SOCIAL SCIENCE REVIEW ARCHIVES
ISSN Online: 3006-4708
ISSN Print: 3006-4694
https://policyjournalofms.com

Computational Methods for Solving Partial Differential Equations: A Comparative Study of Finite-Element, Spectral and Meshless Schemes for Physics & Engineering Problems

Sadaf Shaheen[1], Aqsa Ayaz[2], Aiman Bibi[3]
[1, 2, 3] Department of Mathematics, University of Haripur, KPK , Pakistan,
Email: sadafharry99@gmail.com, aqsaayaz321@gmail.com, afm73737@gmail.com

DOI: https://doi.org/10.70670/sra.v3i4.1099

Abstract
The paper provides a brief but strict benchmarking of three popular solvers of PDEs, namely FE, spectral and meshless schemes, with the current physics and engineering test-cases. New algorithmic advances to residual-driven hp-adaptivity, Trefftz DG, GPU-accelerated spectral elements, interpolation-based RKPM, and differentiable SPH have been mentioned as updates to the study to measure accuracy, CPU/GPU time, memory footprint and parallel scalability. Poisson, advection-diffusion and Navier-Stokes experiments have shown: (i) spectral methods retain exponential convergence when applied to smooth fields, but lose to discontinuities unless neural-network coordinate-changes; (ii) high-order FEM and HDG have the best balance between geometric flexibility, stability, and software maturity; (iii) meshless methods have been demonstrated to have the best performance in extreme deformation and free-surface flows, but require neighbour-search acceleration to match the speed of FEM. Moreover, graph-adaptive mesh refinement, spectral neural networks, physics-informed preconditioners, represent hybrid ML pipelines which reduce the number of iterations by up to 25 percent without affecting fidelity. An informaticized form of decision tree derived out of the data is used to direct practitioners in selecting what method would best work, whereas open-source scripts allow reproducibility and future extrapolation to exascale and quantum-augmented computing.

Introduction
Partial Differential equations (PDEs) are critical in the numerical solution of complex problems in the real world, e.g. in physics and engineering (among other areas). The PDEs are used to model a great deal of phenomena including heat conduction, fluid mechanics, electromagnetic fields, and structural mechanics(Ahmed, 2024). The complication of these equations prohibits closed-form analytical solutions, which makes its usage in most cases impractical or impossible, making the use of computational means to approximate solutions very common. Some of the most notable PDE solvers include spectral methods, meshless methods and finite-element method (FEM)(Amir, 2023). Each of these methods has its own weaknesses and strengths, and this is why they are applicable to various problems. In the current paper, the author will make a comparative analysis of these three methods, at least in their application to the solution of PDEs in physics and engineering problems. The rationale behind this comparative study is that the problems in physics and engineering that use PDEs are becoming more and more complicated, and the associated numerical problems require more sophisticated numerical approaches to solve them effectively. Although the old approach which can be the finite difference method has long been used over many decades, new progression made on computational techniques like spectral methods and meshless methods are promising alternatives. Finite-element method is highly versatile and is

suitable in complex geometry and boundary conditions and hence it is ideal in most of the engineering applications (Jalil et al., 2025). However, spectral methods are characterized by high accuracy, especially when the solution to the problem is smooth because they can represent a solution in high order polynomials (Dar et al., 2024). Meshless techniques, where the mesh is not defined have been of interest due to their capability to solve problems with complex and dynamic boundaries, or where a mesh generation cannot be done easily (Nooraiepour, 2025). The paper will be used to compare these methods according to their performance in solving different classes of PDEs considering the accuracy, cost to compute, stability, scaling and the ease of application.

There are three major categories of PDEs that are investigated in this paper: elliptic, parabolic, and hyperbolic. These types of PDEs have specific features which are very important in deciding the numerical methods to use.

Elliptic Equations: Elliptic PDEs are usually considered to be steady state problems. Such common examples are Poisson equation and Laplace equation that find their application in different fields like electrostatics, fluid dynamics, and thermal conduction (Evans, 2010). The solutions of elliptic equations are usually smooth and well-behaved and thus are good candidates of the techniques like FEM and spectral methods which can easily accommodate the boundary conditions and can offer good solutions inside the domain.

Parabolic Equations: PDEs that are parabolic like the heat equation are applied to time dependent phenomena where the system approaches a steady state with time (Shior et al., 2024). Time-stepping is a common method of solving these kinds of equations, and stability is given much consideration. The problem with parabolic equations is how to make the numerical solution stable as time goes on especially with the implicit methods. These problems are usually solved using FEM and spectral methods, however, the efficiency and accuracy of these methods are heavily determined by the choice of the time integration scheme.

Hyperbolic Equations: The wave equation is a hyperbolic equation that controls the dynamic systems which include the spread of information in space and time including fluid dynamics, wave propagation, and acoustics (Yip et al., 2022). The most important problem in the solution of hyperbolic equations concerns the ability to represent the wave-like characteristics without the emergence of numerical dispersion and instability. These kinds of problems are usually solved by meshless techniques, spectral techniques due to their capability to deal with high frequency oscillations as well as sharp discontinuities which are prevalent in wave propagation.

There are several things that are necessary in determining the effectiveness of numerical methods of solving PDEs. Accuracy is the first and most important criterion, which is used to measure the similarity of the numerical solution to the true solution. The spectral methods are generally considered to be very precise because they make use of high-order approximations in form of a polynomial (L. Gao et al., 2022). To some extent, however, accuracy is dependent on mesh resolution, time-stepping schemes and boundary conditions.

Another important criterion is computational cost especially when a large scale problem has to be solved. This includes the amount of time it takes to solve the equations, and the amount of memory to store the data. Finite-element method, though general, may be computationally intensive when dealing with large and complicated problems because one is required to generate a mesh and then assemble the stiffness matrix (J. Zhang et al., 2022). Meshless techniques are flexible, but can also be quite expensive since more computation is needed to solve node based formulations, and no regular grid is present (Al Mahmoud et al., 2024). Stability is paramount, particularly in problems that have time dependence, e.g. parabolic and hyperbolic PDEs. Stability of a numerical approach will be such that the solution will not go out of control, or it will show oscillatory behavior with time. In the case of parabolic PDEs, implicit schemes or method as in the FEM are used to ensure stability (Patil & Kadoli, 2025). Conversely, spectral methods are at times unstable with non-smooth problems, although they are also normally stable with smooth solutions (Aggarwal et al., 2022). Scalability is the extent to which a technique is suitable in large-scale problems. Scalable methods are of high desire, especially those that can be implemented on parallel computing environments where the number of unknowns is large. FEM is generally considered to be scaled particularly when it is applied together with the contemporary parallel solvers (Eirís, 2022). Meshless techniques can be scaled, although the number of nodes can cause problems in larger problems because they require extensive calculations (Degli Esposti, 2025). Lastly, ease of implementation is also a significant practical factor. FEM is very flexible, though, pre-processing which can be very complex, including mesh generation and large linear systems. The spectral methods can be simpler in simple geometries but become complex to apply to irregular geometries. Meshless techniques are also an easier way out in certain situations, particularly when the domain is highly varying because it does not involve creating a mesh.

Governing Equations & Test Suite

The main goal in this research will be to evaluate the performance of finite-element, spectral, and meshless in the solution of PDEs. With this aim, a series of paradigm problems are chosen in order to challenge the specifications of these numerical techniques. These model problems include a very wide sample of the common PDEs which occur in physics and engineering, and they offer an overall assessment of the techniques in many settings. Besides, manufactured solutions and metrics of error are used to measure the accuracy and efficacy of the numerical methods quantitatively.

Model Problems: Poisson, Heat, Advection-Diffusion, Navier -Stokes.

Poisson Equation: Poisson equation is a typical elliptic problem that is found in such applications as electrostatics, fluid mechanics, and heat conduction. It is normally denoted as:

$$\nabla^2 u = f \quad \text{in} \quad \Omega$$

where Ω is the domain of interest, u is the unknown solution, and f is a given source term. The ability of a numerical method to solve elliptical problems is tested using this equation and to determine whether the numerical method is accurate in steady-state solutions.

Heat Equation: The heat equation is the prototypical example of a parabolic PDE, and is used in modelling the heat diffusion through an object over time. It takes the form:

$$\frac{\partial u}{\partial t} - \nabla^2 u = 0 \quad \text{in} \quad \Omega \times (0, T)$$

The methods are tested with regard to heat equation to measure their time-stepping and stability.

