Position: Significant impact of numerical precision in scientific machine learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

The machine learning community has focused on computational efficiency, often leveraging lower-precision formats such as FP16, instead of the standard FP32. In contrast, little attention has been paid to higher-precision formats, such as FP64, despite their critical role in scientific domains like materials science, where even small numerical differences can lead to significant inaccuracies in physicochemical properties. This need for high precision extends to the emerging field of *machine learning for scientific tasks*, yet it has not been thoroughly investigated. According to several studies and our toy experiment, models trained with FP32 show insufficient accuracy compared to those trained with FP64, indicating that higher precision is also crucial in scientific machine learning, as in traditional scientific computing. This precision issue limits the potential of scientific machine learning that can replace the traditional scientific computings in practical research. Our position paper not only highlights these precision-related issues but also recommends reporting comparisons between FP32 and FP64 results, encouraging the release of FP64 models. We believe that these efforts can enable machine learning to contribute meaningfully to the natural sciences, ensuring both scientific reliability and practical applicability.

1 Introduction

The rapid advancements in natural language processing (NLP) and computer vision (CV) in the field of machine learning (ML) have accelerated the broad application across various domains [56, 73, 83, 51, 81]. Specifically, **ML for scientific tasks**–*which has begun to resolve intellectually demanding problems in scientific fields*–has been highlighted across disciplines, opening new possibilities for scientific breakthroughs. In recognition of these breakthroughs, the 2024 Nobel Prize in Chemistry honored the contributions of scientific ML, highlighting innovations such as AlphaFold and RoseTTAFold [39, 4, 99]. These models transformed scientific research by rapidly delivering results that once required significant resources and time-consuming experiments or simulations. Building on these successes, scientific ML not only addresses traditional labor-intensive workflows but also finds hidden patterns within complex data, thereby providing human researchers with direct insights into novel discoveries across natural sciences [109, 67, 41, 117, 97, 116].

In the context of methodology, the development of scientific ML naturally follows the broader trends and paradigms of the ML research field. In the early stages of NLP and CV, most work focused on discriminative tasks (*e.g.*, named entity recognition and image classification) [107, 24] before gradually shifting to generative tasks (*e.g.*, machine translation and text-to-image generation) [18, 89]. Further, generative approaches have advanced sequentially, moving from variational autoencoders (VAEs) to generative adversarial networks (GANs), and more recently, to diffusion models [43, 32, 35, 95]. In a similar manner, numerous scientific domains have rapidly adopted the latest advances from

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

the ML community. For example, among various areas of bioinformatics, research on DNA sequence data initially leveraged discriminative models such as DeepVariant [78] and DeepSEA [122], and over time, the trends moved to generative models including ExpressionGAN [124] and Evo [71]. Similarly, material structure prediction in the field of materials and drug discovery has followed this trend from VAEs [85, 31, 55] and GANs [79, 42, 1] to diffusion models [36, 75, 118].

In parallel with these advances, the scaling law, one of the most recent paradigm in ML research, emphasizes performance enhancement by progressively increasing the size of models, training datasets, and computational resources [40, 94]. Building upon this idea of incremental scale expansion, researchers have successfully tested the approach of *bigger is better* across diverse fields, including NLP, CV, reinforcement learning, and time-series forecasting [119, 22, 34, 70, 92]. Following this pattern, scientific ML is also adopting this paradigm, and in fact, large models designed to address scientific tasks have already begun to appear [71, 120].

As these models grow larger and more complex, they require massive computational resources, presenting significant challenges for both training and inference processess. To address this, the lower numerical precision and quantization are a widely employed strategy, which helps reduce the computational expense [123, 65]. These approaches inevitably involve a trade-off between fidelity and resource efficiency, typically resulting in some accuracy degradation. To minimize such precision-related losses, techniques such as mixed precision training [65] and sophisticated quantization methods [7, 25, 58, 112] have been proposed, which allow researchers to conserve the original accuracy while achieving the advantages of reduced computational costs. Consequently, the ML community has accepted slight accuracy degradation as a natural trade-off for greater efficiency, thereby integrating these lower-precision techniques into real-world applications to balance performance and computational burden.

