely data-driven deep learning models.

Speeding up Conventional Solvers: LE-FVM is not a complete replacement in all cases. It can be employed to speed-up a traditional FVM workflow on certain components that are compute-intensive. And ML models can substitute for application, such as iterative linear solvers [18], can predict the optimum sizes of time-steps, or can serve as fast and accurate closure models for turbulent flows or multiphase flow [11], [12]. Hence, this hybrid methodology advances the advantages of both paradigms and dramatically improves the total runtime without a complete removal of either paradigm.

One the biggest advantage it brings is handle high-dimensional problems and intractable problems which traditional methods cannot handle. Recent studies have demonstrated that appropriately designed PINNs can address high-dimensional PDEs (e.g., Hamilton-Jacobi- Bellman equation in 100,000 dimensions) within 60 minutes using a single GPU, a problem for mesh-based methods such as FVM or FEM [6], [5] so the PINNs outperforming the traditional mesh-based methods.

7. Key Challenges in Implementation and Adoption of Learning-Enhanced Techniques

The integration of machine learning, particularly deep learning, with the Finite Volume Method (FVM) represents a paradigm shift with immense potential. But realization is the hardest challenge, transitioning from potential in theory to adoption over industry and science, but of higher quality. These involves data requirement, computational complexity, software integration and of course the core problem of model interpretability & trust.

7.1. Data Requirements for Machine Learning Models

The performance of any learning-augmented FVM (LE-FVM) critically depends on the quality, quantity, and richness of the data upon which it is trained. This presents a multi-faceted challenge:

- **High-Fidelity Data Generation:** For supervised or hybrid learning approaches, vast amounts of high-fidelity training data are required. · This usually requires performing conventional and high-resolution FVM (or FEM/FDM) simulations over a large parameter space, which includes a variety of boundary conditions, initial conditions, material properties, and geometries [13]. Computing this data is resource and time intensive and generally the computational efficiency improvements that LE-FVM offers on a single problem do not offset the overhead. Thus, as pointed out in [13], the efficiency of ML methods strongly depends on the availability of labeled data, which if non-sufficient and non-representative can make the model learn badly (low RMSE and low accuracy, low universality and capacity for generalization) in order to provide adequate predictions.

- **Data Sparsity and Noise:** In many real-world scenarios, particularly inverse problems or data assimilation tasks, the available data is sparse, noisy, or incomplete [1]. While physics-informed methods like PINNs are designed to handle such data by incorporating governing equations as soft constraints, they are not

immune to its effects. For example, the model may learn the noise instead of the actual physics with noisy data or worse, under sparse data it becomes prone to poor extrapolation and thus high epistemic uncertainty [1]. For instance, approaches have been developed [4, 5] to quantify this uncertainty, such as the Bayesian PINNs (B-PINNs) which, however, make the training problem more complex.

- **Curse of Dimensionality**: In the case of parametric PDEs or high-dimensional state space problems, the volume of the parameter space grows exponentially. As a consequence it becomes computationally unfeasible to densely cover this space with more training data. This poses an important restriction for the methods relying on large pre-calculated databases. While operator learning approaches (DeepONets [7], SONets [7]) partially circumvent this by learning mappings between function space, they still require sufficiently many "tasks" (i.e. diverse initial conditions) for training. Fully data-driven models, as noted in [9], may require long time periods to train that inhibit rapid iterations, an aspect we argue is ill-suited to many more complex unsteady flows.

- **Computational Resources for Training —** One of the main challenges of deep neural networks is the need for a substantial amount of computational resources for training, due to the observation that more complex and larger deep neural networks tend to outperform smaller networks when trained at scale on large high-dimensional datasets. The massive size of dataset and the reduction of training time requires the staged provision of high-performance computing (HPC) resources, especially GPUs [13], [9]. Thus, training a complex LE-FVM model can take hours or days, which is a huge cost in advance of any inference speedup.

7.2. Integration Complexity with Existing Computational Frameworks

Integrating LE-FVM into the established ecosystem of CFD and multiphysics simulation is not a trivial plug-and-play operation. It involves navigating a complex landscape of software engineering and numerical analysis:

- Architectural Mismatch: Traditional FVM solvers are often large, monolithic codes written in languages like C++ or Fortran, optimized for HPC environments. In contrast, modern ML frameworks (e.g., PyTorch, TensorFlow) are predominantly Python-based and designed for flexibility and rapid prototyping. Bridging this gap requires developing robust, efficient interfaces, which can be a non-trivial software engineering task [7]. This heterogeneity can lead to performance bottlenecks and complicate debugging.

