CSIAM Trans. Appl. Math.
doi: 10.4208/csiam-am.SO-2025-0074

Vol. **7**, No. 1, pp. 1-28
March 2026

Data- and Mechanism-Driven Hybrid Computing: A New Paradigm for Scientific and Engineering Computation

Jerry Zhijian Yang[1,2,3,4,*] and Pingwen Zhang[3,5]

[1] *School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China.*
[2] *National Center for Applied Mathematics in Hubei, Wuhan 430072, China.*
[3] *Institute for Math and AI, Wuhan University, Wuhan 430072, China.*
[4] *Hubei Key Laboratory of Computational Science, Wuhan University, Wuhan 430072, China.*
[5] *School of Mathematical Sciences, Peking University, Beijing 100871, China.*

Received 14 July 2025; Accepted 2 November 2025

Abstract. Data- and mechanism-driven hybrid computing refers to the integration of traditional mechanism-based computing with data-driven methods. In this article, we present three typical patterns of this emerging paradigm: (1) mechanism-driven model optimization via data-driven refinement, (2) data-driven model construction with physical constraints, and (3) alternating optimization of mechanism-driven and data-driven models. We present several concrete examples to illustrate how hybrid computing improves accuracy, efficiency, and robustness across a variety of computational tasks.

AMS subject classifications: 65N20, 68U01, 68T01

Key words: Data-driven methods, mechanism-driven models, hybrid computing.

1 Introduction

Scientific and engineering computations primarily serve four key purposes: perception, design, prediction and comprehension. Over the past few decades, researchers have developed a wide range of algorithms to address various problems aligned with these goals, applying them to real-world domains such as oil exploration, structural design, weather forecasting and materials modeling. Traditionally, these computations rely on mathematical models such as partial differential equations (PDEs) to capture physical mechanisms,

*Corresponding author. *Email addresses:* `zjyang.math@whu.edu.cn` (J. Z. Yang), `pzhang@pku.edu.cn` (P. Zhang)

and utilize numerical solvers and optimization techniques for computation [16,65]. More recently, the rise of machine learning has introduced a data-driven paradigm, offering new approaches to modeling and computation that have significantly advanced the field of scientific computing [22]. Before delving deeper, we briefly outline these four objectives as follows:

1. Perception. Perception involves extracting information about the environment or a given situation from measurement data. For instance, in oil exploration, inverse problems are solved to infer subsurface properties of the Earth – such as the spatial distribution of reservoirs-based on indirect measurements like seismic waves or electromagnetic fields [81]. Another representative example is situation awareness (SA) [23], where the goal is to construct a comprehensive understanding of the current and near-future state of an environment and its components. This understanding can then support informed decision-making and strategic planning in various domains, such as military operations or traffic control systems.
2. Design. Design tasks utilize computer simulations to assist in creating, modifying, analyzing, or optimizing engineering systems. These tasks are widely applied in the field of computer-aided design (CAD) [35]. For instance, in bridge engineering, finite element analysis can predict structural behavior under different loads and environmental conditions. Such predictive capabilities enable engineers to fine-tune design parameters to ensure compliance with safety and performance standards.
3. Prediction. Prediction focuses on forecasting the future evolution and state of dynamic systems. A prominent example is numerical weather prediction (NWP) [13], which simulates atmospheric dynamics using fluid mechanics equations to generate reliable weather forecasts several days or even weeks in advance.
4. Comprehension. Comprehension leverages numerical computation as a powerful tool for developing physical models that align with experimental observations. For example, in molecular simulations, potential energy surfaces of materials are constructed by fitting both ab initio calculations and experimental data [77]. These surfaces enable a deeper understanding of material behavior at the atomic level. Building upon this, molecular dynamics simulations can further uncover fundamental mechanisms underlying physical phenomena such as phase transitions.

Across these computational tasks, methods are commonly evaluated based on three key attributes:

1. Accuracy. It refers to how closely the computational results match the true values.
2. Efficiency. It measures the computational resources such as time and memory.
3. Robustness. It indicates the method's ability to perform reliably under varying conditions, such as changing parameters, model uncertainties or noisy data.

To improve performance in these aspects, numerous frameworks and algorithms have been proposed during the past several decades. However, significant challenges persist across different applications. In perception and comprehension tasks, the core difficulty lies in accurately reconstructing a system's state or building models from limited observational data, while adhering to established physical laws. In design problems, the efficiency and robustness of simulation algorithms are critical to supporting downstream optimization processes. For prediction tasks, the goal is typically to develop accurate, explainable models capable of long-term forecasting.

Traditionally, these challenges were addressed using mechanism-based paradigms, where mathematical models derived from first principles in physics, chemistry, and other sciences are solved via numerical algorithms [16, 65]. More recently, the rapid advancement of machine learning has led to a powerful data-driven paradigm, which builds models and solves complex problems by learning directly from observational or simulation data. In general, while both paradigms have achieved immense success independently, their inherent limitations have become increasingly apparent. Mechanism-based models offer high interpretability but often struggle with systems of immense complexity or phenomena that are difficult to model from first principles. Conversely, data-driven models demonstrate remarkable flexibility and adaptability but may lack physical interpretability and can be sensitive to the quality and quantity of available data.

Recognizing these complementary strengths and weaknesses has catalyzed the development of a third, hybrid paradigm: data- and mechanism-driven hybrid computing. This emerging approach seeks to synergistically combine physical principles with data-driven techniques, showing enormous potential to advance scientific and engineering computation. Taking the NWP as an example: Traditional NWP methods generate forecasts by numerically solving high-dimensional partial differential equations derived from atmospheric physics. While these models are interpretable and capable of long-range forecasting, their accuracy often degrades rapidly due to uncertainties in initial conditions and model parameters. To mitigate this, operational forecasting incorporates data assimilation, which fuses past forecasts with newly acquired observational data to produce more accurate initial states for subsequent predictions. Furthermore, this hybrid methodology – combining data-driven learning with physics-based modeling – not only improves short-term forecasting but also enables the generation of long-term reanalysis datasets. These datasets provide accurate reconstructions of historical weather patterns, which can be used to train more efficient machine learning-based weather prediction (MLWP) models and to gain deeper insights into long-term climate change.

In this review, we provide a systematic overview of this rapidly evolving field. We start by tracing its historical development from the perspectives of both mechanism-based and data-driven paradigms, highlighting their respective contributions and evolution. We then categorize data- and mechanism-driven hybrid computing into three patterns, each embodying a distinct mode of integration designed to address specific problem settings. Based on this categorization, we present several illustrative examples to demonstrate how hybrid computing enhances both the accuracy, efficiency and robustness of a wide range of computational tasks.

1.1 The mechanism-based computing paradigm

The early stages of scientific and engineering development were characterized by a reliance on fundamental principles and laws of natural phenomena such as physics and chemistry to establish models and perform related calculations. For example, Newton established the classical mechanics system based on Newton's laws of motion and universal gravitation, which could accurately calculate celestial motion and mechanical behavior of objects. Mathematicians such as Euler established classical models like the Euler equations in fluid mechanics to describe fluid motion. These examples epitomize mechanism-based computation, which uses mathematical expressions of natural laws and solves problems through mathematical analysis and numerical calculation methods, providing important theoretical support for engineering technology and scientific research.

In the 20th century, the rise of computer technology greatly expanded the growth potential of mechanism-based computation. Numerical methods such as the finite element method and finite difference method became increasingly mature [12, 82], enabling complex physical problems to be numerically solved through computers. In aerospace, computational fluid dynamics (CFD) methods based on aerodynamic mechanisms were widely applied to aircraft design, simulating airflow distribution around aircraft by solving fluid dynamics equations such as the Navier-Stokes equations [1], and optimizing aircraft shape design. In structural mechanics, the finite element method was used to calculate stress and deformation of structures such as buildings and bridges under force conditions to ensure structural safety. In particular, mechanism-based scientific and engineering computation has also provided significant benefits in the development of nuclear weapons and other areas of national defense and security.

1.2 The data-driven computing paradigm

In the mid-to-late 20th century, with the rapid development of information technology, data collection and storage capabilities continuously improved, amassing vast amounts of data across various fields. Statistical methods were widely applied in data analysis, such as regression analysis and cluster analysis [30], used to discover patterns, make predictions, and inform decisions. In economics, for instance, statistical models were built from large datasets to analyze and predict economic trends. In medicine, clinical data analysis helped evaluate drug efficacy and assess disease risks.

The 21st century has witnessed a paradigm shift in data-driven computing, fueled by the emergence of machine learning and, in particular, deep learning techniques [48]. They enabled automatic feature extraction from massive datasets, achieving remarkable results in fields such as image recognition and natural language understanding. For example, facial recognition systems trained on large datasets can now achieve high identification accuracy [58], and voice assistants can interpret human language based on large-scale audio data [31]. Data-driven computing emphasizes the scale and quality of data, demonstrating strong adaptability and performance in real-world tasks.

1.3 The data- and mechanism-driven hybrid computing paradigm

Although the above mechanism-driven and data-driven computational approaches have achieved significant success in various problem cases, they still struggle to achieve a comprehensive balance in terms of computational accuracy, efficiency, and robustness. Table 1 below compares several advantages and disadvantages of the two paradigms.

Among the significant limitations outlined in Table 1, the sensitivity of mechanism-driven models to their parameters is a critical challenge that has driven the development of an integrated paradigm. For instance, the well-known "butterfly effect" in weather forecasting shows how minute uncertainties in initial conditions can yield vastly different outcomes. Similarly, in computational fluid dynamics, slight adjustments to empirical constants within a given turbulence model, such as the $k-\epsilon$ model, can significantly alter the predicted drag forces and vortex shedding frequencies. In the same vein, for a specific constitutive law modeling a material, minor variations in its material parameters fitted from experimental data can fundamentally change the simulation's prediction of stress concentration locations and fatigue life, leading to completely different conclusions about a structure's durability. It is precisely to mitigate such uncertainties and to better balance accuracy, efficiency, and robustness that a new paradigm based on the integration of data and mechanism-driven computing has made substantial progress in recent years. One early and influential example of this integration is numerical weather prediction itself, where mechanistic models grounded in atmospheric dynamics are enhanced through data assimilation techniques that incorporate observational data, substantially

Table 1: Comparison of data-driven and mechanism-driven approaches.