However, the tolerance for lower-precision techniques raises substantial concerns in the field of scientific computing. Scientific computing primarily aims to solve fundamental physics equations that are difficult to solve manually by simplifying or discretizing the inherently continuous and infinite real-world phenomena to make them computationally tractable. As a consequence, even tiny differences in numerical precision can lead to significant issues regarding the reliability of computational results. Our experimental findings demonstrate that *single precision's sensitivity to numerical deviations can substantially influence the accuracy of fundamental physical equations*. As a result of this high sensitivity, small numerical differences can cause significant changes in physicochemical properties, such as absorption coefficient, defect energies, or reaction pathway predictions, thereby reducing the reliability of results, especially when accurate predictions are crucial for critical decisions. One critical aspect is that these challenges related to numerical precision are not confined to traditional computational science, as ML models are increasingly being utilized in various studies to replace prevalent simulations. In other words, traditional computational science requires high precision, making it essential to verify whether FP32 produces valid results before using ML models, as numerical precision is key to maintaining reliability.

In this position paper, we argue for the significant role of numerical precision in scientific ML research, emphasizing the need for evaluating and analyzing its impact on results derived from varying precision levels. To this end, we first highlight real-world examples from established computational simulations where numerical precision directly impacts on their results. We then explain that the importance of numerical precision is not confined to traditional scientific computing alone but is also deeply related to ML applications in scientific domains. Specifically, we provide examples involving ML potential models and physics-informed neural networks (PINNs), which are actively studied in both ML and science domains, demonstrating the critical role of numerical precision in these areas [82, 44, 50]. Additionally, we explore the implications of large language models (LLMs) in scientific ML on precision-related considerations.

In conclusion, we present concrete recommendations for the ML community and potential research directions based on our discussions. We then provide alternative viewpoints to our position, offer responses, and conclude. Since the main role of ML in scientific research is to deepen understanding in traditional domains, the issues we raise must be rigorously examined. When relatively simple actions by ML researchers can remove barriers that hinder natural scientists from applying ML models, these measures become essential, not optional. As scientific machine learning is still in its early stages, we hope that thorough debate will help minimize trial-and-error in future research.

2 Importance of numerical precision in scientific computing

The main goal of scientific computing is solving complex physics equations through computational power, especially when manual solutions are impractical or nonexistent. Specifically, many-body problems including multiple object interactions demonstrate the necessity of high-performance computing. Accordingly, various computational methods have emerged to solve fundamental physics equations: molecular dynamics for Newton's Second Law, density functional theory (DFT) [38] for the Schrödinger equation, and the finite-difference time-domain (FDTD) [115] method for Maxwell's equations. Despite the algorithmic progress outlined above, the fidelity of these simulations is bounded by how continuous physical variables are encoded on digital hardware. Modern digital processors represent real numbers as finite-length bit strings, so continuous equations—ranging from $F = ma$ to the Schrödinger and Maxwell formulations—cannot be solved exactly. To bridge this gap, scientists adopt controllable approximations: reformulating the problem (*e.g.*, the Kohn–Sham equation [45]) or discretising time and space (*e.g.*, molecular dynamics). These methods remain trustworthy only when round-off error is tightly bounded, making double-precision arithmetic the *de-facto* compromise between cost and accuracy. For instance, Quantum ESPRESSO [29], a leading open-source DFT implementation, strictly enforces double precision throughout its code.

To demonstrate the precision's crucial role in scientific computing, we present examples showing how small numerical variations can significantly impact computational results, analyzing these effects in realistic research scenarios. Specifically, we illustrate the influence on materials research scenarios, thereby analyzing the implications and identifying the precise numerical accuracy-related challenges.

2.1 Impact on density functional theory simulation

Quantum mechanics, beginning with Planck's quantum hypothesis [76], revolutionized our understanding of microscopic phenomena. While exact calculations are only possible for simple systems like the hydrogen atom, the Kohn-Sham equation introduced DFT as an efficient approach for many-body electron problems. Using Python-based Simulations of Chemistry Framework (PySCF) [98], we performed geometry optimization calculations for water (H_2O) using both Hartree-Fock (HF) and DFT calculations with B3LYP functional and 6-311++G(d,p) basis set [2, 102, 13, 113].

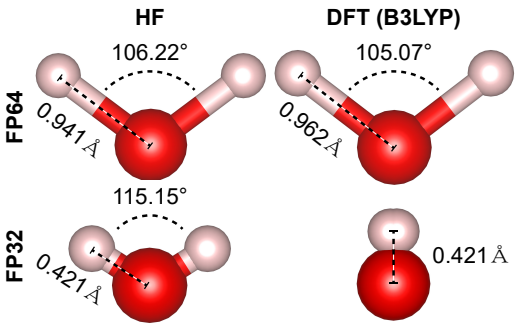

Figure 1: Geometry optimizations of a water molecule using FP64 (top) and FP32 (bottom) with HF (left) and DFT (right) methods. FP64 computations yield physically valid structures, whereas FP32 leads to unrealistic geometries.