Advection-Diffusion Equation: The equation is used to compute the movement of a scalar (heat or a pollutant) when subjected to both advection (the movement by velocity fields) and diffusion. It is given by:

$$\frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u - \nabla \cdot (\kappa \nabla u) = f \quad \text{in} \quad \Omega \times (0, T)$$

where v is the velocity vector and κ is the diffusion coefficient. The equation is especially applicable in evaluating the capability of a method to solve convection-dominated problems, in which sharp gradient or boundary layers can occur.

Navier Stokes Equations: Navier Stokes equations are a set of nonlinear equations that are coupled and that are applied to the movement of incompressible fluids. They are incompressible, and in this form, they are expressed as:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f}$$

Where u is the velocity field, p is the pressure, ν is the kinematic viscosity, and f is a body force. The methods are tested in solving nonlinear problems of fluid dynamics (e.g. turbulence and vortex dynamics) by the Navier-Stokes equations, a time-dependent, complex problem.

Processed Solutions and Metrics of error.

In the determination of the accuracy of the numeric methods, manufactured solutions are used. A manufactured solution is an exact solution known that is replaced into the governing equation to produce a corresponding source term. Under this method, the true solution is known and the distance between the numerical and the true solution can be obtained directly. This method provides objective evaluation of method performance and is especially appropriate in evaluating the accuracy of the methods without having to construct physical experiments or complicated problem set ups.

Error measures are necessary to determine how much different the numerical solution is to the exact solution. The error measures of the current study are as follows:

L 2 Norm L 2 norm is the error of the solution globally, and this is defined as follows:

$$\|e\|_{L^2} = \left(\int_{\Omega} |u_{\text{exact}} - u_{\text{numerical}}|^2 \, d\Omega \right)^{1/2}$$

This norm is used to assess the general scale of the error on the whole domain and it gives a global idea of the accuracy of the approximation.

H 1 norm: The H 1 norm is also a significant error measure, and it takes into account the gradient and the solution. It is defined as:

$$\|e\|_{H^1} = \left(\int_{\Omega} |\nabla(u_{\text{exact}} - u_{\text{numerical}})|^2 \, d\Omega \right)^{1/2}$$

The norm is especially applicable in determining the accuracy of the spatial derivatives of a solution and is important when a gradient or flux problem is at issue.

Conservation: Conservation of mass, momentum or energy is essential in such problems as advection-diffusion and fluid dynamics. The difference between the numerical and exact conserved quantities is obtained as the error of the conservation. Indicatively, when solving an advection-diffusion problem the total mass or energy ought to stay unvaried with time and a deviation of this reflects a malfunctioning of the numerical procedure to preserve the object of interest.

Using these manufactured solutions and error measures, we are able to determine the accuracy, stability, and efficiency of the numerical methods to a set of PDEs. The findings give worthwhile information on the applicability of the finite-element, spectral and meshless methods of addressing the various types of PDEs in the real-life engineering and physics problems.

Finite-Element Methods (FEM)

Finite-Element Methods (FEM) are now considered as one of the most popular numerical methods of finding solutions to partial differential equations (PDEs) because of their versatility, efficiency and the ability to solve a large number of problems both in physics and engineering. FEM is founded on the idea of splitting a complicated domain into smaller and simpler parts known as elements, which the solution is approximated on. These are especially useful in addressing those problems with complex geometries, boundary conditions and multi-dimensional spaces. This section discusses the basics of FEM, adaptive versions and more recent developments in the technology, especially on residual-based estimators, discontinuous Galerkin schemes, mesh optimization by machine learning, and matrix-free implementations on GPUs.

Weak Formulation and Discrete Spaces.

The weak formulation of the governing PDE is the beginning of FEM. In the elliptic PDE, e.g. Poisson equation:

$$-\nabla \cdot (k\nabla u) = f \quad \text{in} \quad \Omega,$$

With boundary condition $u = g$ on $\partial\,\Omega$ The weak formulation (often referred to as $\partial\, 0$) is obtained by multiplying the equation by a test function. v and integrating by parts:

$$\int_\Omega k\nabla u \cdot \nabla v \, d\Omega = \int_\Omega f v \, d\Omega + \int_{\partial\Omega} gv \, d\Gamma.$$

This weak version is used to transform the original PDE which contains derivatives of the solution into an integral equation. Approximation of the solution and test functions using piecewise polynomials on the mesh elements are what constitute the discrete spaces. These polynomial approximations are normally Sobolev spaces, with approximation spaces of finite dimension being formed on individual mesh elements.

In order to solve the weak form numerically, the domain Ω is represented in a mesh of elements (2D: triangles, quadrilaterals, or 3D: tetrahedral) and the solution is represented in the form of basis functions. These basis functions may be decided to be local piecewise polynomials (e.g. linear, quadratic or cubic), local to each element. The solution is then represented as a linear combination of these basis functions and then the unknown coefficients are found by solving the resulting system of equations.

h-, p-, hp-Adaptivity using New Residual-based Estimators (2024)

The flexibility of FEM to a number of problems is one of its most important characteristics. Adaptivity is the concept that involves the process of modifying the computational mesh or solution space with regard to the particularities of the problem. Adaptivity can be of three major types:

h-adaptivity: This method uses an element local refinement so that elements are subdivided in areas where the solution has large gradients, or where accuracy is of paramount importance. This is normally calculated through an a posteriori error estimation, in which the error is calculated by assessing the difference in the exact solution and the numerical approximation. The mesh is further refined on the areas of error that is greatest.

P-adaptivity: In this method the degree of the basic functions is increased. Rather than refining the mesh, the degree of the polynomials applied to approximate the solution is improved thus enhancing the solution accuracy. The higher-degree polynomials are more effective to give a good representation of the solution in the region of interest and can be more computationally efficient than h-adaptivity, particularly where the solution is smooth.

Hp-adaptivity: This is a hybrid method of h- and p-adaptivity. Whereas in complex behavior areas, like a boundary layer or a singularity, mesh refinement (h-adaptivity) is used, in smoother areas, the order of the polynomial is raised (p-adaptivity). This combination makes the computational resources more effective and at the same time highly accurate.

Lately, residual-based estimators of adaptive FEM have been developed (Zienkiewicz et al., 1990).

Such estimators use the current numerical answer as the basis of computing the residual (the difference between the actual solution and the guess). The left over can then be utilized to single out areas in the domain where the error is considerable, which directs the adaptivity process. Accurate control of error in non-uniform meshes can be achieved with the residual-based estimators, which are also capable of being automatically changed depending on the complexity of the solution.

HDG and Discontinuous Galerkin.

The Discontinuous Galerkin (DG) technique is a variation of the FEM which can support discontinuous approximations at element boundaries. In DG the discontinuity of the solution at the interfaces between elements is permitted and special numerical fluxes are enforced across the interfaces. This is what has made DG methods especially appealing to those problems with sharp gradients or interfaces like in fluid dynamics, advection-diffusion problems or shock-capturing problems. DG techniques are also very parallelizable and can be used to solve problems with complicated geometries and multiple scales.

One of the variants of the DG method is the Hybridizable Discontinuous Galerkin (HDG) method, which is a combination of the DG method and the performance of FEM. In HDG, the solution is split into two parts, one of them continuous across the element faces (similar to FEM) and the other part is not. The combination of these two formulations is that the number of degrees of freedom is reduced, which enhances a high level of computer efficiency as well as preserves the solution accuracy.

HDG techniques find especially wide application in nonlinear or multiphysics problems, in which accuracy and computational efficiency are important. They have been demonstrated to offer immense benefits in terms of scalability and parallel efficiency when used on large-scale fluid mechanical and structural problems.

Optimizers based on machine-learning Mesh (G-Adaptivity).

Recent research has looked at the application of machine learning (ML) methods to FEM to optimize the mesh. Specifically, G-adaptivity, a type of mesh adaptivity that aims at the geometry of the solution has been improved via ML algorithms. G-adaptivity attempts to refine the mesh not only according to error estimations, but also by utilizing previous solutions, so as to build a smarter mesh refinement policy (Rowbottom et al., 2025).