Method	Advantages	Disadvantages
Mechanism-driven computing	• Strong interpretability. • Theoretical guarantees. • Extrapolation ability. • Reliable predictions over long time.	• High computational cost. • Requirement of mechanisms modeling. • Sensitive to model parameters.
Data-driven computing	• High computational efficiency. • Extracts patterns automatically. • Generalization ability with enough data.	• Lack of interpretability. • Sensitive to the quality and quantity of data. • Lack of applicable convergence analysis.

improving forecast accuracy [43]. This demonstrated that fusing physical principles with data-driven correction can outperform either approach alone.

In recent years, this hybrid paradigm has rapidly expanded across a wide array of scientific and engineering fields. In materials science, the combination of first-principles calculations with machine learning addresses a key challenge by making free-energy calculations routine at a fraction of the computational cost and with high accuracy [41]. In biomedicine, the integration of biophysical mechanisms with large-scale biomedical data facilitates more accurate disease diagnosis and supports drug discovery [51]. In the energy sector, physical models of power systems are combined with real-time operational data to enhance the efficiency of scheduling and resource management. A representative example is deep potential molecular dynamics, which uses first-principles simulations to generate high-fidelity training data and then employs deep neural networks to learn interatomic potentials [29], achieving both high accuracy and computational efficiency.

The overarching goal of data- and mechanism-driven hybrid computing is to enhance predictive accuracy, computational efficiency, and model interpretability by synergistically leveraging physical laws and data within a unified computational framework, ultimately aiming to provide more robust and effective solutions for complex scientific and engineering applications. In the following sections, we identify and elaborate on three representative integration patterns that characterize the hybrid computing landscape:

1. Mechanism-driven model optimization via data-driven refinement.
2. Data-driven model construction with physical constraints.
3. Alternating optimization of mechanism-driven and data-driven models.

Each of these patterns will be illustrated with concrete examples drawn from different scientific and engineering domains (see Fig. 1 and Table 2).

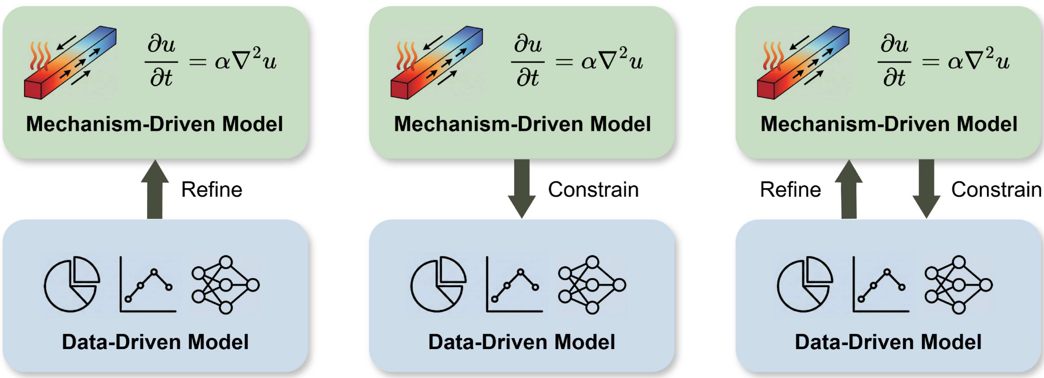


Figure 1: Three paradigms of hybrid computing integrating data- and mechanism-driven models. Left: Data-driven refinement of mechanism-based models. Middle: Physically constrained data-driven modeling. Right: Alternating optimization of data- and mechanism-driven models.

Table 2: Summary of the three hybrid computing paradigms.

Paradigm	Definition	Examples
Mechanism-driven model optimization via data-driven refinement	Refine a mechanism-based model by leveraging data to optimize its parameters or unknown components.	• DPMD (Section 2.1) • Data-driven priors (Section 2.2)
Data-driven model construction with physical constraints	Embed physical constraints into a data-driven model to ensure its physical consistency.	• PINNs (Section 3.1) • Constitutive learning (Section 3.2)
Alternating optimization of mechanism-driven and data-driven models	Enable a mechanism-driven model and a data-driven model to co-evolve through a process of reciprocal enhancement.	• AlphaZero (Section 4.1) • Weather forecasting (Section 4.2)

2 Mechanism-driven model optimization via data-driven refinement

This paradigm utilizes mechanism-based models as the foundational framework, subsequently leveraging data to refine or optimize key aspects such as model parameters, boundary conditions, or even unknown functional components. The primary objectives are to enhance predictive accuracy and improve computational efficiency. However, relying solely on mechanism-based models often encounters significant limitations.

One prominent application domain where such challenges arise is molecular simulation. While foundational theories like quantum mechanics provide a rigorous basis for describing molecular and atomic interactions, their direct application to large systems often incurs prohibitive computational costs. For instance, density functional theory (DFT) offers a robust quantum mechanical framework for electronic structure calculations [32, 45], but its high computational complexity can limit its scalability, especially in long-timescale or large-system simulations, thereby posing challenges for studying complex molecular phenomena.

Another domain where similar limitations are encountered is the solution of inverse problems. While mechanism-based forward models are typically employed, obtaining stable and physically meaningful solutions often requires regularization terms crafted from expert knowledge or heuristics – such as L_1 regularization to promote sparsity or total variation regularization to encourage smoothness. However, these handcrafted priors are inherently subjective and may fail to capture the complex statistical properties or underlying structures of real-world data, thereby constraining solution fidelity and generalizability.

The following subsections examine how the paradigm of mechanism-driven model optimization via data-driven refinement offers effective strategies to address the limitations of traditional mechanism-based approaches. This section focuses on two representative examples: deep potential molecular dynamics and the application of deep priors in inverse problems. The first example addresses a comprehension problem, where the goal is to model interatomic interaction mechanisms using data-driven methods. In contrast, the second example represents a typical perception problem, aiming to recover the underlying state from limited observational data. Together, these cases demonstrate how integrating data-driven models can significantly enhance the accuracy, efficiency, and adaptability of physical modeling.

2.1 Deep potential molecular dynamics

Molecular dynamics is based on physical principles such as Newtonian mechanics, describing the motion and interaction of molecules and atoms under the action of forces. In traditional molecular dynamics simulations, precise calculations of interatomic interaction forces are required, typically based on quantum mechanics or classical force field models, which demand enormous computational resources.

Deep potential molecular dynamics (DPMD) introduces deep learning methods, constructing deep potential (DP) energy functions through learning from vast amounts of molecular structure and energy data to approximately describe interatomic interactions [67,77]. This data-driven deep potential energy function greatly improves computational efficiency while maintaining certain accuracy. By combining the deep potential energy function with molecular dynamics equations of motion, integrated computing of mechanism and data is achieved. In simulating complex processes such as material phase transitions and chemical reactions, this approach utilizes physical mechanisms to ensure simulation rationality and accuracy while leveraging data-driven methods to enhance computational speed and scalability, providing a powerful tool for research in materials science, chemistry, and other fields.

We provide a brief introduction to the development of DP model. The Schrödinger equation, proposed in 1926, is a fundamental principle for describing atomic systems comprising nuclei and electrons [57]. In its static form, the equation is expressed as

$$\hat{H}\Psi = E\Psi,$$

where $\Psi(\mathbf{r},\mathbf{R})$ represents the wave function, with $\mathbf{r}$ indicating the positions of the electrons and $\mathbf{R}$ denoting the positions of the nuclei. The Hamiltonian operator, $\hat{H}$, is defined as

$$\hat{H} = -\sum_i \frac{\hbar^2}{2m_i}\nabla_i^2 - \sum_{i,I}\frac{Z_I}{|r_i - R_I|} + \sum_{i<j}\frac{1}{|r_i - r_j|}.$$

It is important to note that the Schrödinger equation does not account for relativistic effects. Solving the Schrödinger equation provides comprehensive insight into the behavior of atomic systems. However, the high dimensionality of the equation presents significant challenges in numerical method development. For instance, the dimensionality reaches 39 for a single water molecule, posing substantial computational difficulties [21].

Born and Oppenheimer [10] noted that the relaxation of electronic degrees of freedom occurs much more rapidly than that of nuclear degrees of freedom. Consequently, one can treat the nuclear coordinates as fixed parameters and solve only for the electronic wave function. The ground state solution under this approximation is given by

$$E(\mathbf{R}) = \min_{\Psi} \frac{\int \Psi^*(\mathbf{r};\mathbf{R}) \hat{H} \Psi(\mathbf{r};\mathbf{R}) d\mathbf{r}}{\int \Psi^*(\mathbf{r};\mathbf{R}) \Psi(\mathbf{r};\mathbf{R}) d\mathbf{r}}.$$

This expression defines a mapping from the nuclear coordinates to the ground state energy of the system. The function $E(\mathbf{R})$ is often referred to as the potential energy surface (PES), which is central to modeling atomic systems. Compared to the original Schrödinger equation, the Born-Oppenheimer approximation reduces the degrees of freedom, thereby simplifying the problem. However, the electronic Schrödinger equation remains high-dimensional. For example, in a water molecule, the dimensionality is reduced to 30, which still presents challenges related to the curse of dimensionality.