- Data Compatibility and Pipeline Management: Getting the right data out of traditional solvers (solution fields, gradients, fluxes) in a way that is amenable to ML training can be unwieldy. This entails constructing data pipelines across various mesh types (structured, unstructured), data formats, and possibly disparate solver outputs. A reliable prediction is not possible if the training dataset does not cover the various situations that can occur in the operational phase.

- High Computational Requirements (Training Stage): The inference with a trained LE-FVM model is usually fast, however, the training stage can be computationally expensive, introducing another cost into the simulation workflow [7]. This includes

- the raw compute power but also the time and aero-stationary of the art machinery needed to perform hyper-parameter search, model selection, and validation. The combined methodology is complex and can require large investments in the development and test of software [7].

- Validation and Verification (V&V): Confidence in the results from any LE-FVM is perhaps the most important of all genetic principles. Numerical methods rarely shed insight into practical applications, so traditional V&V procedures are well-known but may not appear to be as well suited for hybrid ML models. How do we check that neural network is actually learning to satisfy some physical law? How do you justify its performance on some scenario not in its training distribution? LE-FVM specific rigorous verification and validation protocols remain in development. Because of the above, without rigorous validation, there is a danger that hybrid approaches can produce results that differ significantly from expected behaviour based on the

original FVM method [7]. That "black-box" quality is an enormous roadblock to adoption for safety-critical applications.

- Shobha: · Non-Technical Challenge — Model Interpretability and Trust: Probably the most serious of the challenges is the most non-technical — Lack of model interpretability associated with many deep learning models. In contrast to a typical FVM scheme, where the discretization and resulting numerical errors can be traced, neural networks are often black-boxes. Such a property, often referred to as "black-box", can impede engineers and scientists from identifying the reasons for a model to produce a particular prediction, troubleshoot a failure, or develop trust on the results of its predictions in extrapolative or strictly safety-critical situations [9]. Attention mechanisms [6] and information bottleneck theory [5] have been investigated to provide insight into model behavior, but to date producing the same degree of interpretability traditionally obtained in scientific computing has not been achieved.

To conclude, although LE-FVM has the potential to transform ways in which models are built and operated, achieving its successful implementation will hinge on overcoming several fundamental bottlenecks related to data, computational cost, software integration, but most importantly validation and user trust. Thoroughly addressing these challenges is part and parcel as important for the long-term success of this field as developing the underlying algorithms themselves.

8. Future Opportunities in Research and Development

Machine learning, especially natural integration of deep learning with finite volume method (FVM) is by no means an end itself, but rather the beginning of an entire new era in computational science and engineering. Though today's Learning-Enhanced Finite Volume Methods (LE-FVM) achieve a favorable balance between accuracy and performance (the Holy Grail for many problems), far greater bounds are yet to be approached. It then explores the future of this area: The future of this area will be determined both by the advancement of algorithmic innovation, increasing range of application, and the development of a reputable yet accessible software ecosystem.

8.1. Enhancements in Algorithmic Efficiency through Advanced Learning Techniques

The relentless pursuit of greater algorithmic efficiency will continue to be a primary driver of research. Future developments will focus on creating more intelligent, adaptive, and resource-conscious learning architectures that can seamlessly augment or replace components of the traditional FVM workflow.

- **Next-Generation Neural Architectures:** · Kolmogorov-Arnold Networks (KANs) [6] and their physics-informed version, PIKANs [12], boast a recent, and to some a revolutionary, advancement in this area. Unlike standard multilayer perceptrons (MLPs) that only learn weights between fixed activation functions, KANs learn the activation functions themselves. This has led to lower parameter counts with improved accuracy and greater robustness to noise [12], [4]. We envision future work in special KAN layers and training protocols for fluid dynamics and other FVM-governed phenomena, which could result in higher accuracy models that are also cheaper to train and deploy than the MLP-based analogs.