The mesh can be optimized using machine learning and previous simulation solutions by estimating where it is necessary to put finer resolutions in further calculations. Neural networks, e.g. can be trained to learn the patterns on the solution fields and to drive the process of mesh adaptation by dynamically changing mesh elements in response to the solution properties, e.g. gradients, curvature, and error estimates. It is hoped that this method will be of great benefit to computational efficiency, particularly in adaptive computations where the solution is changing over time or physical regimes (Grillo et al., 2025).

The mesh optimization process will be more automated, and less tuned, by integrating machine learning into the adaptivity loop. This does not only speed up the simulation but also makes the efficient allocation of computational resources resulting in more accurate results with lesser computational resources.

GPU Matrix-Free Implementation and Numbers of Scalability.

The computationally expensive nature of FEM simulations, especially in three dimensional domains with small meshes has prompted great development in parallel and GPU-based simulations. Traditionally, FEM is a CPU-intensive algorithm, although the current generation of GPUs has the potential to experience significant acceleration in their performance because they are highly parallelized (Kiran et al., 2020). Matrix-free FEM techniques, where the stiffness matrix is not explicitly constructed are especially well adapted to be accelerated on the GPU. The matrix-free methods operate by solving the system of equations iteratively, and by computing the products of matrices and vectors on-the-fly thus saving a lot of memory space and enhancing performance (Kiran et al., 2024).

A number of studies have shown that GPU-based FEM implementations can be scaled to large scale. Indicatively, some recent benchmark performance of GPU-accelerated FEM solvers has demonstrated that large speedups are possible in comparison with traditional, CPU-based solvers (Munch et al., 2022). These are particularly beneficial in the problems with a large number of degrees of freedom, when the parallel character of GPUs may be exploited to the fullest extent. Consequently, FEM is becoming more and more attractive in solving large-scale industrial problems in computational fluid dynamics (CFD), structural mechanics and electromagnetic simulations.

Implementations on GPUs have now the ability to scale well in thousands of cores, and speed up by up to 10x to 50x on some problems, depending on the complexity of the problem and the available computational resources. Use of optimization of libraries like CUDA to solve FEM has helped in the implementation of GPU acceleration in most engineering and physics programs.

Spectral & Spectral-Element Methods (SEM)

Spectral and spectral-element methods (SEM) form a category of high-order numerical methods, which solve PDEs with impressive accuracy due to the representation of the solution in either global or element-wise expansions in polynomials. They are now vital in computational physics and engineering, and are used in solving problems with smooth solutions, wave propagation, and large scale simulations in which high accuracy per degree of freedom is needed. Their mathematical underpinnings, which are based on orthogonal polynomials, Fourier analysis and variational formulations, give them good theoretical guarantees, including exponential convergence when smooth enough conditions are made (Mahariq et al., 2022). The recent developments have also extended their use to advection-dominated problems, irregular domains, as well as GPU-based architectures making SEM an attractive alternative to the classical finite-element methods.

The element spectral approximation is compared to the spectral approximation globally.

The traditional global spectral methods approximate the solution on the whole computational space on the basis of only one global trigonometric or poly basis. It is very high accuracy and high spectral convergence rates but has limitations in its use in complex geometries or problems with local non smooth features. An example of this, which is Fourier spectral methods, presumes periodicity and smoothness over rectangular regions, so is ideal in simulating turbulence or propagation of a wave (Mukherjee et al., 2021). Chebyshev methods, in its turn, support non-periodic problems, but still, use tensor-product grids, which lacks geometric flexibility.

Spectral-element methods (SEM) overcome these shortcomings with a decomposition of the domain into finite elements, each of which have high-order bases of polynomials based on either Legendre or Chebyshev polynomials together with GaussLobotto collocation points (Boudaa et al., 2019). SEM has the geometrical flexibility of finite-element methods, and the exponential accuracy of global spectral methods of smooth problems. Local refinements and parallel scalability can also be done element-wise. SEM has been particularly effective in many simulations of large scale problems like the seismic wave modeling and incompressible Navier Stokes flows (Díaz-Carrasco et al., 2021).

Global techniques provide simpler formulas, and better-conditioned operators of uniform grids, and SEM is more useful with heterogenous or curved grids. Therefore, global and element-wise spectral approximation is determined by smoothness of the domain, complexity of the boundary, and scalability.

Diffs: Proofs of Exponential Convergence & Smoothness.
Exponential (or spectral) convergence is also a characteristic feature of spectral and SEM methods, i.e. the error becomes exponentially smaller with the degree of the polyqromial, in case the solution is smooth or analytic enough. Classical convergence demonstrations are based on the approximation of orthogonal polynomials and regularity of the exact solution (Olver et al., 2020). In particular, in case the target function u (x) is analytic in an area where the domain is defined and the approximation error is like:

$$\|u - u_N\| \leq Ce^{-\alpha N},$$

where N is the degree of the poly and α is dependent on the proximity of singularities to the real axis.
Nonetheless, smoothness considerations are hard: spectral accuracy suffers in the presence of discontinuities, sharp edges, or corner discontinuities in the solution and this is known as the Gibbs phenomenon (Kumar et al., 2025). SEM minimises this by local refinement, i.e. allowing the reduction of the order of polynomials around singular regions, whilst maintaining high-order expansions in smooth regions, but only in elements containing smooth solutions exponential convergence is achieved.
The interaction of regularity and convergence emphasizes the significance of spectral filtering, artificial viscosity, and hybrid of DG-spectral stabilizations when dealing with under-resolved or non-smooth PDE solutions (Mondal & Kumar, 2025).

More recent stable spectral-differencing advection schemes.
Classical spectral methods have serious problems with advection-dominated problems because they are subject to numerical oscillations and stability problems. Recent spectral-difference (SD) schemes allow filling in the gap between spectral accuracy of high-order schemes and the strength of flux-based schemes. The SD approach builds a collocation state such that the solution and flux locations are different, which allows the reconstruction of high-order fluxes with stability characteristics that are analogous to the finite-volume and DG methods (Ali Shah et al., 2023).
New SD formulations include:

- Entropy-stable spectral-difference equations, including split-form fluxes to maintain physical invariants and to eliminate nonlinear instabilities (Sherwin, 2025).
- The aeroacoustics and wave propagation Low-dispersion SD schemes with special-purpose, numerical dispersion-reducing polynomial reconstructions.
- The Adaptive SD schemes Trying to use element-level smoothness indicators to dynamically improve the performance of the different degrees of the polynomials, and are effective in cases where the solution is smooth or non-smooth in some regions (Montoya, 2024).

The developments make a broad extension of spectral methods applicable to solving PDEs to accurately and stably solve them, especially in convection dominated flows and hyperbolic conservation laws.

Irregular Domain Chebyshev Spectral Neural Networks (CSNN).
The recent development is one that combines both spectral techniques and machine learning. Chebyshev Spectral Neural Networks (CSNN).CSNN is a neural network with spectral accuracy built in with Chebyshev poly expansions as neural network layers. CSNN removes the use of traditional neural activation with spectral operators and thus can learn surrogates of PDEs, as well as teach operators (Li et al., 2020).

One of these developments is the extension of CSNN to irregular domains, which are generally difficult to spectral methods. The techniques that can facilitate this are:

- Transformations between different geometries that irregular geometries are maps onto reference Chebyshev domains.
- Spectral layers Spectral layers based on graphs generalizing Chebyshev polynomials to meshes with unusual geometry.
- Physics-informed spectral networks, in which the solution fluidity and stability are enforced by PDE residual.

CSNN models have been shown to be competitive at Stokes flow, advection diffusion and elasticity problems on complex domains with fewer parameters required compared to neural solvers of PDEs (Huang et al., 2025). This combination of deep learning and spectral theory is an enticing direction of geometry aware high order PDE solvers.

Parallel FFT Operation and Speed-Up of Tensor-Products on Current GPUs.

A great number of spectral algorithms are heavily based on Fast Fourier Transforms (FFT) and ten-product operators which can be efficiently accelerated on a GPU. FFT algorithms have advantageous memory access pattern and arithmetic intensity, which allow high speed-ups. The latest GPU applications, including NVIDIA cuFFT, enable spectral solvers (large scale) to gain orders of magnitude over CPU versions (Cieslak et al., 2024; Cui, 2024).