A significant advancement in achieving computationally feasible solutions is the development of Kohn-Sham density functional theory (KS-DFT) [45]. This approach simplifies the high-dimensional electronic Schrödinger equation to a three-dimensional problem under the assumption of ground state conditions. KS-DFT stands as one of the most widely used first-principles methods for modeling atomic systems due to its effective balance between accuracy and computational efficiency. Its widespread success, the computational complexity of KS-DFT remains $\mathcal{O}(N^3)$, where N represents the number of electronic degrees of freedom in the system. This cubic scaling results in a rapid increase in computational cost as the system size expands. From 2006 to 2019, the peak performance of the world's fastest supercomputers improved by a factor of 550, yet the maximum number of electronic degrees of freedom that could be handled only increased by 8.8 times, closely aligning with the $\mathcal{O}(N^3)$ scaling law [37].

To facilitate the simulation of large-scale atomic systems using accessible computational resources, the empirical force field (EFF) approach has been developed. This approach circumvents the electronic structure problem posed by the KS-DFT equation by directly constructing the PES, denoted as $E(\mathbf{R})$. The EFF methodology formulates the PES based on physical principles and mathematical asymptotic behavior, employing adjustable parameters that are typically fitted to experimental observation data. The inception of EFFs dates back to the early 20th century and they have rapidly evolved into practical solutions for computer simulations. Notable examples of successful EFFs include Amber [68], OPLS [42], and GROMOS [17] for biomolecular systems, as well as the embedded atom method (EAM) [18] and modified embedded atom method (MEAM) [5] for materials systems. The primary advantage of EFFs is their computational complexity of $\mathcal{O}(N)$, where N is the number of nuclei, allowing for the simulation of large-scale atomic systems with significantly reduced computational resource requirements compared to KS-DFT.

Despite widespread use, EFFs typically show lower accuracy than first-principles calculations due to two main limitations. First, the functional form used to fit the PES is constrained with limited expressive power, which can become problematic as the complexity of the atomic system increases, such as with the introduction of multiple elements or complex structures like interfaces and defects. Consequently, EFFs are often designed to address a limited range of fitting targets. Second, while EFFs tend to perform well for properties within their training scope, their ability to generalize to properties outside this scope is not guaranteed. These limitations not only complicate the development of EFFs but also restrict their applicability to problems beyond the training data, thereby limiting their capacity to address broader scientific challenges.

Deep learning has emerged as a potent tool for modeling high-dimensional functions [48], thereby offering promising opportunities for PES modeling. The training of deep learning models falls under the category of supervised learning. By utilizing a set of nuclear configurations $\{\mathbf{R}_i\}$ and corresponding ground-state energies $\{E(\mathbf{R}_i)\}$ computed via first-principles methods, the model parameters can be optimized to minimize the discrepancy between the model's predictions and first-principles energies. This optimization is typically achieved using advanced training techniques such as stochastic gradient descent. It is important to note that deep learning-based PES modeling is not purely data-driven; the models must adhere to fundamental physical principles during simulations. For instance, conservation laws for energy, momentum, and angular momentum impose constraints on the model's design. To ensure energy conservation, the model's force must be conservative, calculated as the negative gradient of the energy. Similarly, translational and rotational symmetries of the energy must be upheld to preserve momentum and angular momentum, in accordance with Noether's theorem. Deep learning PES models effectively address the first limitation of EFF by providing the capacity to accurately fit complex PES. This capability arises from the universal approximation properties of deep neural networks [4]. However, similar to EFFs, deep learning PES models often face challenges regarding generalizability, a topic that will be explored in detail later in this work.

Since 2007, various deep learning potential energy models have been introduced [7, 59], with the deep potential model being a notable example [77]. The DP model, trained using data labeled by first-principles calculations, achieves accuracy comparable to these calculations while reducing computational complexity to $\mathcal{O}(N)$, similar to EFFs. This reduction in complexity enables the scaling of atomic system simulations to larger scales on modern high-performance supercomputers. For instance, the DP model has demonstrated scalability to atomic systems comprising up to 10 billion atoms while maintaining simulation accuracy [28, 37]. This capability has facilitated the resolution of numerous scientific challenges, such as computing the water phase diagram [79], discovering the interfacial reactive uptake of N_2O_5 by atmospheric aerosols [24], and developing fatigue-resistant ferroelectrics [9].

The primary challenge encountered by deep learning-based PES models is their limited ability to generalize to out-of-distribution data. Out-of-distribution refers to scenarios where the test dataset's distribution does not overlap with the training data in the latent space representation developed by the deep learning models. There are two main approaches to address this issue.

The first approach to addressing the generalization issue is data-driven. This involves systematically generating additional training data to ensure that all configurations relevant to the application are encompassed within the training data's distribution. However, this method can be inefficient. In high-dimensional spaces, determining whether a particular data point lies within a given distribution is non-trivial. Additionally, indiscriminately generating large volumes of data is impractical due to the high cost associated with obtaining training labels, which are typically derived from first-principles calculations. In this context, active learning [64] or concurrent learning [78] algorithms have been developed. These algorithms iteratively refine the training dataset. In each iteration, the model is allowed to explore new data points in a manner akin to its intended use, such as through molecular dynamics simulations [78] or crystal structure predictions [71]. During this process, the model's error on the newly explored data is estimated. Only data points with relatively large errors are selected for labeling, as they are deemed most critical for enhancing the model's accuracy. This iterative approach continues until the model's error on the explored data ceases to decrease. The concurrent learning approach effectively produces models with consistent accuracy across the application-relevant configuration space while minimizing the need for expensive first-principles calculations for data labeling.

The second approach involves embedding more prior knowledge into the model, effectively altering the development of its hidden representations. This allows for non-overlapping data distributions to share common elements in the newly encoded representation. A notable example of this strategy is the encoding of atom types into a uniform hidden space, providing a consistent representation across different atom types. In scenarios where two configurations have atoms with identical coordinates but differ only in chemical species, the atom type encoding facilitates a more unified representation compared to methods that employ separate networks for different atom types. This technique has been demonstrated to significantly enhance the generalizability of models, particularly in the context of high-entropy alloys [75].

First-principles models are universally accurate, meaning they are always correct with respect to atomic coordinates and chemical species. Consequently, it is theoretically feasible to develop a universal model that serves as a surrogate for first-principles calculations across all scenarios. Recent efforts have emerged in this direction [6,19,50]. A prominent example is GNoME [50], proposed by DeepMind, which has led to the discovery of approximately 381,000 new materials, thereby significantly accelerating the materials discovery process. When trained on data encompassing a vast chemical and configurational space, these models can extract chemical knowledge from their hidden representations. For instance, a visualization of the atom type encoding in the DPA-1 model within a three-dimensional space reveals an arrangement of elements mirroring the periodic table, albeit in a spiral configuration. The spiral's direction corresponds to the rows of the periodic table, while the normal direction aligns with the columns [75]. This finding suggests the potential for discovering new principles through the training of large atomic models in the future.

2.2 Solving inverse problems with data-driven priors

Inverse problems in image processing and material modeling involve recovering original signals or model parameters from noisy observations. Mathematically, such problem can be formulated as follows: given noisy measurement data $\mathbf{y}\in\mathbb{R}^q$ modeled by $\mathbf{y}=\mathcal{F}(\mathbf{x})+\boldsymbol{\eta}$, where $\mathcal{F}:\mathbb{R}^p\to\mathbb{R}^q$ denotes a known forward measurement operator and $\boldsymbol{\eta}\in\mathbb{R}^q$ represents observational noise, the goal is to recover the unknown state $\mathbf{x}\in\mathbb{R}^p$. In practice, the forward operator $\mathcal{F}$ encapsulates the underlying physical or measurement process. For instance, in image deblurring, $\mathcal{F}$ typically corresponds to a convolution operator with some filter function. In the inverse source problem of Poisson equation $\Delta u(x)=f(x)$, $\mathcal{F}$ maps the source term f into the observable solution u.

In general, solving such an inverse problem may be ill-posed. To address this issue, analytical regularization techniques are commonly employed to enforce the existence, uniqueness, and stability of the solution. This leads to

$$\mathbf{x}^*=\underset{\mathbf{x}\in\mathbb{R}^p}{\operatorname{argmin}}\left\{\mathcal{L}\big(\mathcal{F}(\mathbf{x}),\mathbf{y}\big)+g(\mathbf{x})\right\}.$$

Here, $\mathcal{L}$ denotes a loss functional that enforces data consistency, g represents the regularization term, which incorporates prior information about the solution. Different choices of g reflect different assumptions or desired properties of the unknown state $\mathbf{x}$. For example, sparsity can be promoted via ℓ_1-regularization, while smoothness can be encouraged using total variation (TV) regularization.

With the significant success of deep learning in computational vision, recent research has shifted focus towards various data-driven regularization methods [2]. One approach is to solve the inverse problem through a Bayesian inference. In this framework, given a noise distribution ρ for $\boldsymbol{\eta}$, the measurement likelihood can be expressed as $p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x})=\rho(\mathbf{y}-\mathcal{F}(\mathbf{x}))$. To ensure the well-posedness and fidelity of the reconstruction, prior information regarding $\mathbf{x}$ can be expressed through a prior density $\pi(\mathbf{x})$, which captures the data distribution pertinent to a domain-related dataset. With this, the posterior distribution of $\mathbf{x}$ can be formulated using the Bayesian formula

$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y})\propto p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x})\pi(\mathbf{x})=\rho\big(\mathbf{y}-\mathcal{F}(\mathbf{x})\big)\,\pi(\mathbf{x}).$$

Beyond point-wise predictions, Bayesian inference offers the advantage of providing uncertainty quantification while also presenting a natural framework for integrating data-driven knowledge into the observation model. By computing the gradient of the logarithm for $p_{\mathbf{X}|\mathbf{Y}}$, we can decouple the score of posterior distribution as the sum of log-likelihood and the score of prior density. This enables a Plug & Play (PnP) posterior sampling approach [47] across various measurement forms: By initially pretraining a shared unconditional prior $\pi(\mathbf{x})$, we can solve inverse problems with different analytical observation operators without retraining a conditional density for each distinct measurement $\mathcal{F}$. In practice, the posterior sampling can be achieved by the unadjusted Langevin algorithm (ULA)

$$X_{k+1} = X_k + \delta t \nabla_{\mathbf{x}} \log p(\mathbf{y}|X_k) + \delta t \nabla_{\mathbf{x}} \log \pi(X_k) + \sqrt{2\delta t}\boldsymbol{\eta}_k,$$

where

$$p(\mathbf{y}|\cdot) \triangleq p_{(\mathbf{Y}|\mathbf{X})}(\mathbf{y}|\cdot), \quad \delta t > 0$$

is step size and $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ is a random noise. Using these samples, the predicted state can be computed as the average of the samples, with the standard deviation serving as an error calibration metric.