Figure 1 shows the results of geometry-optimized water molecules obtained from HF and DFT calculations under FP32 and FP64 numerical precision conditions. When utilizing FP64, both HF and DFT calculations successfully converged within three optimization steps with satisfying the convergence criteria. Since DFT explicitly accounts for electron correlation effects [12], it is generally expected to provide more accurate results than HF, a trend that is also reflected in our findings. Comparing bond lengths, the reference [19] O-H bond length is 0.957 Å, while HF exhibits a deviation of 0.016 Å (1.7 % error), and DFT yields a smaller deviation of 0.005 Å (0.5 % error). Similarly, for the bond angle, HF deviates by $1.7°$ (0.7 % error) from the reference value of $104.52°$, whereas DFT shows a smaller deviation of $0.55°$ (0.5 % error). However, when using FP32, significant numerical instabilities arise, preventing the convergence of optimization steps. In the case of HF calculations, the gradient of hydrogen atoms stagnates between 0.2–0.4 Ha/Bohr, which is significantly above the desired convergence threshold of 10^{-6} Ha/Bohr. For DFT calculations, the issue becomes even more pronounced, as the gradient values rapidly diverge beyond 10^5 Ha/Bohr, resulting in termination before reaching the maximum step. As a result, when using FP32, the HF calculation exhibits a substantial 50 % error, while the DFT calculation produces a molecular structure impossible to exist in reality, as illustrated in Figure 1. A detailed examination using the atomic coordinates is shown in Appendix B.1.

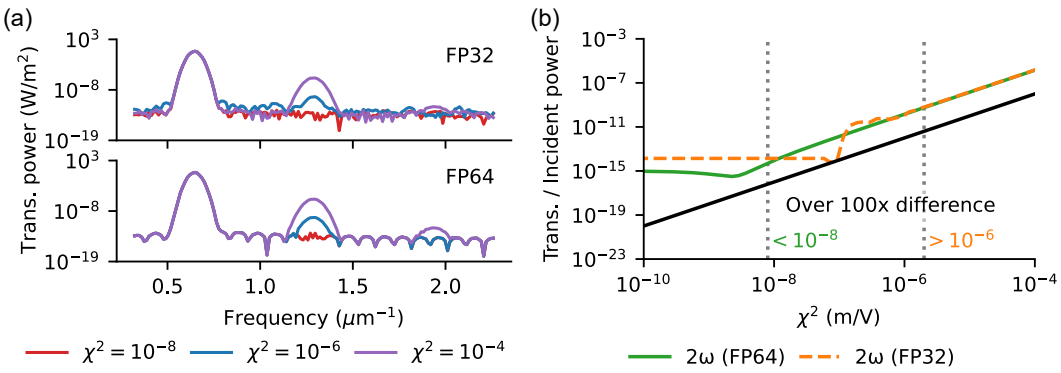

Figure 2: (a) Transmittance spectra comparison between FP32 (top) and FP64 (bottom) in Kerr media, showing FP32's failure to accurately model higher harmonics and low-power wave patterns below 10^{-10} W/m^2. (b) Computed second harmonic susceptibility shown in FP64 (green solid) and FP32 lines (orange dashed) compared to theoretical quadratic behavior (black). FP64 maintains accuracy to 10^{-8} m/V, while FP32 deviates above 10^{-6} m/V, making it unsuitable for typical nonlinear materials.

2.2 Impact on finite difference time domain simulation

Electromagnetism, established by Maxwell's equations [64, 63], provides the theoretical foundation for understanding electromagnetic waves. However, solving Maxwell's equations for complex phenomena is computationally challenging. To address this, FDTD discretizes Maxwell's equations in time and space. Using Meep [72], an open-source FDTD software, we investigated numerical precision effects on electromagnetic simulations, comparing FP32 and FP64 in nonlinear Kerr media simulations. We simulated a Kerr medium (refractive index$=1.65$) excited by an electromagnetic wave source ($\lambda = 1.55$ μm, $\Delta\lambda = 0.15$ μm).