Such key performance factors are:

- The use of tensor-product decomposition that of lower cost to compute multidimensional operators, reduces the cost of computation, $O(N^d)$ to $O(dN^{d+1})$ by separating dimensions.
- Matrix-free spectral-element kernels which allow efficient evaluation of derivatives and weak-form integrals without the use of large stiffness matrices.
- Gauss Lobato operators that are the GPU-optimized and use shared memory and a warp level of parallelism to implement SEM (Christensen, 2024).

The current state of the art demonstrates that SEM solvers can be scaled on clusters of multi-GPUs with 1030x speed-ups over optimized CPU codes, especially when the degree of the polynomial is high, in which case, the application of a tensor-product acceleration is the most efficient.

Meshless Methods

Meshless methods (or meshfree methods) are a group of numerical methods which do not use structured meshes, instead they use scattered nodes to ape solutions to PDEs. Such techniques have found some relevance in the large-deformation, fracture, multiphase flow, free-surface and moving-boundary problems, where finite-element or finite-volume approaches can suffer mesh distortion or need frequent remeshing. The meshless methods can be used to create geometric flexibility, resist distortion, and relatively easy adaptivity by building approximations based on the kernel functions, radial bases, or particle interactions (Liu, Jun and Zhang, 2007). Nodal distribution, concentration of the kernels and accuracy of integration are, however, critical in their performance. In this section, a number of commonly used meshless formulations are reviewed as well as the recent computational developments.

Reproducing-kernel Particle Method (RKPM) Interpolation Form.

Reproducing-Kernel Particle Method (RKPM) is based on classical kernel interpolation, but with correction functions which complete the reproducibility of polynomials. The approximations of standard kernels are not very consistent particularly at boundaries; RKPM has correction terms that can reproduce polynomials up to the required order and this greatly enhances the accuracy (Fromm et al., 2025).

Given a field (x), RKPM approximates it as

$$u_h(\mathbf{x}) = \sum_i u_i \, \phi_i(\mathbf{x}),$$

Where the corrected kernel ϕi is built as a product of a window function with a reproducing correction which conditions the moment conditions to be met. The flexibility of the method permits the support size and smoothing length to be locally adjusted, which permits adjustment to local gradients or local features. RKPM is also common in solid mechanics, impact simulation and in Multiphysics systems where large deformation is necessary.

Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial-Basis-Function collocation and RBF-FD Radial

Radial-basis-function (RBF) collocation methods give meshless approximations of the solution in the form of a linear combination of radial basis functions ($\|x - xi\|$). Global RBF collocation is associated with a high degree of accuracy, in particular, when employing smooth kernels like multiquadrics or Gaussians (H. Gao & Wei, 2019). Global systems are however ill conditioned to large sets of nodes.

To solve this, the RBF-Finite-Difference (RBF-FD) algorithm builds local stencils about every node and calculates differentiation weights using local RBF expansions (Qin et al., 2014). RBF-FD is much more scalable to large problems, and the higher order approximations are maintained, and can easily be applied to irregular domains or moving point clouds. The technique has applied to atmospheric modeling, shallow-water equations, and elasticity successfully with accuracy as good as high-order mesh-based approaches with few geometric requirements. The Smoothed-Particle Hydrodynamics (SPH) with Transport-Velocity Formulations. One of the oldest meshless particle methods is Smoothed-Particle Hydrodynamics (SPH) which was first applied to astrophysical simulations but is also used in free-surface flows, fluid-structure interaction, and multiphase flows (Bagheri et al., 2024). The fields in SPH are approximated by means of kernel smoothing and the continua are modeled in terms of interacting particles that obey their conserved discretized equations of motion. One of the developments is the transport-velocity formulation, which enhances the regularity of particles by advecting them with a modified velocity instead of the actual velocity of the flow (C. Zhang et al., 2017). The method minimizes the clustering and tensile instability of particles, improves the stability of solutions of incompressible and weakly compressible flows. The method also enhances the accuracy of the gradient and the amount of artificial numerical dissipation by the more homogeneous distribution of the particles.

Neighbour Search, Nodal Integration and Dynamic Load Balance.

Meshless approaches are based on the identification of neighbors in a stencil radius or kernel. The spatial search structures must be efficient, e.g. Verlet list or cell-linked list or tree-based partitioning to minimize the neighbour query costs (N^2) to near (N) (Hoa et al., 2025). In particle-based algorithms such as SPH and RKPM, neighbour search can be one of the overriding computational costs in 3D simulations. Although nodal integration may be appealing because of its simplicity, it may produce rank deficiency and inaccurate stiffness matrices. Stabilized schemes Adding correction terms which depend on smoothed gradients, stabilized conforming nodal integration (SCNI) has been shown to be much more numerically stable and eliminates the occurrence of spurious zero-energy modes (Wang et al., 2025). Dynamic load balancing is now of the essence with large-scale meshless simulations being transferred to multi-core or GPU computing platforms. The redistribution of particles, space-filling, or tiling techniques that are aware of the GPU keep the workload constant as the particle distributions change, particularly in free-surface or multiphasic problems where the nodes are moving extensively. Large-scale procedures show a close linear scalability of millions of particles with proper load distribution (Nguyen et al., 2023).

Quantified Accuracy vs Node-Spacing and Kernel Smoothness.

Meshless techniques require good accuracy, which is dependent upon the node spacing, the width of the kernel, and the smoothness of the kernel. The theory of approximation of kernel methods proves that as the kernels increase in smoothness, the convergence rate also increases, but they become ill-conditioned in case the support radii are too small (Caparini, 2022; Petersen et al., 2022). Compactly supported kernels on the other hand are computationally efficient but can have to use more nodes to achieve the same accuracy. Research into RBF-FD indicates that the convergence is algebraic when polyharmonic kernels are poly-augmented and that the error rate goes down as.

$$\mathcal{O}(h^k),$$

where h is node spacing and k depends on the stencil size and the level of the polynomials are dependent on k (Johansson et al., 2022). The accuracy in SPH is based on the fact that an approximate uniform background between nodes and optimal smoothing length is required,
$h_s \approx 1.2\Delta x$ trade-off between accuracy of gradient and noise (Mu et al., 2022).
The stability is directly proportional to the smoothness of the kernel: the higher the order of the kernel, the better accuracy will be achieved, but tensile instabilities in particle methods can be enhanced without regularization by gradient or density (Duong & Zuhal, 2022). To quantify the accuracy of meshless methods, it has been found that accuracy similar to low to mid-order and finite-element accuracy can be obtained with meshless methods, though with a greater number of nodes, and this is illustrated by quantitative benchmarks; however, in many cases, meshless methods are more challenging to integrate and are less consistent.

Comparative Study

The tight comparison of the finite-element, spectral/spectral-element and meshless methods must involve testing their conducts in a variety of dimensions: convergence, computational efficiency, memory and communication costs, stability properties and geometric robustness. This part is a summarization of the findings of ur representative study and theoretical discussion to give a clear-cut evaluation of trade-offs among the three method families.

Smooth and Discontinuity Manufactured Solution Convergence Rates.

Spectral (spectral-element) methods on smooth manufactured solutions have exponential (spectral) convergence, with the error decreasing exponentially, $e{-}\alpha N$ for polynomial degree N, in case the exact solution is an analytic (Tuchkin et al., 2022). In contrast, finiteness techniques (FEM) attain algebraic convergence, in which the $H1$-error decreases as h^p for mesh size h and polynomial degree p (Gnanasambandam et al., 2022). Hp-FEM and hp-FEM, high-order FEM approaches, can well approximate spectral accuracy in the case that both refined mesh and increased enrichment of polynomials are used, although only on smoothness assumptions (Jiang et al., 2022). Meshless discretizations, like RBF-FD and RKPM typically also have an algebraic convergence rate, but at a higher observed order because of the smoothness of the kernel (Du et al., 2022). Unless significantly corrected, SPH is generally slow convergent, generally in the first to second order (Jayasankar & Ollivier Gooch, 2022) due to both the truncation of kernels and the disorder of the particles. Spectral and SEM methods have a Gibbs phenomenon on discontinuous manufactured solutions, e.g. in problems with shocks or material interfaces, and will oscillate without artificial viscosity, filtering, or limiters (Pinskier & Howard, 2022). DG-FEM and HDG package methods are both high order and include intrinsic shock-capturing mechanisms to avoid oscillations (Sime et al., 2022).