In recent years, a range of deep generative models have been utilized to learn the prior density for tasks like image restoration. These include methods such as normalizing flow (NF) [15], generative adversarial networks (GANs) [52], denoising score matching-based (DSM) approaches [47], and consistency models-based (CM) methods [54]. In contrast to traditional regularization techniques, these methods leverage domain-specific data to achieve regularization that often results in superior reconstruction quality.

3 Data-driven model construction with physical constraints

This approach primarily uses data-driven methods to construct models while embedding physical constraints (such as conservation laws, symmetry, etc.) to ensure physical consistency. This strategy allows for the flexibility of data-driven techniques while upholding fundamental scientific principles, leading to more robust and reliable models.

For instance, in fluid dynamics research and other fields governed by partial differential equations, when constructing predictive models using deep learning, the governing equations (like the Navier-Stokes equations) are embedded as essential physical constraints. Physics-informed neural networks (PINNs) [55] exemplify this by incorporating these equations into the training process, often through penalty terms in the loss function, ensuring the model's predictions conform to fundamental physical laws. Similarly, in the data-driven modeling of constitutive relations for materials, large amounts of experimental or simulation data are used in conjunction with explicit physical constraints. These constraints – such as material symmetry, frame indifference, and specific tangent stiffness properties – are crucial for ensuring consistency with continuum mechanics and maintaining numerical stability, and are enforced through specialized architectures or loss functions in learning algorithms like neural networks. Furthermore, in areas like the production and evaluation of high-precision maps, traditional surveying mechanisms provide a foundational framework, while vast quantities of crowdsourced and sensor data, processed by machine learning, can be used to enhance, update, and validate these maps, with physical or logical consistency rules acting as embedded constraints to ensure map integrity and accuracy.

The following subsections detail how physics-informed neural networks, learning constitutive relations, and high-precision mapping each embody the paradigm of data-driven model construction with embedded physical constraints. While PINNs can be directly applied to design problems, such as computer-aided design for bridges and dams, the second example demonstrates how incorporating physical constraints into the learning process can deepen our understanding of the intrinsic properties of materials. The third example, by contrast, represents a typical perception problem, commonly encountered in fields such as autonomous driving, where accurate environmental mapping is essential for decision-making and navigation.

3.1 Physics-informed neural networks

Inspired by the remarkable success of deep learning in computer science, solving PDEs using neural networks has emerged as a highly promising research direction. Among these, physics-informed neural networks method has risen as a particularly practical approach [55], renowned for its simplicity and adaptability. Fundamentally, PINNs address both forward and inverse PDEs by iteratively optimizing neural networks to minimize a composite loss function, consisting of least-squares data-fitting terms and physics-informed residual terms.

Taking the inverse source problem of the Poisson equation as an example, we can elucidate the fundamental concept of PINNs as follows. Let $\Omega\in\mathbb{R}^d$ denote a bounded spatial domain. The objective is to reconstruct the source term $f^*:\mathbb{R}^d\to\mathbb{R}$ based on a noisy measurement u^δ and a boundary condition $g(\mathbf{x})$. This problem can be equivalently framed as addressing the following optimal control problem:

$$\begin{aligned}&\min_{u,f}\mathcal{J}(u)=\|u-u^\delta\|^2_{L^2(\Omega)}\\ &\text{s.t.}\ \begin{cases}-\Delta u(\boldsymbol{x})=f(\boldsymbol{x}), & \forall \boldsymbol{x}\in\Omega,\\ \quad u(\boldsymbol{x})=g(\boldsymbol{x}), & \forall\boldsymbol{x}\in\partial\Omega.\end{cases}\end{aligned}$$

To arrive at an unconstrained optimization problem, a natural way is to incorporate a physics-informed penalty term that encapsulates the PDE constraint

$$\min_{u,f}\mathcal{L}(u,f)=\mathcal{J}(u)+\mathcal{R}(u,f)+\mathcal{B}(u),$$

where the residual term $\mathcal{R}(u,f)$ and boundary constraint $\mathcal{B}(u)$ are defined as

$$\mathcal{R}(u,f)=\|\Delta u+f\|^2_{L^2(\Omega)},\quad \mathcal{B}(u)=\|u-g\|^2_{L^2(\partial\Omega)}.$$

Contrary to the traditional approach of solving a forward problem and an adjoint problem alternatively, PINNs employ two neural networks, $u_\theta(\boldsymbol{x})$ and $f_\eta(\boldsymbol{x})$, to parameterize the solution u and source f respectively. The optimal network parameters θ^* and η^* are determined by minimizing a joint loss as

$$(\theta^*,\eta^*)\in\underset{(\theta,\eta)}{\operatorname{argmin}}\,\mathcal{L}\big(u_\theta(\boldsymbol{x}),f_\eta(\boldsymbol{x})\big).$$

This strategy enables PINNs to seamlessly incorporate the PDE constraint, boundary conditions, and observation data fitting into a unified loss, thereby embedding the underlying physical principles into the solutions for u and f. In practice, the loss functional $\mathcal{L}(u_\theta(\boldsymbol{x}),f_\eta(\boldsymbol{x}))$ is usually discretized by Monte Carlo integration

$$\widehat{\mathcal{L}}\big(u_\theta(\boldsymbol{x}),f_\eta(\boldsymbol{x})\big)=\widehat{\mathcal{J}}(u_\theta)+\widehat{\mathcal{R}}(u_\theta,f_\eta)+\widehat{\mathcal{B}}(u_\theta),$$

where $\widehat{\mathcal{J}}_{\mathcal{D}_1}(u_\theta)$, $\widehat{\mathcal{R}}_{\mathcal{D}_2}(u_\theta,f_\eta)$ and $\widehat{\mathcal{B}}_{\mathcal{D}_3}(u_\theta)$ are defined as

$$\begin{aligned}
\widehat{\mathcal{J}}_{\mathcal{D}_1}(u_\theta)&=\frac{1}{N_1}\sum_{i=1}^{N_1}\big(u_\theta\big(\boldsymbol{x}_i^{obs}\big)-u_i^{obs}\big)^2,\\
\widehat{\mathcal{R}}_{\mathcal{D}_2}(u_\theta,f_\eta)&=\frac{1}{N_2}\sum_{i=1}^{N_2}\big(\Delta u_\theta\big(\boldsymbol{x}_i^{in}\big)+f_\eta\big(\boldsymbol{x}_i^{in}\big)\big)^2,\\
\widehat{\mathcal{B}}_{\mathcal{D}_3}(u_\theta)&=\frac{1}{N_3}\sum_{i=1}^{N_3}\big(u_\theta\big(\boldsymbol{x}_i^{bd}\big)-g\big(\boldsymbol{x}_i^{bd}\big)\big)^2.
\end{aligned}$$

Here $\mathcal{D}_1=\{\boldsymbol{x}_i^{obs},u_i^{obs}\}_{i=1}^{N_1}$ is the set of observation data, $\mathcal{D}_2=\{\boldsymbol{x}_i^{in}\}_{i=1}^{N_2}$ and $\mathcal{D}_3=\{\boldsymbol{x}_i^{bd}\}_{i=1}^{N_3}$ are collocation points sampled randomly in Ω and on $\partial\Omega$ respectively. Utilizing optimization algorithm such as stochastic gradient decent, the weights in neural networks can be trained efficiently.

Over the past few years, considerable effort has been devoted to the algorithm refinement [25, 40, 70] and theoretical analysis [38, 39, 60] of PINNs. Along with these, PINNs have found extensive applications across diverse domains such as computational fluid mechanics [14], solid mechanics [69], and structural topology optimization [36]. Notably, the development of a multi-stage boosting strategy has enabled PINNs to achieve prediction errors approaching machine precision $\mathcal{O}(10^{-16})$, establishing them as a powerful paradigm for solving partial differential equations [70].

3.2 Learning constitutive relations with physical constraints

Constitutive relations, such as stress-strain laws, play a significant role in material modeling. It establishes a connection between mechanical stress tensors and kinematical quantities. Integration of these relations with governing equations, encompassing the continuity equation, momentum balance, and energy conservation, yields a closed system of equations, defining well-structured boundary value problems. Phenomenologically, these relations are often established by initially postulating a functional form and subsequently refining the model through empirical experimentation. Alternatively, atomistic models can be employed to derive constitutive relations from fundamental principles.