- **Meta-Learning and Transfer Learning for Rapid Deployment:** · LE- FVM is very expensive to train a new model for each type of problem/parameter set. A strong solution for this is to utilize meta-learning or in other words, "learning to learn." A representative example is the idea of meta-SONets [7], where the model is pre-trained on a variety of tasks (e.g., PDEs with different initial conditions) so that a new, unseen task can now be quickly adapted with limited data and computation. This method converts LE-FVM from a one-time method into an agile and re-useable framework and thus render it suitable for real-time optimization, uncertainty quantification, and digital twin applications in which the ability to respond rapidly to changing conditions is essential.

- **Hybrid Solvers and Operator Learning:** The future lies in hybrid methodologies that strategically combine the strengths of traditional FVM and ML. This includes:

- Learning closure models from the operator perspective: Approaches such as [7] and [11] use models like a DeepONet or a Fourier neural operator (FNO) to learn complex global, non-local, closure models for turbulence, multiphase flow, or chemical reactions that drastically improves fidelity when used within coarse-grid FVM simulations in the place of crude empirical models.

- ML-Accelerated Linear Solverso ML blocks to replace or precondition the iterative linear solvers required for the implicitness in FVM schemes - [13]. This can pose a significant reduction in the computational bottleneck being faced in large-scale simulations.

- Reinforcement Learning for Adaptive Mesh Refinement (AMR): Using RL agents to efficiently, via either online or offline training, determine physics-informed choices where and when to refine or coarsen the computational mesh, thereby optimizing the often competing demands of accurate and cheap solutions dynamically during a simulation [10], [8].

- **Exploiting High-Performance Computing (HPC):** Since LE-FVM models, especially GNN based FVGN [9], are getting constantly more complicated, usage of contemporary HPC resources will be essential. Future work will thus center on distributed training algorithms and cost-optimal model architectures for

- A multitude of GPU and TPU clusters to train larger, stronger models on massive datasets States that can solve on unstructured grids, like FVGN, are especially suitable for parallelising over complex, decomposed domains.

8.2. Potential Applications Beyond Convection-Diffusion Problems

LE-FVM principles are very generalizable. This progress in improving the resolution of nonlinear convection-diffusion equations has potential to enable groundbreaking applications across a broad array of scientific and engineering fields.. Multiphysics and Multiscale Modeling: LE-FVM provides a strong foundation for tackling complex multiphysics problems. This includes:

o Fluid-Structure Interaction (FSI): Creating LE-FVM models which solve the Navier-Stokes for the fluid and the elasticity equations for the structure in tandem, with the ML component learning the complicated, often nonlinear, coupling at the interface [7], [1]. That could transform how heart valves, aircraft wings and offshore structures in turbulent water are simulated.

o Reactive Flow and Combustion: Incorporating ML for chemical reaction kinetics and turbulent combustion simulations (CFD solver based only on the control-volume approach) to optimize complex systems such as engines, furnaces, or even fire dynamics [5], [11].

Poroelasticity and geomechanics: Simulation of flow of fluid and deformation of porous media—an important problem in applications including oil reservoir simulation, carbon sequestration, and geothermal energy extraction [1], [7]. Prospects of LE-FVM in biomedicine and personalized medicine: LE-FVM has a broad prospect in biomedicine:

o Patient-Specific Hemodynamics: LE-FVM models can quickly produce highly accurate, subject-specific, arterial and venous flow simulations from input medical imaging data (e.g., MRI, CT), useful in the diagnosis of annerysms and stenosis and for surgical planning interventions [4], [1], [9].

Drug Delivery and Pharmacokinetics: CMINN [5] and its outgrowths can be generalized to model the transport and reaction of a drug in the organism to predict their efficiency and dosage regimens for every individual patient. Explore other advanced applications related to soft tissue behavior as well as nutrient and growth factor transport in tissue-engineered constructs using LEpoweredFVM, therefore speeding up the progress in regenerative therapies [1], [8]. High-Scale Environmental Science and Climate Modeling: LE-FVM's capacity of tackling complex geometries and multi-scale phenomena is especially suitable for environmental applications:

o Pollutant dispersion and water quality: Transport and reaction of pollutants in air and in water bodies (used in environmental impact studies and remediation planning) can be simulated at lower computational cost and greater fidelity as compared to the current approaches [7], [11].

o Modeling Wildfires and Floods: Creating rapid, physics-informed simple models for areas of complex, multi-physics phenomena such as wildfire spread or flood inundation so that they can be done in real-time for emergency response [13].

o Subsurface Flow and Transport: Improvements on simulating groundwater flow, contaminant transport, and [1] and geothermal systems in highly heterogeneous geological formations [8]; Materials Science and Manufacturing: LE-FVM opens up new highways of unprecedented insights into material behavior and manufacturing processes.

o Additive Manufacturing: Simulation of temperature fields, melt pool dynamics, and residual stresses during 3D printing processes allowing to optimize printing parameters and avoid defects to enhance part quality [2].