Figure 2 (a) presents the transmission spectrum of the nonlinear Kerr medium under FP32 and FP64 precision settings. From left to right, the spectral peaks correspond to the fundamental generation induced by the source, the second harmonic generation (SHG), and the third harmonic generation (THG). While the fundamental peak exhibits minimal differences between FP32 and FP64, notable discrepancies arise in the SHG and THG regions. Specifically, FP32 calculations display pronounced background signal instability and intensity variations in harmonic generation, which result from imprecise numerical computation. A particularly notable difference appears in the behavior of the background signal. In FP64 calculations, the background follows a well-defined periodic pattern governed by the electromagnetic wave, whereas in FP32, the background signal appears as unstructured Gaussian-like noise. This phenomenon indicates that the lack of numerical precision in FP32 significantly disrupts the accurate computation of low-intensity transmitted power, particularly for electromagnetic waves in the range of 10^{-11} W/m^2. These findings highlight the fundamental limitations of single precision in reliably capturing weak electromagnetic signals and nonlinear optical effects.

To further analyze the impact of numerical precision, we examined the relationship between second-order nonlinear susceptibility (χ^2) and the transmittance-to-incident power ratio. As shown in Figure 2 (b), the black upward-sloping line represents a quadratic line, serving as a reference line indicating the expected computational trend of transmittance over incident power ratio as nonlinear susceptibility varies. Ideally, the computationally simulated values should align with this reference trend, maintaining the same slope. Comparing the results obtained from FP64 (green solid line) and FP32 (orange dashed line), we observe that as nonlinear susceptibility decreases beyond a certain threshold, the ratio begins to saturate. This saturation point effectively defines the lower bound of computational precision achievable under each numerical setting.

Specifically, for values of χ^2 above 10^{-6}, both FP64 and FP32 provide reliable computational precision. However, for values below this threshold, FP32 results begin to exhibit saturation, rendering further calculations meaningless due to the loss of numerical resolution. In contrast, FP64 maintains simulation accuracy down to approximately 10^{-8}, demonstrating a computational precision that is at least two orders of magnitude higher than that of FP32. This result implies that for most nonlinear materials with χ^2 values below 10^{-6}, transmittance spectrum simulations using FP32

become inherently unreliable. These findings highlight the critical role of numerical precision in computational science, particularly in fields where small numerical deviations can lead to substantial errors. As demonstrated in both DFT and FDTD simulations, the limitations of single precision introduce significant inaccuracies, especially in cases involving highly sensitive physical properties. This also highlights the necessity of carefully selecting numerical precision levels when conducting computational simulations, particularly in scientific machine-learning applications where maintaining the reliability of results is essential. While FP32 is adequate for various routine or weakly nonlinear calculations, our benchmarks show that in strongly nonlinear regimes it can introduce critical artifacts; the exact thresholds and representative case studies are provided in Appendix B.2.

3 Numerical precision issue in scientific ML

As demonstrated in the previous section, numerical precision can significantly affect the outcomes of traditional scientific simulations and potentially influence the results of scientific research. This naturally leads to an important question: **Do ML models designed for scientific tasks also suffer from similar precision-related issues?** To investigate this question, we survey various studies that apply ML to scientific research, searching for cases where the precision issue has been reported. We also conduct simple toy experiments to further assess the impact of numerical precision in ML-based scientific tasks. Through these analyses, we seek to determine whether the precision issue is a *significant challenge* or *just a theoretical concern*.

3.1 Impact on machine learning potential

The first example we present is an ML potential[1] [44, 50], which is closely related to Section 2.1. Fundamentally, ML potential models aim to compute potential energy and the associated forces for a given material structure, offering a much faster alternative to traditional quantum mechanical calculations. Due to their wide range of applications, ML potentials have been extensively studied not only in physics, materials science, chemistry, and biology but also within the ML community [14, 80, 93, 30, 90, 10, 9]. In addition, property prediction and generation for material or drug discovery have also been actively explored, making ML potentials a familiar subject for ML researchers. In our position paper, we focus specifically on neural network potentials, a class of ML potentials built on neural architectures. Since ML research often treats energy and force values in the same manner as other material properties, our discussion extends naturally to broader property prediction tasks.

A key challenge in ML potential studies lies in effectively representing and processing atomic information in three-dimensional space while ensuring rotational and translational equivariance or invariance. To tackle this, the field has evolved from vanilla graph neural networks [88] and transformers [106] to more specialized architectures that satisfy these constraints, achieving higher prediction accuracy [90, 28, 86, 10, 9, 27, 100, 54, 26]. As a result, many recent models are now integrated into widely used libraries or simulation software, such as the Atomic Simulation Environment (ASE) [5, 52] and LAMMPS [77, 101]. This demonstrates that ML potential models are increasingly employed in practical research; thus, any numerical precision issues arising in these models could have significant implications for scientific discoveries.