With the development of deep learning, data-driven methods have been proposed to construct the constitutive relations from both direct observations (such as strain and stress) and indirect measurements (like loads and deformation). Analogous to traditional phenomenological models, it is essential to incorporate various physical constraints into the model architecture or learning process. For instance, in a recent work by Xu *et al.* [73], researchers devised a symmetric positive definite (SPD) neural network to model relationships for solid materials. Taking the nonlinear elastic materials for example, the constitutive relation reads as

$$\dot{\boldsymbol{\sigma}} = H\dot{\boldsymbol{\epsilon}}.$$

Here, $\boldsymbol{\sigma}$ and $\boldsymbol{\epsilon}$ represent the stress and strain tensors, respectively, and H denotes the tangent stiffness matrix, which should be symmetric positive definite (SPD) in cases involving strain hardening. The SPD property of H ensures the unique solvability of the associated boundary-value problem, while the rate form necessitates a time consistency constraint as

$$\lim_{\Delta\boldsymbol{\epsilon}\to 0} \Delta\boldsymbol{\sigma} = 0.$$

To ensure both the SPD property and time consistency, [73] proposes a SPD-neural network (SPD-NN) $H_\theta \triangleq L_\theta L_\theta^\top$ to learn the constitutive relation in an incremental form

$$\boldsymbol{\sigma}^{n+1} = L_\theta(\boldsymbol{\epsilon}^{n+1}) L_\theta(\boldsymbol{\epsilon}^{n+1})^\top (\boldsymbol{\epsilon}^{n+1} - \boldsymbol{\epsilon}^{n}) + \boldsymbol{\sigma}^{n}.$$

In practice, N sequential strain-stress data $\{(\boldsymbol{\epsilon}_i^1, \boldsymbol{\sigma}_i^1), \ldots, (\boldsymbol{\epsilon}_i^n, \boldsymbol{\sigma}_i^n)\}_{i=1}^N$ at n time steps can be used to learn the L_θ by solving the following problem:

$$\underset{\theta}{\arg\min}\, \mathcal{L}(\theta) = \sum_{i=1}^{N}\sum_{j=1}^{n} \Big[\boldsymbol{\sigma}_i^j - \big(L_\theta\big(\boldsymbol{\epsilon}_i^{j-1}\big) L_\theta\big(\boldsymbol{\epsilon}_i^{j-1}\big)^\top \big(\boldsymbol{\epsilon}_i^{j} - \boldsymbol{\epsilon}_i^{j-1}\big) + \boldsymbol{\sigma}_i^{j-1}\big)\Big]^2.$$

Based on experimental results, the SPD-NN has demonstrated itself as a more stable network architecture for learning constitutive relations, ensuring compliance with time consistency and the second-order work criterion.

In a subsequent work [33], we introduce a symmetric deep network architecture for approximating the first Piola-Kirchhoff stress (DPK). This framework allows for learning the constitutive relation with atomistic accuracy while preserving the inherent stress symmetry. Our focus lies in modeling the stress-deformation relationship for hyperelastic materials, where the first Piola-Kirchhoff stress $\mathbf{P}$ is defined as the derivative of strain energy density with respect to deformation $\mathbf{F}$. Adhering to the principles of material frame-indifference and material symmetry, this relation must satisfy two crucial constraints

$$\begin{aligned} &\mathbf{P}(\mathbf{QF}) = \mathbf{QP}(\mathbf{F}), && \forall \mathbf{Q} \in SO(3), \\ &\mathbf{P}(\mathbf{F}) = \mathbf{P}(\mathbf{FH})\mathbf{H}^\top, && \forall \mathbf{H} \in G, \end{aligned}$$

where $SO(3)$ stands for the proper orthogonal group and G refers to the material symmetry group.

To precisely enforce the aforementioned physical constraints within the constitutive network, we first review the definition of classical virial stress and validates its symmetry. A key observation gained from this analysis is that the symmetry of virial stress results from the intrinsic symmetry of potential function and crystal structure. Inspired by this, we design DPK as follows. For a material with crystallographic point symmetry group G_p, we first symmetrize a finite trainable weight vectors $\mathbf{W}=\{\mathbf{w}_1,\ldots,\mathbf{w}_N\}$ using the image set of G_p on $\mathbf{W}$

$$\tilde{\mathbf{W}}=\{\mathbf{H}_i\mathbf{w}_j \,|\, \mathbf{H}_i\in G_p, \mathbf{w}_j\in\mathbf{W}\}.$$

Obviously, $\tilde{\mathbf{W}}=\{\tilde{\mathbf{w}}_1,\ldots,\tilde{\mathbf{w}}_M\}$ forms an invariant subset of G_p: For any $\mathbf{H}\in G_p, \mathbf{H}\tilde{\mathbf{W}}=\tilde{\mathbf{W}}$. Based on this, we propose a symmetric network using the idea of harmonic approximation

$$\mathbf{P}_{\theta,\mathbf{W}}(\mathbf{F})=\mathbf{F}\sum_{i=1}^{M} g(\|\mathbf{F}\tilde{\mathbf{w}}_i\|;\theta)\tilde{\mathbf{w}}_i\otimes\tilde{\mathbf{w}}_i.$$

Here $g(\cdot;\theta)$ is a deep neural network function with parameter θ. By construction, this model can exactly satisfy the above two constraints, while its effectiveness and transferability have been numerically demonstrated.

In general, a physics-informed network, incorporating physical constraints into the data-driven model, is favored for stable and precise learning. Moreover, the integration of physical principles into the model design allows for training with reduced data size.

3.3 Production and evaluation of high-precision maps

Traditional high-precision map production relies on deterministic mechanisms in professional surveying and mapping. Using high-accuracy instruments and sophisticated surveying algorithms, it captures precise information such as road geometry and topographic features to construct the foundational framework of the map, which forms the core support at the mechanistic level. However, facing complex and changing urban environments, relying solely on traditional surveying methods encounters efficiency bottlenecks and cost challenges.

In crowdsourced mapping, a large amount of crowdsourced data from vehicle sensors, mobile devices, and other sources contains rich real-time road conditions and environmental information. Using machine learning algorithms, key features such as road changes and traffic sign updates can be automatically extracted from these massive, multi-source, and spatiotemporally heterogeneous data. For example, deep learning-based object detection algorithms can identify newly added traffic facilities or obstacles on roads, continuously optimizing the model's ability to recognize target objects in different scenarios through learning from large amounts of annotated data.

In the specific practice of high-precision map creation, we innovatively integrate the advantages of both mechanisms and data. In the point cloud data processing stage, on one hand, control point information obtained based on surveying mechanisms establishes initial map coordinate references and geometric models; on the other hand, machine learning algorithms are applied to analyze and process large-scale point cloud data. For instance, clustering analysis [27,80] and feature matching algorithms, combined with spatial distribution patterns and statistical features of data, achieve efficient classification, fusion, and error correction of point clouds, enhancing the accuracy of map element expression. In road topology modeling, by combining mechanism knowledge such as traffic flow theory with traffic congestion pattern recognition results based on big data analysis, a road network model that better conforms to actual traffic operation conditions is constructed, precisely reflecting the dynamic changes in road connectivity and capacity [74]. Through this deep integration, high-precision maps achieve qualitative leaps in accuracy, timeliness, and richness, powerfully promoting the development of fields such as autonomous driving and intelligent traffic management, laying a solid foundation for urban intelligence construction, and providing an extremely valuable example of this hybrid computing in complex geographic information system applications, demonstrating the enormous potential and broad prospects of cross-domain technology integration in solving complex practical problems.

4 Alternating optimization of mechanism-driven model and data-driven model

This third paradigm features a dynamic, reciprocal process where mechanism-based and data-driven models iteratively refine each other. This continuous loop progressively enhances predictive capability, with insights from one component guiding and improving the other, distinguishing it from approaches where data primarily tunes a fixed model or physics merely constrains data-driven construction.

Taking AlphaGo Zero as an example, in the Go game process, policy networks and value networks are constructed based on the rules of Go (mechanism knowledge) [62], providing the model with basic decision logic. Then, through self-play, large amounts of simulation data are continuously generated and used to iteratively optimize the networks. In each iteration, network parameters and structures are adjusted using techniques such as backpropagation algorithms based on the error between the model's predicted move strategies and game evaluations compared to the actual outcomes of the games. Simultaneously, the weights of mechanism knowledge and data in model decisions are adaptively adjusted according to error conditions, continuously enhancing the model's decision-making capabilities. This example illustrates how the alternating optimization of these hybrid components can effectively design high-quality strategies in competitive environments.

Similarly, in weather forecasting, this iterative optimization is crucial. As a typical prediction problem, atmospheric physics provides the foundational mechanism-based models for forecasting weather evolution. However, these complex models rely on initial conditions and parameters that are inherently uncertain and benefit from observational data. Data assimilation techniques represent a data-driven component, where vast amounts of real-world observations are used to correct and refine the state of the atmospheric model. This refined state then serves as a better starting point for the mechanism-based forecast. This process can be viewed as an iterative loop: the physics model makes predictions; observations provide data to correct or update the model's understanding, leading to improved subsequent forecasts. Over longer timescales, this process can also inform how future observations are assimilated or even guide structural improvements or re-parameterization of the physics model itself.

The subsequent subsections will explore these examples to illustrate how this alternating iterative optimization drives advancements in complex problem-solving and prediction.

4.1 From AlphaGo to AlphaZero

AlphaGo and AlphaZero are typical applications of data- and mechanism-driven hybrid computing in the field of artificial intelligence [61, 63], demonstrating how complex problem-solving can be achieved by combining data-driven methods with mechanism-based model optimization. The success of AlphaGo was not only due to the use of deep learning to learn from large-scale human Go game data, but also the incorporation of game rules and decision-making mechanisms via Monte Carlo tree search (MCTS), resulting in superhuman performance [66].

AlphaGo is primarily based on deep neural networks (DNN), extracting features from large quantities of human game records through supervised learning and reinforcement learning, representing a typical data-driven algorithm. Specifically, AlphaGo uses two neural networks: a policy network and a value network. The policy network is used to predict the probability of each move, mathematically expressed as

$$P(a\,|\,s) = \text{softmax}\big(f_\theta(s)\big),$$

where $P(a\,|\,s)$ represents the probability of selecting action a in state s, $f_\theta(s)$ is the output of the neural network, and θ represents the network parameters. The value network is used to evaluate the winning probability of the current board position, with its output expressed as

$$V(s) = g_\phi(s),$$

where $V(s)$ represents the expected winning rate of state $s, g_\phi(s)$ is the output of the value network, and ϕ represents the network parameters.