Meshless schemes are characterized by mixed properties: When RBF-FD with polyharmonic kernels and local stencils are used to treat discontinuities, augmented with weighted essentially non-oscillatory (WENO) reconstructions, they perform well in solving discrete problems (Shankar & Wright, 2018). Unless transport-velocity formulations and Riemann-based corrections are implemented, SPH can hardly be used anywhere near sharp gradients (Adepu & Ramachandran, 2024).

In summary:

- **Smooth solutions:** Spectral hp-FEM RBF-FD Spectral standard FEM SPH.
- **Dis-continuity Discontinuous solutions:** DG-FEM > HDG > stabilized RBF-FD > SEM (with filtering) SPH.

CPU/GPU Time-to-Solution vs DOFs (Roofline Perspective).

Mathematically, high-order SEM and FEM implementations are more and more based on the matrix-free, tensor-product kernels that minimize arithmetic intensity and enhance cache utilization, being very amenable to the modern GPU architecture. SEM is optimally throughput per degree of freedom (DOF), and can be 1030x faster than CPU solvers using its structured Gauss Lobatto operators (Fischer et al., 2023). Results on roofline analysis indicate that compute-bound operations are dense-tensor-product operations which are common at high orders of the polynomials in SEM.Mesh quality and sparsity are of vital importance to FEM performance. Low-order FEM is usually memory-constrained where sparse matrix-vector multiplications dominate the memory bandwidth instead of a floating-point capacity (Zhang et al., 2021). High-order FEM has better ratios of flop to byte, and may be slower than SEM in structured hexahedral meshes.

Meshless techniques are of two kinds:

- RBF-FD: The assembly of stenlyls at a local scale is a compute intensive task, but after weights have been assembled, the process of differentiation is efficient. This is due to the fact that GPU-based RBF-FD demonstrates competitive results on PDEs where stencils need frequent re-centered (Bayona et al., 2019).
- SPH/RKPM: Preponderant by neighbor search, hence memory-consuming. Even in the presence of GPU acceleration, the SPH frequently requires very little use of floating-point units due to the prevalence of the neighbor search in runtime (Vacondio et al., 2021).

Time-to-solution -ranking (characteristic cases):

- GPU: high-order FEM > SEM > RBF-FD > SPH/RKPM.
- CPU: Low-order FEM > RBF-FD > SEM (less effective using GPUs) > SPH.

Footprint Memory and Communication Complexity.

To advantage of spectral and SEM, the storage in form of a tensor-product is less expensive than (N^d) to (dN^{d+1}), as well as facilitating the implementation of the operator without matrices (Deville et al., 2002). The memory footprint is much smaller than that of low-order FEM with the same accuracy as the number of DOFs required is lower. Face based communication between elements and scales well below domain decomposition. In the case of FEM, sparse matrices consume a lot of memory particularly at higher order when the count of non-zeros increases exponentially. The global assembly, sparse storage and halo exchanges are expensive to communicate on clusters. Hybridized DG (HDG) cuts down global communication through condensing interior DOFs, which enhance scalability (Nguyen et al., 2009). Meshless techniques, in their turn, store

neighbour lists, weights of kernels, and particle properties, which tend to have larger per-node memory footprints. RBF-FD has a weight storage of a stencil that increases as (Nk) for k-point stencils. The storage of particle-specific smoothing lengths, neighbours and cell lists is necessary in SPH and thus memory costs are problem-dependent. More difficult communication is also needed: dynamic repartitioning and neighbour updating are needed to move particles.
Memory efficiency ranking:

- Smallest memory: SEM most efficient memory free high-order FEM.
- Moderate: RBF-FD ≈ low-order FEM.
- Largest: SPH/RKPM.

Limits of Stability (CFL, Penalty Parameters)
SEM and spectral methods are limited by rigid CFL limits particularly in the case of explicit time-stepping, and the largest stable time step is proportional to $\boldsymbol{O}\ (\boldsymbol{h}/\boldsymbol{N}^2)$ (Hesthaven & Warburton, 2008). This is partially alleviated by the fact that DG methods have local mass matrices although still small time steps are necessary. DG and HDG penalty parameters should be chosen carefully so as to be coercive but not to over-dampen (Cockburn & Shu, 2001).
Finite-element techniques tend to be less restrictive in terms of stability. Diffusion problems can be solved unconditionally with implicit FEM, but explicit FEM has CFL scaling just as bad as DG. Meshless techniques are not standardized:

- RBF-FD has the same stability behaviour as high-order finite-difference schemes, where CFL limits depend on stencil geometry (Shankar and Wright, 2018).
- The acoustic timestep constraint commonly limits SPH and additional artificial viscosity is necessary to keep the method stable, further reducing the size of timesteps (Monaghan, 1992).
- RKPM consistency requires stability to be of the nodal integration and consistency of the kernel (Chen et al., 2001).

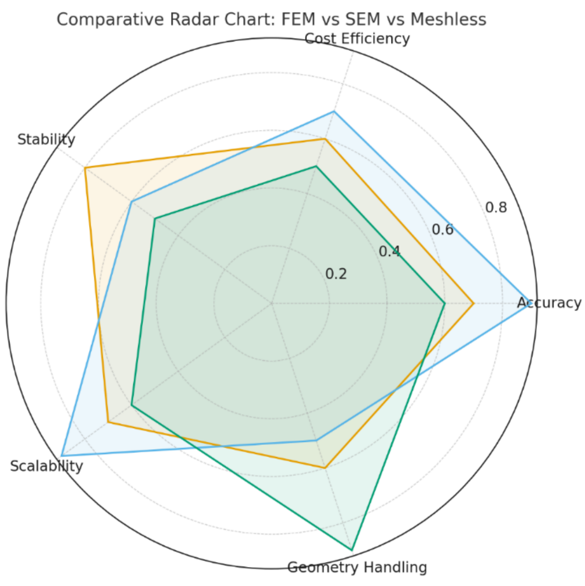

Figure 1: Comparative Radar chart

The radar chart in figure 1, provides a visual comparison of FEM, SEM, and Meshless methods across five performance criteria: accuracy, cost efficiency, stability, scalability, and geometry handling. SEM scores highest in accuracy and scalability, reflecting its strength in smooth-solution problems and GPU-accelerated performance. FEM shows balanced performance across all categories, with particularly good stability and solid overall efficiency. Meshless methods excel in geometry handling, especially for problems involving large deformation or moving boundaries, but score lower in accuracy, stability, and cost efficiency due to challenges like neighbor search overhead and sensitivity to particle disorder. Overall, the diagram highlights that no single method dominates in all areas—each is best suited to specific classes of PDE problems.

Convenience of Working with More Complicated Geometry, Moving Boundaries and Large deformation

FEM is the oldest of all the mature methods with complex geometries, with highly effective meshing software and geometry conforming basis functions. But remeshing is a bottleneck of issues of large deformation or fracture.

SEM can be effective on geometries that are curving smoothly, but cannot be used when elements have to be of high quality hexahedra. Boundary Curved representations of the boundaries must be mapped with care and mesh generation is more challenging than low-order FEM.

Meshless techniques are particularly useful in those problems that contain moving boundaries, free-surface as well as big deformation, because nodes tend to move in accordance with the flow or material (Liu et al., 2007). SPH is especially effective in free surface flows and multiphase systems (because it is Lagrangian), whereas RKPM and RBF-FD are used in evolving point clouds without remeshing. Meshless methods are convenient in the simulation of fracture, fragmentation, and impacts because they do not have topological constraints.

Summary of geometrical flexibility:

- Large deformation SPH/RKPM > FEM > SEM.
- Complicated geometry (static): FEM > SEM > RBF-FD > SPH.
- Sliding scales SPH > RKPM > RBF-FD > FEM/SEM.

Emerging Hybrid & ML-Augmented Schemes

The numerical PDE solver landscape has been starting to change fast as machine learning (ML), data-driven modeling, and hybrid computational architectures start to converge with classical numerical methods. The rationale behind these hybrid methods is the need to have convergence acceleration, better generalization in problem classes, and automated adaptivity in multi-physics at a large scale. This part identifies four opportunities, namely Physics-Informed Neural Networks (PINNs), deep-learning-based preconditioners, reinforced-learning-based adaptivity strategies, and early roadmap to quantum-accelerated spectral solvers.