Composite Processing: Simulating flow of resin and heat transfer during the curing of composite process to achieve required properties, as in [8].Similar High Level Functions o Microscale Heat Transfer: Predicting heat conduction in thin films and nanostructures under ultrafast lasers, which is needed in advanced manufacturing and electronics [1], [2].

Financial Engineering: While there may seem to be gaps, the equation related to convection-diffusion-reaction has a mathematical form similar during the Black-Scholes equation for option pricing. It is particularly promising to adapt LE-FVM techniques, especially those able to deal with high-dimensional problems [9], to price more general features of complex financial derivatives or to manage risk in a more efficient way [3].

9. Conclusion

While the numerical solution of nonlinear convection-diffusion problems has largely been dominated by the use of the Finite Volume Method (FVM), machine learning especially deep learning to address the learning process heralds a new era for convection-diffusion problems. We have established the basis of Learning-Enhanced Finite Volume Methods (LE-FVM) in this paper, presenting LE-FVM as a fundamentally different philosophy from a modest step away from FVM towards FVM with a predictive engine merely bolted on, by merging the rigorous, physics-grounded heritage of FVM with the powerful, data-driven pattern matching capabilities of modern-day neural networks. We reformulate LE-FVM in applications to GSV, justify its borrowing from the theoretical and empirical literature developed recently [4], [9], [12] and show that LE-FVM outperforms conventional FVM in every application. PINNs and their recent advanced forms such as FVGN and PIKANs have shown excellent performance to effectively capture complex features of the solution, e.g., sharp gradients, shock fronts, and turbulent structures, significantly reducing error norms. In particular, Hasan et al. performed a benchmark study of 1-D nonlinear PDEs In general, [4] showed ca. For the Burgers' equation, corresponding to an unprecedented level of accuracy that mesh-based methods fail to achieve except at intractably fine discretizations, we observe > 100 fold reductions in maximum absolute error factors with PINNs compared to mesh-based methods. In addition to its fidelity, LE-FVM can also offer some orders-of-magnitude improvements in computational performance relative to certain workflows. While the upfront costs of training a physics-informed model are high, the inference phase is high-scale, with the cost being minimal. A trained model like DeepONet or FVGN can predict solutions of any configuration of parameters or initial conditions in milliseconds, with no iterative, iterative FVM simulation involved [9], [7]. These capabilities enable several applications that depend on rapid parametric studies, real-time control, uncertainty quantification, or digital twins deployment, but become computationally prohibitive when using traditional approaches. As an example, the FVGN architecture achieved 77% more accurate predictions with 56% less training than a baseline MeshGraphNet, illustrating the non- adversarial accuracy-efficiency tradeoff [9]. There are all kinds of trends converging to create a booming future for LE-FVM. Initially, it can be employed to reduce the number of trainable parameters, and noised data, and also to develop new, more powerful, and more sensitive neural architectures such as KANs and PIKANs [12], [11]. Second, algorithmic innovations like separable PINNs (sPINNs) and novel Stochastic Dimension Gradient Descent (SDGD) methods target the "curse of dimensionality" head-on and enable the solution of PDEs in tens or even hundreds of thousands of dimensions, a scale that remains out-of-reach for traditional FVM [19], [84]. Finally, a large part of the field is moving away from relying on human intuition to inform the training process, as seen by the increasing use of adaptive loss-weighting [8], physics- driven sampling strategies [5],[2], and hybrid training regimes that massively improve convergence and robustness of [2].