AlphaGo can also be viewed as a mechanism-based search algorithm, primarily manifested in MCTS, which evaluates the potential value of each move through simulated gameplay. The core idea of MCTS is to construct a search tree through multiple simulated games and select the optimal action based on the following formula:

$$a^* = \arg\max_a \left(Q(s,a) + c\sqrt{\frac{\ln N(s)}{N(s,a)}} \right),$$

where $Q(s,a)$ is the average return of action a, $N(s)$ is the number of visits to state s, $N(s,a)$ is the number of times action a has been taken, and c is the exploration parameter. MCTS gradually optimizes the search process by combining the outputs of the policy network and value network, ensuring that each decision both conforms to data-driven predictions and follows the game mechanisms of Go.

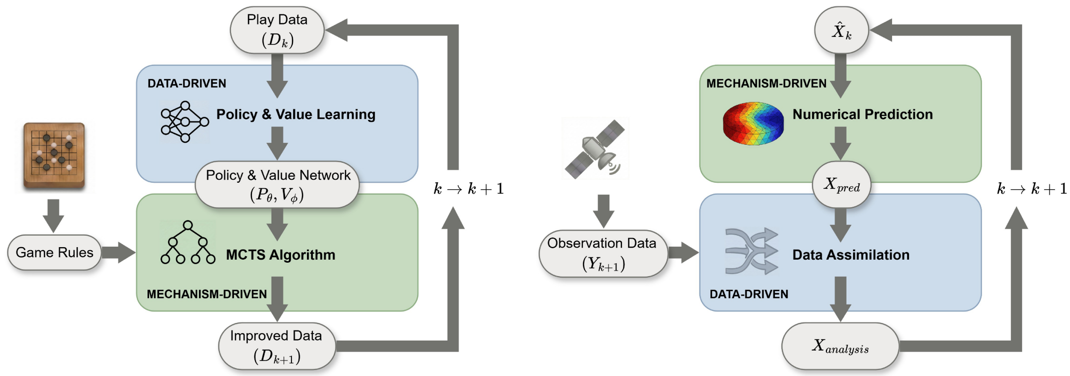
AlphaZero takes a step further based on AlphaGo, completely abandoning reliance on human game records and generating data solely through self-play. The core innovation of AlphaZero lies in deeply integrating deep neural networks with MCTS, forming a pure mechanism-data integrated computational framework. AlphaZero's neural networks not only learn the features of the game position but also continuously optimize policy and value functions through self-play. Its training process can be represented as the following optimization problem:

$$\min_{\theta,\phi} \mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(r+\gamma V_\phi(s') - V_\phi(s)\right)^2 + \lambda \|\nabla_\theta P_\theta(a|s)\|^2\right],$$

where $\mathcal{D}$ is the dataset generated by self-play, r is the immediate reward, γ is the discount factor, and λ is the regularization parameter. Through this alternating iterative optimization process, AlphaZero not only enhances the predictive capability of the model but also strengthens its generalization performance, enabling it to excel in other board games such as chess.

In the examples of AlphaGo and AlphaZero, the neural network, which outputs policy and value predictions, functions as the data-driven model, while the MCTS algorithm and the game rules form the mechanism-driven components. Their success exemplifies the core principles of hybrid computing, integrating data-driven and mechanism-driven approaches. While AlphaGo leverages data-driven models to enhance its mechanism-driven strategy, AlphaZero employs an iterative optimization process that alternates between data-driven and mechanism-driven models. Specifically, in each iteration of AlphaZero's reinforcement learning, we first leverage the mechanism-driven model improved by data-driven model (MCTS guided by the latest neural network) alongside the game rules to generate training data through self-play. These data are then used to update the data-driven model (the policy and value network). Through repeated iterations of this alternating process, a converged policy-value network is achieved, enabling the development of a refined game strategy (MCTS combined with the converged network). A schematic overview of this optimization cycle is presented in Fig. 2(a).

Such a integrated computational paradigm is not only applicable to game problems like Go but also provides important references for modeling and optimization of other complex systems. By combining data-driven methods with mechanism models, AlphaGo and AlphaZero demonstrate the enormous potential of data- and mechanism-driven hybrid computing in solving complex problems.



(a) AlphaZero (b) Weather prediction

Figure 2: Illustration of alternating optimization between data-driven and mechanism-driven models. Left: Alternating optimization in AlphaZero, combining data-driven policy/value learning with mechanism-driven MCTS. Right: Integration of mechanism-based prediction and data-driven assimilation in weather forecasting.

4.2 Integrated computing in weather forecasting

Traditional numerical weather prediction models are grounded in well-established physical principles such as atmospheric dynamics and thermodynamics [8, 53, 56, 72], with theoretical frameworks composed of equations like the Navier-Stokes equation, heat conduction equation, and water vapor equation to describe atmospheric motion, heat transfer, and water vapor transport processes. However, the atmospheric system is essentially a highly complex chaotic system with significant nonlinearity and uncertainty, involving multiscale interactions, complex terrain effects, turbulent motion, and many other complex factors. These complexities pose numerous challenges for traditional models in practical applications. For example, when simulating mesoscale and small-scale weather systems, due to limitations in model resolution and difficulties in physical process parameterization, it is often challenging to accurately capture the true evolution of atmospheric processes, resulting in certain degrees of deviation in forecast results.

With the rapid advances in modern meteorological observation technologies, multi-source observation methods such as satellite remote sensing, radar detection, and meteorological station observations provide massive data resources for meteorological research. These data contain rich atmospheric information, reflecting the actual state of the atmosphere from different spatiotemporal scales and perspectives. In this context, data assimilation technology becomes a key bridge connecting mechanism models with observational data. Let $(\mathbf{X}_k)_{k\in\mathbb{N}}$ and $(\mathbf{Y}_k)_{k\in\mathbb{N}}$ be a sequence of atmospheric states and the observation data, which satisfies the following dynamics model and observation model:

$$
\begin{aligned}
\mathbf{X}_{k+1} &= \mathcal{F}_k(\mathbf{X}_k, \mathbf{V}_k),\\
\mathbf{Y}_{k+1} &= \mathcal{G}_{k+1}(\mathbf{X}_{k+1}, \mathbf{W}_{k+1}).
\end{aligned}
$$

Here $k\in\mathbb{N}$ denotes the time index, $\mathcal{F}_k$ is a time-dependent operator corresponding to the underlying physical mechanism, $\mathcal{G}_{k+1}$ is a time-dependent observation model, $(\mathbf{V}_k)_{k\in\mathbb{N}}$

and $(\mathbf{W}_k)_{k\in\mathbb{N}}$ consist of independent random variables representing model uncertainty and measurement noise, respectively. At each time step k, NWP is carried out by solving the first forward problems numerically. However, due to model uncertainty and numerical errors, biases may arise in the estimation of $\mathbf{X}_{k+1}$. Data assimilation aims to enhance the estimation of $\mathbf{X}_{k+1}$ by optimally combining the observation data $\mathbf{Y}_{k+1}$ and the prediction of $\mathcal{F}_k$.

Statistically, the physical model $\mathcal{F}_k$ defines a non-homogeneous Markov process with transition probability density $p_{\mathbf{X}_{k+1}|\mathbf{X}_k}(\mathbf{x}|\mathbf{x}_k)$, and the observation model $\mathcal{G}_{k+1}$ introduces a measurement likelihood $p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}}(\mathbf{y}|\mathbf{x})$. The goal of data assimilation is then to estimate the posterior distribution of the atmospheric state $\mathbf{X}_{k+1}$ conditioned on all available observations $\mathbf{Y}_{[k+1]} := (\mathbf{Y}_i)_{i\le k+1}$, that is

$$\pi_{k+1}(\mathbf{x}|\mathbf{y}_{[k+1]}) := p_{\mathbf{X}_{k+1}|\mathbf{Y}_{[k+1]}}(\mathbf{x}|\mathbf{y}_{[k+1]}).$$

Given the previous posterior distribution $\pi_k(\cdot|\mathbf{y}_{[k]})$, the current state $\mathbf{X}_{k+1}$ can be predicted by utilizing the transition kernel, resulting in the following prior distribution of $\mathbf{X}_{k+1}$:

$$q_{k+1}(\mathbf{x}|\mathbf{y}_{[k]}) := \int p_{\mathbf{X}_{k+1}|\mathbf{X}_k}(\mathbf{x}|\mathbf{x}_k)\pi_k(\mathbf{x}_k|\mathbf{y}_{[k]})\mathrm{d}\mathbf{x}_k,$$

where the Chapman-Kolmogorov identity is applied. Using the Bayes formula, the posterior distribution $\pi_{k+1}(\mathbf{x}|\mathbf{y}_{[k+1]})$ can be expressed as the product of the measurement likelihood and the prior distribution

$$\pi_{k+1}(\mathbf{x}|\mathbf{y}_{[k+1]}) \propto p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}}(\mathbf{y}|\mathbf{x})q_{k+1}(\mathbf{x}|\mathbf{y}_{[k]}).$$

Therefore, the process of data assimilation can be achieved by recursively solving this Bayesian posterior sampling problem (from $\pi_k(\mathbf{x}|\mathbf{y}_{[k]})$ to $\pi_{k+1}(\mathbf{x}|\mathbf{y}_{[k+1]})$). With samples from π_{k+1}, one can derive the minimum mean square error estimation of $\mathbf{X}_{k+1}$ and quantify uncertainty by calculating the expectation and standard deviation, respectively. The workflow of this recursive process is illustrated in Fig. 2(b). The figure illustrates the iterative cycle of data assimilation for weather prediction. The process begins with a mechanism-driven prediction stage, where the state from the previous step, $\hat{\mathbf{X}}_k$ (approximation of $\mathbf{X}_k$), is advanced in time using a numerical solver for the governing dynamical equations to produce a forecast, $\mathbf{X}_{pred}$. This forecast is then corrected in a data-driven updating stage, where data assimilation techniques incorporate new observational data ($\mathbf{Y}_{k+1}$) to yield an improved analysis state, $\mathbf{X}_{analysis}$. Finally, this analysis state is fed back as the initial condition for the subsequent prediction cycle, ensuring the model is continuously refined.