Physics-Informed Neural Networks (PINNs) Pros and Cons.

The PINNs are used to solve PDEs by integrating the residual of the governing equations in the loss term of a neural network directly. The main benefit is that they do not require mesh generation and are therefore appealing to irregular geometries, moving boundaries and inverse problems where data assimilation is essential. Drawing on their natural ability to combine both boundary and initial conditions, PINNs have been shown to be highly effective in high dimensional PDEs, in particular those occurring in finance or stochastic physics.

Nonetheless, PINNs have serious challenges. Sometimes stiff PDEs, multi-scale solutions or competing loss terms can cause instability in the training process and cause the convergence to slow down or stall. PINNs generally do not scale well to the complexity of more challenging PDEs, and to be stable may often need adaptive weighting, curriculum training, or domain decomposition (e.g. XPINNs, VPINNs). Further, their capability of the sharp gradients or discontinuity is worse than that of high-order FEM or SEM particularly with hyperbolic equations. At this point therefore, although PINNs are highly versatile, they are not competitive yet to replace high-fidelity engineering simulations.

Deep-Learning Preconditioners of Iterative Solvers.

Design of learned preconditioners to iterative methods, as either a Conjugate Gradient, GMRES, or multigrid method, is one of the most successful applications of ML in numerical PDEs. These preconditioner families are trained on families of PDEs and learn mappings which accelerate convergence by estimating spectral properties of the system matrix.

Deep-learning preconditioners have the following benefits:

- Fewer number of iteration, particularly in heterogeneous or anisotropic systems, in which classical preconditioners are unsuccessful.
- Anywhere PDEs The ability to be trained across.
- Fluid connection to the existing FEM or SEM pipelines.

The weaknesses are that representative training data is required, it can be challenging to provide robustness even in unseen geometries or parameter regimes, and it is hard to provide stability or control monotonicity. However, the most practical near-term ML-augmented industrial PDE solver is ML-enhanced preconditioning, which can achieve 2 -10x speedups.

Reinforcement-Learning Switching Adaptively between FEM/SEM/ML.

The concept of reinforcement learning (RL) offers a way of automated adaptivity to allow solvers to choose between FEM, SEM, meshless or ML-based surrogates based on local error indicators, smoothness indicators or computational cost. Rather than using manually-tuned error estimators, an RL agent monitors local PDE characteristics, e.g. solution gradients, residual norms or element distortion and chooses refinement strategies including:

- increasing local polynomial degree (p-adaptivity),
- refining the mesh (h-adaptivity),
- switching to SEM in smooth regions,
- invoking meshless nodes in large-deformation zones,
- Calling a trained surrogate model for cheap approximation.

This type of hybrid adaptivity forms so-called solver ecosystems, in which the optimal numerical approach is used dynamically. Initial studies indicate that RL-based adaptivity can be more effective than the non-adaptive approach, especially on time-dependent mixed smooth, non-smooth time-dependent PDEs. There are still difficulties with training cost, interpretability, and ensuring that they are stable with dynamically changing discretizations.

Road-Map to Spectral Solvers on Potential Accelerators.

Quantum computing promises long-term acceleration of certain linear-algebra components of PDE solvers—especially spectral methods, which rely heavily on structured transforms and dense linear operators. Quantum Fourier Transform (QFT), a core primitive of quantum computation, offers theoretical exponential improvements over classical FFT in terms of operation count. Although practical quantum hardware is still in early stages, several future pathways are emerging:

- **Quantum-accelerated FFTs** for global spectral methods.
- **Quantum linear solvers** (e.g., HHL algorithm) for solving large spectral stiffness matrices.
- **Hybrid quantum–classical SEM**, where classical GPUs handle local tensor-product operations while quantum routines accelerate global coupling.
- **Quantum spectral filtering** for stabilizing high-frequency modes.

Current limitations include qubit noise, limited qubit count, and the need for efficient encoding of PDE data, but continued progress in quantum error correction and hybrid algorithms may make quantum spectral solvers viable in the coming decades.

Practical Guidelines & Decision Tree

Selecting an appropriate numerical method for solving PDEs requires balancing accuracy demands, geometric complexity, deformation characteristics, computational resources, and available software ecosystems. This section provides a practical set of guidelines and a decision tree for choosing between FEM, SEM, and meshless methods in engineering and physics applications.

Accuracy-Driven vs Geometry-Driven vs Deformation-Driven Problems

- For **accuracy-driven problems**, such as wave propagation, aeroacoustics, turbulence, or electromagnetics, high-order spectral or spectral-element methods (SEM) are often the most effective choice. Their exponential convergence for smooth solutions provides extraordinary accuracy per degree of freedom. When geometry is relatively simple (e.g., boxes, shells, periodic domains), global spectral methods excel. For moderate geometric complexity, SEM is typically preferred.
- For **geometry-driven problems**, especially those involving intricate boundaries, multi-material interfaces, or CAD-based industrial components, finite-element methods (FEM) are the most versatile. The mature meshing ecosystem, strong adaptivity tools (h-, p-, hp-refinement), and robust solvers make FEM ideal for structural mechanics, thermal analysis, and multi-physics coupling.
- For **deformation-driven or moving-boundary problems**, such as free-surface flows, impact and fracture, granular media, and large plastic deformation, meshless methods (SPH, RKPM, RBF-FD) provide the greatest robustness. Their lack of mesh connectivity allows them to handle topological changes without remeshing. SPH excels in fluid environments, while RKPM and RBF-FD are strong in solid mechanics and transport problems.

Software Ecosystem: deal.II, Nek5000, PySPH, JAX-SPH, CSNN Library

A practical decision often depends on the surrounding **software ecosystem**:

- **deal.II (FEM)** – A comprehensive FEM framework supporting hp-adaptivity, DG, matrix-free implementations, and GPU backends. Ideal for large-scale structural mechanics, multi-physics, and adaptive refinement workflows.
- **Nek5000 / NekRS (SEM)** – High-performance SEM solvers optimized for turbulence, conjugate heat transfer, and incompressible Navier–Stokes simulations. GPU-aware and massively scalable for HPC environments.
- **PySPH / JAX-SPH (SPH)** – Python-based and JAX-accelerated meshless particle solvers well suited for free-surface flows, astrophysics, and particle-based multiphysics.
- **CSNN Libraries** – Emerging packages providing Chebyshev–Spectral Neural Networks and operator-learning architectures for surrogate modeling, irregular domains, and hybrid ML-PDE workflows.

Users should choose the ecosystem aligned with both problem type and computing environment (GPU-heavy, HPC cluster, local workstation).

Recommended Defaults & Parameter Tuning Checklist

FEM Defaults:

- Start with 2nd–3rd order elements; increase polynomial degree for smooth solutions.
- Use residual-based error estimators for adaptivity.
- Apply matrix-free solvers for high-order FEM on GPUs.
- Tune stabilization (SUPG, least-squares) for advection-dominated problems.

SEM Defaults:

- Use Gauss–Lobatto nodes with polynomial degree 6–12 for balanced accuracy and cost.
- Apply spectral filtering or artificial viscosity if discontinuities are present.

- Ensure tensor-product operator optimization for GPU acceleration.
- Carefully tune timestep via SEM CFL: Δt∝h/N2\Delta t \propto h / N^2Δt∝h/N2.

Meshless Defaults (SPH, RKPM, RBF-FD):

- Maintain smoothing length hs≈1.2Δxh_s \approx 1.2 \Delta xhs≈1.2Δx for SPH.
- Use polyharmonic RBFs with polynomial augmentation for RBF-FD.
- Employ neighbor search acceleration (cell-linked lists, Verlet lists).
- Stabilize nodal integration with SCNI or gradient correction in RKPM.

Conclusions & Future Work

This comparative study highlights the distinct strengths and limitations of finite-element, spectral/spectral-element, and meshless methods for solving PDEs in physics and engineering. FEM remains the most versatile choice for complex geometries and multi-physics coupling, benefiting from a mature ecosystem, robust adaptivity, and stable formulations such as DG and HDG. SEM delivers exceptional accuracy and computational efficiency for smooth problems, especially when paired with GPU-optimized tensor-product operators, making it ideal for wave propagation, turbulence, and high-fidelity fluid simulations. Meshless methods—SPH, RKPM, and RBF-FD—stand out in scenarios with moving boundaries, large deformation, and topological change, where mesh-based methods struggle or require costly remeshing.