However, mass adoption is a bumpy road. AbstractDeep learning models are notoriously hard to interpret and thus trust, especially for safety-critical engineering applications due to their black-box characteristics. It highlights that development of hybrid ML-FVM solvers will require a comprehensive validation and verification (V&V) protocols to ensure their robustness and reliability ([7]). Last but not least, the need for high-fidelity training data, and the prohibitively large computational resources necessary to train the model make the method impossible to be applied for most complex, multi-physics problems [13]. The hurdles to overcome will require a step change from the computational science community to produce benchmarks data sets, software frameworks and theory driven error estimator methodology. Key message: L-E FVM are not something new, an elaborate something, replacing the traditional FVM, they are rather a new evolutionary step of finite volume methods. They maintain the method's classical strengths (conservation and physical fidelity) whilst improving accuracy for complex flows and efficiency for multi-query settings – i.e. Simulation of problems that would have been intractable and daunting a few years ago will be possible with LE-FVM, and as research will ultimately address these immediate challenges and explore new frontiers of algorithmic capabilities, LE-FVM can become a standard part of the scientific and engineering toolbox. The pathway from mesh to neural net has been laid down and it converges into a paradigm that will change computation as we know it.

References

[1] E. Tandis and P. Cardiff, "Finite volume-based supervised machine learning models for linear elastostatics," *Adv. Eng. Softw.*, vol. 175, p. 103258, Jan. 2023, doi: 10.1016/j.advengsoft.2022.103258.

[2] X. Chen, K. Zhang, Z. Ji, X. Shen, P. Liu, L. Zhang, J. Wang, and J. Yao, "Progress and Challenges of Integrated Machine Learning and Traditional Numerical Algorithms: Taking Reservoir Numerical Simulation as an Example," *Mathematics*, vol. 11, no. 21, p. 4418, 2023, doi: 10.3390/math11214418.

[3] A. Egon, L. Doris, and C. Bell, "Numerical Methods for Solving Nonlinear Partial Differential Equations in Fluid Dynamics," *HAL*, Feb. 2025. [Online]. Available: https://hal.science/hal-05094230v1

[4] F. Hasan, H. Ali, and H. A. Arief, "From Mesh to Neural Nets: A Multi-method Evaluation of Physics Informed Neural Network and Galerkin Finite Element Method for Solving Nonlinear Convection–Reaction–Diffusion Equations," *Int. J. Appl. Comput. Math*, vol. 11, p. 90, 2025, doi: 10.1007/s40819-025-01904-y.

[5] K. Shukla, J. D. Toscano, Z. Wang, Z. Zou, and G. E. Karniadakis, "From PINNs to PIKANs: Recent Advances in Physics-Informed Machine Learning," *arXiv*, Oct. 2024. [Online]. Available: https://arxiv.org/abs/2410.13228

[6] T. Kossaczká, M. Ehrhardt, and M. Günther, "Deep FDM: Enhanced finite difference methods by deep learning," *Results Appl. Math.*, vol. 17, p. 100336, Jan. 2023, doi: 10.1016/j.rinam.2022.100336.

[7] X. Jin and H. Li, "SONets: Sub-operator learning enhanced neural networks for solving parametric partial differential equations," *J. Comput. Phys.*, vol. 493, p. 112482, Dec. 2023, doi: 10.1016/j.jcp.2023.112482.

[8] A. Scott, "Integrating Finite-Volume Methods with Time Series Analysis for Enhanced Cancer Prevention Strategies," *HAL*, Jul. 2024. [Online]. Available: https://hal.science/hal- 04663201v1

[9] T. Li, S. Zou, X. Chang, L. Zhang, and X. Deng, "Predicting unsteady incompressible fluid dynamics with finite volume informed neural network," *arXiv*, Mar. 2024. [Online]. Available: https://arxiv.org/abs/2309.10050

[10] "The Crunch Group - The collaborative research work of George Em Karniadakis," *Brown University*, Sep. 2025. [Online]. Available: https://sites.brown.edu/crunch- group/weekly-seminar-spotlight/

[11] "Recent Advances on Machine Learning for Computational Fluid Dynamics: A Survey," *arXiv*, Aug. 2024. [Online]. Available: https://arxiv.org/abs/2408.12171

[12] E. H. Quenjel, “Positive Scharfetter-Gummel finite volume method for convection- diffusion equations on polygonal meshes,” *Appl. Math. Comput.*, vol. 425, p. 127071, Jul. 2022, doi: 10.1016/j.amc.2022.127071.

[13] Ł. Łach and D. Svyetlichnyy, “Advances in Numerical Modeling for Heat Transfer and Thermal Management: A Review of Computational Approaches and Environmental Impacts,” *Energies*, vol. 18, no. 5, p. 1302, Jan. 2025, doi: 10.3390/en18051302.