Over the past decades, a range of methods has been developed for implementing data assimilation. For instance, particle filter methods approximate $\pi_k(\mathbf{x}|\mathbf{y}_{[k]})$ with a finite weighted sum of point-mass distributions; ensemble Kalman filters model the posterior density with a Gaussian distribution; 3D-Variational and 4D-Variational methods utilize maximum of posteriori estimation (MAP) for updateing the states. More recently, with the advancements in generative learning, diffusion models have also been applied to data assimilation, in which a deep neural network is employed to represent the score of prior or posterior distribution [20, 34].

With a precise approximation of the posterior density, data assimilation methods can effectively incorporate observational data into numerical prediction models based on Bayesian principles, dynamically adjusting and optimizing the model's initial conditions, boundary conditions, and key parameters [76]. For example, in regional heavy rainfall forecasting, utilizing satellite cloud imagery information such as cloud top temperature and height, combined with radar echo data reflecting precipitation particle intensity and distribution characteristics, along with meteorological station observations of surface pressure, temperature, humidity, and wind speed, significantly improves the forecast accuracy of rainfall system development, movement path, and intensity through data assimilation techniques optimizing numerical models. Similar to NWP, data assimilation also plays a critical role in the MLWP [8, 46]. Apart from enhancing the initial estimation of states, data assimilation aids in generating high-quality reanalysis data crucial for model training. According to the literature [11], certain MLWP models trained with such reanalysis data can deliver comparable and even higher prediction accuracy in an operational-like context, with lower computational cost and higher speed.

In recent years, the MOML algorithm developed by our research team has achieved a series of theoretical and application results in mechanism-data integration practices in weather forecasting [49]. The MOML algorithm integrates the advantages of various advanced machine learning algorithms including multiple linear regression (LR), random forest (RF), support vector regression (SVR), gradient boosting decision tree (GBDT), and extreme gradient boosting (XGBoost). In the data processing stage, scientific imputation and quality control methods are adopted to preprocess the missing values and anomalies commonly present in observational data, ensuring data reliability and validity. Simultaneously, feature selection and dimension reconstruction are performed on massive model data, observational data, and geographic information data based on atmospheric dynamics mechanisms and machine learning algorithms, significantly improving computational efficiency while ensuring forecast accuracy, laying a solid foundation for operational implementation. In practical applications, the MOML algorithm demonstrates significant advantages over traditional routine model output statistics (MOS) [26] and the European centre for medium-range weather forecasts (ECMWF) model in forecasting meteorological elements such as temperature, relative humidity, wind speed, and wind direction, with average forecast accuracy improved by more than 10%, powerfully demonstrating the enormous potential of data- and mechanism-driven hybrid computing in enhancing meteorological forecast precision. During the Beijing 2022 Winter Olympics meteorological services, the MOML algorithm significantly improved forecast accuracy by integrating high-frequency observational data and model forecast data, providing precise meteorological services for the event. Therefore, we believe that only by organically combining atmospheric physical mechanisms with multi-source data through advanced integrated computing technologies can we break through the limitations of traditional weather forecasting, achieve high-precision meteorological forecasts, and provide strong guarantees for the development of meteorological science and the stable operation of society and economy.

5 Conclusion

This review has explored the emerging paradigm of data- and mechanism-driven hybrid computing, which combines the interpretability and rigor of mechanism-based modeling with the flexibility and adaptability of data-driven approaches. We proposed a categorization of this hybrid landscape into three representative patterns of integration. Through a range of illustrative examples addressing different types of computational tasks, we demonstrated how the fusion of physical principles and data-driven techniques can significantly enhance model accuracy, generalizability, and computational efficiency.

However, this emerging paradigm faces several notable challenges and limitations. A primary limitation is the interpretability trade-off. Mechanism-driven models are valued for their transparency; however, this advantage is often diluted when they are coupled with "black-box" components like deep neural networks. The resulting hybrid model can obscure the underlying physical reasoning of a prediction, challenging its reliability in high-stakes applications. Another significant consideration is the computational cost. This includes not only the expense of running the coupled simulation but also the often-immense cost of generating and training with large, high-fidelity datasets. Furthermore, the performance of hybrid models remains sensitive to the quality and coverage of data, while ensuring their generalization to out-of-distribution scenarios, where novel physics may be encountered, is still an active and critical area of research.

Looking ahead, continued advances in machine learning and computational capabilities are expected to accelerate progress in this field [3,44]. Future research will likely focus on developing more sophisticated integration frameworks, establishing benchmarks for rigorous model comparison, and extending hybrid methods to new scientific frontiers. Ultimately, data- and mechanism-driven hybrid computing represents a fundamental shift in scientific methodology, with the potential to drive the next wave of discovery and innovation across science and engineering.

Acknowledgement

This work is supported by the Hubei Natural Science Foundation (Grant 2025AFA002), by the National Natural Science Foundation of China (Grants 12125103, U24A2002, and 12288101), and by the Fundamental Research Funds for the Central Universities.

References

[1] J. D. Anderson, *Computational Fluid Dynamics: The Basics With Applications*, McGraw-Hill, 1995.

[2] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, *Solving inverse problems using data-driven models*, Acta Numer., 28:1–174, 2019.

[3] R. E. Baker, J. M. Peña, J. Jayamohan, A. Jérusalem, *Mechanistic models versus machine learning, a fight worth fighting for the biological community?*, Biol. Lett., 14(5):20170660, 2018.

[4] A. R. Barron, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Trans. Inf. Theory, 39(3):930–945, 1993.

[5] M. I. Baskes, *Modified embedded-atom potentials for cubic materials and impurities*, Phys. Rev. B, 46(5):2727, 1992.

[6] I. Batatia et al., *A foundation model for atomistic materials chemistry*, arXiv:2401.00096, 2023.

[7] J. Behler and M. Parrinello, *Generalized neural-network representation of high-dimensional potential-energy surfaces*, Phys. Rev. Lett., 98(14):146401, 2007.

[8] K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, *Accurate medium-range global weather forecasting with 3D neural networks*, Nature, 619(7970):533–538, 2023.

[9] R. Bian et al., *Developing fatigue-resistant ferroelectrics using interlayer sliding switching*, Science, 385(6704):57–62, 2024.

[10] M. Born and R. Oppenheimer, *Zur Quantentheorie der Molekeln*, Ann. Phys., 389(20):457–484, 1927.

[11] Z. B. Bouallégue et al., *The rise of data-driven weather forecasting: A first statistical assessment of machine learning-based weather forecasts in an operational-like context*, Bull. Am. Meteorol. Soc., 105(6):E864–E883, 2024.

[12] S. Brenner and R. Scott, *The Mathematical Theory of Finite Element Methods*, in: Texts in Applied Mathematics, Springer Science & Business Media, 2007.

[13] J. A. Brotzge et al., *Challenges and opportunities in numerical weather prediction*, Bull. Am. Meteorol. Soc., 104(3):E698–E705, 2023.

[14] S. Cai, Z. Mao, Z. Wang, M. Yin, and G. E. Karniadakis, *Physics-informed neural networks (PINNs) for fluid mechanics: A review*, Acta Mech. Sin., 37(12):1727–1738, 2021.

[15] Z. Cai, J. Tang, S. Mukherjee, J. Li, C.-B. Schönlieb, and X. Zhang, *NF-ULA: Normalizing flow-based unadjusted Langevin algorithm for imaging inverse problems*, SIAM J. Imaging Sci., 17(2):820–860, 2024.

[16] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, Interscience Publishers, 1962.

[17] X. Daura, A. E. Mark, and W. F. Van Gunsteren, *Parametrization of aliphatic CHn united atoms of GROMOS96 force field*, J. Comput. Chem., 19(5):535–547, 1998.

[18] M. S. Daw and M. I. Baskes, *Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals*, Phys. Rev. B, 29(12):6443, 1984.

[19] B. Deng, P. Zhong, K. Jun, J. Riebesell, K. Han, C. J. Bartel, and G. Ceder, *CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling*, Nat. Mach. Intell., 5(9):1031–1041, 2023.

[20] Z. Ding, C. Duan, Y. Jiao, J. Z. Yang, C. Yuan, and P. Zhang, *Nonlinear assimilation via score-based sequential Langevin sampling*, arXiv:2411.13443, 2024.

[21] P. A. M. Dirac, *Quantum mechanics of many-electron systems*, Proc. R. Soc. Lond. Ser. A, Contain. Pap. Math. Phys. Character, 123(792):714–733, 1929.

[22] W. E, *The dawning of a new era in applied mathematics*, Notices Amer. Math. Soc., 68(4):565–571, 2021.

[23] M. R. Endsle, *Toward a theory of situation awareness in dynamic systems*, Hum. Factors, 37(1):32–64, 1995.

[24] M. Galib and D. T. Limmer, *Reactive uptake of N_2O_5 by atmospheric aerosol is dominated by interfacial processes*, Science, 371(6532):921–925, 2021.

[25] Z. Gao, L. Yan, and T. Zhou, *Failure-informed adaptive sampling for PINNs*, SIAM J. Sci. Comput., 45(4): A1971–A1994, 2023.

[26] H. R. Glahn and D. A. Lowry, *The use of model output statistics for forecasting*, Mon. Weather Rev., 100(6):449–462, 1972.