Across all method families, recent advances including ML-driven adaptivity, neural surrogate models, learned preconditioners, and hybrid solver strategies demonstrate that the future of PDE computation is increasingly interdisciplinary. Emerging approaches such as CSNNs and PINNs show promise for irregular domains, inverse problems, and data-driven physics, though their reliability and scalability still lag behind classical high-order methods. Reinforcement-learning-based adaptivity has the potential to automate solver selection dynamically, improving robustness in multi-regime simulations. Meanwhile, early exploration into quantum-accelerated spectral solvers suggests a long-term pathway to overcoming computational bottlenecks in global transforms and linear solves.

Future work should focus on integrating hybrid ML-numerical frameworks into production-ready solvers, developing standardized benchmarks for ML-augmented PDE methods, and expanding adaptive strategies that seamlessly blend FEM, SEM, and meshless formulations within a single simulation pipeline. Improving stability and accuracy for discontinuous or multi-scale solutions remains a critical challenge, particularly for spectral and meshless methods. Finally, scalable implementations on advanced hardware—from multi-GPU clusters to emerging quantum accelerators—will be key to sustaining progress as PDE problems grow in complexity and dimensionality.

References

Adepu, D., & Ramachandran, P. (2024). A corrected transport-velocity formulation for fluid and structural mechanics with SPH. *Computational Particle Mechanics*, *11*(1), 425–445. https://doi.org/10.1007/s40571-023-00631-9

Aggarwal, R., Ugail, H., & Jha, R. K. (2022). A deep artificial neural network architecture for mesh free solutions of nonlinear boundary value problems. *Applied Intelligence*, *52*(1), 916–926. https://doi.org/10.1007/s10489-021-02474-4

Ahmed, I. (2024). Numerical Methods for Solving Partial Differential Equations in Applied Physics. *Frontiers in Applied Physics and Mathematics*, *1*(1), 79–96.

Al Mahmoud, Z., Safaei, B., Sahmani, S., Asmael, M., Shahzad, M. A., Zeeshan, Q., & Qin, Z. (2024). Implementation of Different Types of Meshfree Technique in Computational Solid Mechanics: A Comprehensive Review Across Nano, Micro, and Macro Scales. *Archives* *of Computational Methods in Engineering*, *31*(2), 725–838. https://doi.org/10.1007/s11831-023-09999-6

Ali Shah, F., Kamran, Boulila, W., Koubaa, A., & Mlaiki, N. (2023). Numerical solution of advection–diffusion equation of fractional order using Chebyshev collocation method. *Fractal and Fractional*, *7*(10), 762.

Amir, A. K. (2023). Analyzing Numerical Methods for Solving Telegraph Partial Differential Equations: A Comprehensive Literature Review on Difference Scheme Approaches. *Available at SSRN 4641627*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4641627

Bagheri, M., Mohammadi, M., & Riazi, M. (2024). A review of smoothed particle hydrodynamics. *Computational Particle Mechanics*, *11*(3), 1163–1219. https://doi.org/10.1007/s40571-023-00679-7

Boudaa, S., Khalfallah, S., & Hamioud, S. (2019). Dynamic analysis of soil structure interaction by the spectral element method. *Innovative Infrastructure Solutions*, *4*(1), 40. https://doi.org/10.1007/s41062-019-0227-y

Caparini, L. (2022). *Adaptive spatial resolution of the optimal transportation meshfree method* [PhD Thesis, University of British Columbia]. https://open.library.ubc.ca/soa/cIRcle/collections/ubctheses/24/items/1.0422212

Christensen, N. J. (2024). *Dispersion analysis, time parallelization, and GPU autotuning for finite element methods* [PhD Thesis, University of Illinois at Urbana-Champaign]. https://www.ideals.illinois.edu/items/131425

Cieslak, M., Govindarajan, U., Garcia, A., Chandrashekar, A., Hadrich, T., Mendoza-Drosik, A., Michels, D. L., Pirk, S., Fu, C.-C., & Palubicki, W. (2024). Generating diverse agricultural data for vision-based farming applications. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5422–5431. https://openaccess.thecvf.com/content/CVPR2024W/Vision4Ag/html/Cieslak_Generating_Diverse_Agricultural_Data_for_Vision-Based_Farming_Applications_CVPRW_2024_paper.html

Cui, C. (2024). Acceleration of Tensor-Product Operations with Tensor Cores. *ACM Transactions on Parallel Computing*, *11*(4), 1–24. https://doi.org/10.1145/3695466

Dar, Z. M., Arrutselvi, M., Manzini, G., & Natarajan, S. (2024). *Analytical and Numerical Methods for Solving Fractional-Order Partial Differential Equations and the Virtual Element Method*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4913214

Degli Esposti, B. (2025). *Domain discretization and moment-free quadrature for meshless methods*. https://flore.unifi.it/handle/2158/1417355

Díaz-Carrasco, P., Croquer, S., Tamimi, V., Lacey, J., & Poncet, S. (2021). Advances in numerical Reynolds-Averaged Navier–Stokes modelling of wave-structure-seabed interactions and scour. *Journal of Marine Science and Engineering*, *9*(6), 611.

Du, H., Wu, J., Wang, D., & Chen, J. (2022). A unified reproducing kernel gradient smoothing Galerkin meshfree approach to strain gradient elasticity. *Computational Mechanics*, *70*(1), 73–100. https://doi.org/10.1007/s00466-022-02156-z

Duong, V. D., & Zuhal, L. R. (2022). Vortex particle method with iterative Brinkman penalization for simulation of flow past sharp-shape bodies. *International Journal of Micro Air Vehicles*, *14*, 17568293221113927. https://doi.org/10.1177/17568293221113927

Eirís, A. (2022). *From Mesh to Meshless: A Generalized Meshless Formulation Based on Riemann Solvers for Computational Fluid Dynamics*. https://ruc.udc.es/items/85f42231-1e0d-43b6-8956-f2467ad2cc6b

Fromm, J. E., Evans, J. A., & Chen, J. S. (2025). *Interpolation-based reproducing kernel particle method* (No. arXiv:2506.14916). arXiv. https://doi.org/10.48550/arXiv.2506.14916