[27] F. Guo, Q. Ren, J. Tang, and Z. Li, *Dilated multi-scale fusion for point cloud classification and segmentation*, Multimed. Tools Appl., 81(6):6069–6090, 2022.

[28] Z. Guo, D. Lu, Y. Yan, S. Hu, R. Liu, G. Tan G, and W. Jia, *Extending the limit of molecular dynamics with ab initio accuracy to 10 billion atoms*, in: Proceedings of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, ACM, 205–218, 2022.

[29] J. Han, L. Zhang, R. Car, and W. E, *Deep potential: A general representation of a many-body potential energy surface*, Commun. Comput. Phys., 23(3):629–639, 2018.

[30] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, in: Springer Series in Statistics, Springer, 2009.

[31] G. E. Hinton et al., *Deep neural networks for acoustic modeling in speech recognition*, IEEE Signal Process. Mag., 29(6):82–97, 2012.

[32] P. Hohenberg and W. Kohn, *Inhomogeneous electron gas*, Phys. Rev., 136(3B):B864, 1964.

[33] T. Hu, J. Z. Yang, and C. Yuan, *DPK: Deep neural network approximation of the first Piola-Kirchhoff stress*, Adv. Appl. Math. Mech., 16(1):75–100, 2024.

[34] L. Huang et al., *DiffDA: A diffusion model for weather-scale data assimilation*, arXiv:2401.05932, 2024.

[35] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Comput. Methods Appl. Mech. Engrg., 194(39-41):4135–4195, 2005.

[36] H. Jeong, C. Batuwatta-Gamage, J. Bai, Y. M. Xie, C. Rathnayaka, Y. Zhou, and Y. Gu, *A complete physics-informed neural network-based framework for structural topology optimization*, Comput. Methods Appl. Mech. Engrg., 417:116401, 2023.

[37] W. Jia, H. Wang, M. Chen, D. Lu, L. Lin, R. Car, W. E, and L. Zhang, *Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning*, In: SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, 1–14, 2020.

[38] Y. Jiao et al., *Improved analysis of PINNs: Alleviate the CoD for compositional solutions*, Ann. Appl. Math., 39(3):239–263, 2023.

[39] Y. Jiao, Y. Lai, D. Li, X. Lu, F. Wang, Y. Wang, and J. Z. Yang, *A rate of convergence of physics informed neural networks for the linear second order elliptic PDEs*, Commun. Comput. Phys., 31(4):1272, 2022.

[40] Y. Jiao, D. Li, X. Lu, J. Z. Yang, and C. Yuan, *A Gaussian mixture distribution-based adaptive sampling method for physics-informed neural networks*, Eng. Appl. Artif. Intell., 135:108770, 2024.

[41] R. Jinnouchi, F. Karsai, and G. Kresse, *Making free-energy calculations routine: Combining first principles with machine learning*, Phys. Rev. B, 101(6):060201, 2020.

[42] W. L. Jorgensen and J. Tirado-Rives, *Potential energy functions for atomic-level simulations of water and organic and biomolecular systems*, Proc. Natl. Acad. Sci. USA, 102(19):6665–6670, 2005.

[43] E. Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge University Press, 2003.

[44] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, *Physics-informed machine learning*, Nat. Rev. Phys., 3(6):422–440, 2021.

[45] W. Kohn and L. J. Sham, *Self-consistent equations including exchange and correlation effects*, Phys. Rev., 140(4A):A1133, 1965.

[46] R. Lam et al., *Learning skillful medium-range global weather forecasting*, Science, 382(6677):1416–1421, 2023.

[47] R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra, *Bayesian imaging using Plug & Play priors: When Langevin meets Tweedie*, SIAM J. Imaging Sci., 15(2):701–737, 2022.

[48] Y. LeCun, Y. Bengio, and G. Hinton, *Deep learning*, Nature, 521(7553):436–444, 2015.

[49] H. Li, C. Yu, J. Xia, and P. Zhang, *A model output machine learning method for grid temperature forecasts in the Beijing area*, Adv. Atmos. Sci., 36:1156–1170, 2019.

[50] A. Merchant, S. Batzner, S. S. Schoenholz, M. Aykol, G. Cheon, and E. D. Cubuk, *Scaling deep learning for materials discovery*, Nature, 624:80–85, 2023.

[51] S. M. Moosavi, K. M. Jablonka, and B. Smit, *The role of machine learning in the understanding and design of materials*, J. Am. Chem. Soc., 143(5):1787–1799, 2021.

[52] D. V. Patel, D. Ray, and A. A. Oberai, *Solution of physics-based bayesian inverse problems with deep generative priors*, Comput. Methods Appl. Mech. Engrg., 400:115428, 2022.

[53] J. Pathak et al., *FourCastNet: A global data-driven high-resolution weather model using adaptive Fourier neural operators*, arXiv:2202.11214, 2022.

[54] V. Purohit, M. Repasky, J. Lu, Q. Qiu, Y. Xie, and X. Cheng, *Posterior sampling via Langevin dynamics based on generative priors*, arXiv:2410.02078, 2024.

[55] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, J. Comput. Phys., 378:686–707, 2019.

[56] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, and Prabhat, *Deep learning and process understanding for data-driven Earth system science*, Nature, 566(7743):195–204, 2019.

[57] E. Schrödinger, *Quantisierung als Eigenwertproblem*, Ann. Phys., 386(18):109–139, 1926.

[58] F. Schroff, D. Kalenichenko, and J. Philbin, *A unified embedding for face recognition and clustering*, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 815–823, 2015.

[59] K. Schütt, P.-J. Kindermans, H. E. Sauceda Felix, S. Chmiela, A. Tkatchenko, and K.-R. Müller, *SchNet: A continuous-filter convolutional neural network for modeling quantum interactions*, in: Advances in Neural Information Processing Systems, Curran Associates, Inc., 30:992–1002, 2017.

[60] Y. Shin, J. Darbon, and G. E. Karniadakis, *On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type PDEs*, arXiv:2004.01806, 2020.

[61] D. Silver et al., *Mastering the game of Go with deep neural networks and tree search*, Nature, 529(7587):484–489, 2016.

[62] D. Silver et al., *Mastering the game of Go without human knowledge*, Nature, 550(7676):354–359, 2017.

[63] D. Silver et al., *Mastering chess and shogi by self-play with a general reinforcement learning algorithm*, arXiv:1712.01815, 2017.

[64] J. S. Smith, B. Nebgen, M. Lubbers, O. Isayev, and A. E. Roitberg, *Less is more: Sampling chemical space with active learning*, J. Chem. Phys., 148(24):241733, 2018.

[65] G. Strang, *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, 1986.

[66] G. Tesauro, V. T. Rajan, and R. Segal, *Bayesian inference in Monte-Carlo tree search*, in: Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI 2010), AUAI Press, 580–588, 2010.

[67] H. Wang, L, Zhang, J. Han, and W. E, *DeePMD-kit: A deep learning package for many-body potential energy representation and molecular dynamics*, Comput. Phys. Commun., 228:178–184, 2018.

[68] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case, *Development and testing of a general AMBER force field*, J. Comput. Chem., 25(9):1157–1174, 2004.

[69] L. Wang, G. Liu, G. Wang and K. Zhang, *M-PINN: A mesh-based physics-informed neural network for linear elastic problems in solid mechanics*, Internat. J. Numer. Methods Engrg., 125(9):e7444, 2024.

[70] Y. Wang and C. Y. Lai, *Multi-stage neural networks: Function approximator of machine precision*, J. Comput. Phys., 504:112865, 2024.

[71] Z. Wang et al., *Concurrent learning scheme for crystal structure prediction*, Phys. Rev. B, 109(9): 094117, 2024.

[72] J. A. Weyn, D. R. Durran, and R. Caruana, *Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere*, J. Adv. Model. Earth Syst., 12(9): e2020MS002109, 2020.

[73] K. Xu, D. Z. Huang, and E. Darve, *Learning constitutive relations using symmetric positive definite neural networks*, J. Comput. Phys., 428:110072, 2021.

[74] S. Yang, J. Wu, G. Qi, K. Tian, and S. Li, *Analysis of traffic state variation patterns for urban road network based on spectral clustering*, Adv. Sci. Technol. Eng. Syst. J., 2(6):379–387, 2017.

[75] D. Zhang, H. Bi, F.-Z. Dai, W. Jiang, X. Liu, L. Zhang, and H. Wang, *Pretraining of attention-based deep learning potential model for molecular simulation*, Npj Comput. Mater., 10:94, 2024.

[76] F. Zhang, M. Minamide, and E. E. Clothiaux, *Potential impacts of assimilating all-sky infrared satellite radiances from GOES-R on convection-permitting analysis and prediction of tropical cyclones*, Geophys. Res. Lett., 43(6):2954–2963, 2016.

[77] L. Zhang, J. Han, H. Wang, R. Car, W. E, *Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics*, Phys. Rev. Lett., 120(14):143001, 2018.

[78] L. Zhang, D.-Y. Lin, H. Wang, R. Car, and W. E, *Active learning of uniformly accurate interatomic potentials for materials simulation*, Phys. Rev. Mater., 3(2):023804, 2019.

[79] L. Zhang, H. Wang, R. Car, and W. E, *Phase diagram of a deep potential water model*, Phys. Rev. Lett., 126(23):236001, 2021.

[80] Q. Zhang, J. Cheng, S. Wang, C. Xu, and X. Gao, *Point-selection and multi-level-point-feature fusion-based 3D point cloud classification*, Electron. Lett., 55(24):1344–1346, 2019.

[81] M. S. Zhdanov, *Geophysical Inverse Theory and Regularization Problems*, Elsevier, 2002.

[82] O. C. Zienkiewicz, *The Finite Element Method*, McGraw-Hill, 1977.