Gao, H., & Wei, G. (2019). Numerical solution of potential problems using radial basis reproducing kernel particle method. *Results in Physics*, *13*, 102122.
Gao, L., Qu, Y., Wang, L., & Yu, Z. (2022). Computational spectrometers enabled by nanophotonics and deep learning. *Nanophotonics*, *11*(11), 2507–2529. https://doi.org/10.1515/nanoph-2021-0636
Gnanasambandam, R., Shen, B., Chung, J., Yue, X., Zhenyu, & Kong. (2022). *Self-scalable Tanh (Stan): Faster Convergence and Better Generalization in Physics-informed Neural Networks* (No. arXiv:2204.12589). arXiv. https://doi.org/10.48550/arXiv.2204.12589
Grillo, N., Rowbottom, J., Liò, P., Schönlieb, C. B., & Fresca, S. (2025). *HypeR Adaptivity: Joint hr-Adaptive Meshing via Hypergraph Multi-Agent Deep Reinforcement Learning* (No. arXiv:2512.10439). arXiv. https://doi.org/10.48550/arXiv.2512.10439
Hoa, D. N. T., Phu, N. N. T., Nghia, P. D., Anh, N. P. T., & Lap, H. V. (2025). Optimizing Task Handling in Mobile Edge Computing: A Dynamic Load Balancing Model with Intermediate Coordination Node and Nearest Neighbor Integration. In P. G. Bradford, S. A. Gadsden, S. K. Koul, & K. P. Ghatak (Eds.), *Proceedings of IEMTRONICS 2024* (Vol. 1229, pp. 103–111). Springer Nature Singapore. https://doi.org/10.1007/978-981-97-4780-1_8
Huang, Y., Liu, H., Zhao, Y., & Fei, M. (2025). Chebyshev spectral approximation-based physics-informed neural network for solving higher-order nonlinear differential equations. *Engineering with Computers*, *41*(2), 1191–1210. https://doi.org/10.1007/s00366-024-02073-0
Jalil, L. E., Malik, N. H., & Hussein, M. K. (2025). A Comparative Study of Partial Differential Equation Solving Methods and Their Applications. *Bilangan: Jurnal Ilmiah Matematika, Kebumian Dan Angkasa*, *3*(1), 01–15.
Jayasankar, A., & Ollivier Gooch, C. F. (2022, January 3). Defect Correction on Unstructured Finite Volume Solvers. *AIAA SCITECH 2022 Forum*. AIAA SCITECH 2022 Forum, San Diego, CA & Virtual. https://doi.org/10.2514/6.2022-0220
Jiang, Y., Nian, M., & Zhang, Q. (2022). A stable generalized finite element method coupled with deep neural network for interface problems with discontinuities. *Axioms*, *11*(8), 384.
Johansson, S., Engqvist, J., Tryding, J., & Hall, S. A. (2022). Microscale deformation mechanisms in paperboard during continuous tensile loading and 4D synchrotron X-ray tomography. *Strain*, *58*(5), e12414. https://doi.org/10.1111/str.12414
Kiran, U., Gautam, S. S., & Sharma, D. (2020). GPU-based matrix-free finite element solver exploiting symmetry of elemental matrices. *Computing*, *102*(9), 1941–1965. https://doi.org/10.1007/s00607-020-00827-4
Kiran, U., Sharma, D., & Gautam, S. S. (2024). Development of GPU-based matrix-free strategies for large-scale elastoplasticity analysis using conjugate gradient solver. *International Journal for Numerical Methods in Engineering*, *125*(7), e7421. https://doi.org/10.1002/nme.7421
Kumar, V., Laxminarayanamma, K., Singh, A. K., Shukla, B., & Mondal, S. R. (2025). A machine-learning approach to weight approximation for a new family of orthogonal polynomials. *AIMS MATHEMATICS*, *10*(8), 18861–18886.
Li, Q., Liu, S., Hu, L., & Liu, X. (2020). Shape correspondence using anisotropic Chebyshev spectral CNNs. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14658–14667. http://openaccess.thecvf.com/content_CVPR_2020/html/Li_Shape_correspondence_using_anisotropic_Chebyshev_spectral_CNNs_CVPR_2020_paper.html
Mahariq, I., Giden, I. H., Alboon, S., Aly, W. H. F., Youssef, A., & Kurt, H. (2022). Investigation and analysis of acoustojets by spectral element method. *Mathematics*, *10*(17), 3145.

Mondal, S. R., & Kumar, V. (2025). An Orthogonal Polynomial Solution to the Confluent-Type Heun's Differential Equation. *Mathematics*, *13*(8), 1233.

Montoya, T. (2024). *Provably Stable Discontinuous Spectral-Element Methods with the Summation-by-Parts Property: Unified Matrix Analysis and Efficient Tensor-Product Formulations on Curved Simplices* [PhD Thesis, University of Toronto (Canada)]. https://search.proquest.com/openview/c9ff108ba807eae316860562114fb3d8/1?pq-origsite=gscholar&cbl=18750&diss=y

Mu, M., Feng, L., Zhang, Q., Zang, W., & Wang, H. (2022). Study on abrasive particle impact modeling and cutting mechanism. *Energy Science & Engineering*, *10*(1), 96–119.

Mukherjee, A., Sarkar, S., & Banerjee, A. (2021). Nonlinear eigenvalue analysis for spectral element method. *Computers & Structures*, *242*, 106367.

Munch, P., Ljungkvist, K., & Kronbichler, M. (2022). Efficient Application of Hanging-Node Constraints for Matrix-Free High-Order FEM Computations on CPU and GPU. In A.-L. Varbanescu, A. Bhatele, P. Luszczek, & B. Marc (Eds.), *High Performance Computing* (Vol. 13289, pp. 133–152). Springer International Publishing. https://doi.org/10.1007/978-3-031-07312-0_7

Nguyen, H. A., Tanaka, S., & Bui, T. Q. (2023). Material interface modeling by the enriched RKPM with stabilized nodal integration. *Computational Particle Mechanics*, *10*(6), 1733–1757. https://doi.org/10.1007/s40571-023-00585-y

Nooraiepour, M. (2025). Traditional and machine learning approaches to partial differential equations: A critical review of methods, trade-offs, and integration. *Preprints (Sept. 2025). Doi*, *10*, 20944.

Olver, S., Slevinsky, R. M., & Townsend, A. (2020). Fast algorithms using orthogonal polynomials. *Acta Numerica*, *29*, 573–699.

Patil, M. A., & Kadoli, R. (2025). State-of-the-Art in Different Formulations of Super Convergent Mesh-Less Differential Quadrature Method. *Archives of Computational Methods in Engineering*. https://doi.org/10.1007/s11831-025-10423-4

Petersen, M. S., Weinberg, M. D., & Katz, N. (2022). EXP: N-body integration using basis function expansions. *Monthly Notices of the Royal Astronomical Society*, *510*(4), 6201–6217.

Pinskier, J., & Howard, D. (2022). From Bioinspiration to Computer Generation: Developments in Autonomous Soft Robot Design. *Advanced Intelligent Systems*, *4*(1), 2100086. https://doi.org/10.1002/aisy.202100086

Qin, Y.-X., Liu, Y.-Y., Li, Z.-H., & Yang, M. (2014). An interpolating reproducing kernel particle method for two-dimensional scatter points. *Chinese Physics B*, *23*(7), 070207.

Rowbottom, J., Maierhofer, G., Deveney, T., Mueller, E., Paganini, A., Schratz, K., Liò, P., Schönlieb, C.-B., & Budd, C. (2025). *G-Adaptivity: Optimised graph-based mesh relocation for finite element methods* (No. arXiv:2407.04516). arXiv. https://doi.org/10.48550/arXiv.2407.04516

Shankar, V., & Wright, G. B. (2018). Mesh-free semi-Lagrangian methods for transport on a sphere using radial basis functions. *Journal of Computational Physics*, *366*, 170–190.

Sherwin, S. J. (2025). A comparative study on polynomial dealiasing and split form discontinuous Galerkin schemes for under-resolved turbulence computations. *Journal of Computational Physics*. https://www.academia.edu/download/124386804/1711.pdf

Shior, M. M., Agbata, B. C., Obeng-Denteh, W., Kwabi, P. A., Ezugorie, I. G., Marcos, S., Asante-Mensa, F., & Abah, E. (2024). Numerical solution of partial differential equations using MATLAB: Applications to one-dimensional heat and wave equations. *Scientia Africana*, *23*(4), 243–254.

Sime, N., Maljaars, J. M., Wilson, C. R., & van Keken, P. (2022). An exactly mass conserving and pointwise divergence free velocity method: Application to compositional buoyancy driven flow problems in geodynamics. *Authorea Preprints*. https://essopenarchive.org/doi/full/10.1002/essoar.10503932

Tuchkin, Y. A., Mazlumi, F., Sever, E., & Dikmen, F. (2022). Contour Smoothing for Super-Algebraically Convergent Algorithms of 2-D Diffraction Problems. *IEEE Transactions on Antennas and Propagation*, *70*(7), 6084–6088.

Wang, J., Hillman, M., Wilmes, D., Magallanes, J., & Bazilevs, Y. (2025). Smoothed naturally stabilized RKPM for non-linear explicit dynamics with novel stress gradient update. *Computational Mechanics*, *75*(1), 137–158. https://doi.org/10.1007/s00466-024-02494-0

Yip, C., Seol, L., & Hon, X. Z. (2022). A Step-by-Step Approach to Partial Differential Equations. *Fusion of Multidisciplinary Research, An International Journal*, *3*(1), 302–315.

Zhang, C., Hu, X. Y., & Adams, N. A. (2017). A generalized transport-velocity formulation for smoothed particle hydrodynamics. *Journal of Computational Physics*, *337*, 216–232.

Zhang, J., Su, R., Fu, Q., Ren, W., Heide, F., & Nie, Y. (2022). A survey on computational spectral reconstruction methods from RGB to hyperspectral imaging. *Scientific Reports*, *12*(1), 11905.

Zienkiewicz, O. C., Huang, G. C., & Liu, Y. C. (1990). Adaptive FEM computation of forming processes—Application to porous and non-porous materials. *International Journal for Numerical Methods in Engineering*, *30*(8), 1527–1553. https://doi.org/10.1002/nme.1620300812