Consequently, we aimed to investigate whether existing ML potential models suffer from precision issues. To this end, we surveyed the pretrained checkpoints of various ML potential models available in the ASE library to determine whether they support FP64 precision. Interestingly, among several models, only MACE [9] provides pretrained checkpoints trained in FP64, while other models appear not to have considered FP64 training. Even before detailed analysis, this observation suggests that the ML potential community may not be fully aware of the potential significance of numerical precision.

To preliminarily understand the effect of precision, we conducted two toy experiment using MACE, the only model that provides FP64-trained parameters. In the first experiment, we examined the impact of numerical precision on the accuracy of potential energy surface (PES) reconstruction. We selected an ethanol molecule as a representative small organic system containing multiple atom types (C, H, and O). Then, by moving one of its carbon atoms along a certain path, we observed changes in the potential energy and forces. We compared FP32 and FP64 predictions by applying the built-in type conversion in the MACE code to the FP64-trained checkpoint. The results indicate that the

[1] In other domains, the term *machine learning interatomic potential* (MLIP) is also used.

differences between FP32 and FP64 remain within approximately 1 meV for energy and 0.02 meV/Å for force, which are margins typically considered acceptable in small-molecule simulations. Further experimental details can be found in Appendix B.3 and Figure 4.

Afterwards, we turned on a more advanced task, predicting vibrational properties, which contains richer information than the PES itself. To make the setting more realistic, we computed the vibrational modes of the *oseltamivir* molecule, which is the active ingredient of the anti-influenza drug *Tamiflu*. Table 2 summarize a subset of the calculated vibrational-mode frequencies; for modes 10 and 11, the discrepancies reach about 1.7 and 1.4 cm^{-1}, respectively. The differences up to 1.7 cm^{-1} seems not significant, but this value may affect huge influence when analyzing the vibrational mode from the measured data. In general, the spectral resolution of Raman and infrared spectroscopy instruments, which generally utilized to measure the vibrational properties of materials, is varied from few cm^{-1} for 10,000-50,000 USD to sub cm^{-1} for over 100,000 USD. These comparable spectral resolutions of real-world instruments suggest that researchers may encounter ambiguous cases when interpreting marginal values of certain vibrational modes. For instance, consider a scenario where a researcher obtain a measured vibrational frequency of 78.2 cm^{-1} for the oseltamivir molecule. Which vibrational mode should be assigned to this value? Calculations performed using FP32 precision would suggest mode 10, with a difference of only 0.36 cm^{-1}. Conversely, FP64 precision calculations would favor assignment to mode 11, despite the slightly larger difference of 0.59 cm^{-1}.

The aforementioned results suggest that while FP32 calculations may suffice for tasks such as molecular dynamics simulations based solely on PES, they may fall short when predicting more sensitive physical properties such as vibrational spectra. This emphasizes the importance of task-specific evaluation and precision-aware analysis, particularly when moving beyond PES-level predictions toward richer, experimentally comparable observables.

Nevertheless, our experimental framework was intentionally simplified, and these findings should not be overinterpreted as definitive evidence regarding machine learning models' sensitivity or insensitivity to numerical precision variations. Furthermore, the FP32 model evaluated in this study was initially trained using FP64 precision and subsequently converted to FP32 for inference purposes. A model trained exclusively in FP32 from initialization could exhibit different behavior. In fact, Batatia et al. [8] report that NequIP [10] exhibits different numerical sensitivity when trained in FP32 versus FP64, and Maxson et al. [62] also discuss similar issues. These observations highlight the importance of carefully assessing numerical precision in ML potential models and the need for systematic benchmarks regarding precision.

3.2 Impact on physics-informed neural network

Beyond the fundamental equations mentioned in the previous section, various subfields of natural science describe natural phenomena using differential equations. For example, in fluid dynamics, including weather prediction, Navier-Stokes, continuity, and heat transfer equations are used [105, 11]. Moreover, differential equations such as the Black-Scholes equation [17] are also employed in fields beyond natural sciences, such as financial engineering. Many of these equations either lack general analytical solutions or are too complex to be solved manually. As a result, numerical methods have been developed over time, leading to techniques such as the Euler method, Runge-Kutta methods, and Picard method [20, 96]. These techniques have also influenced modern approaches in ML, including diffusion models, NeuralODEs, and deep equilibrium models (DEQs) [35, 95, 21, 6].

The concept of the PINNs [82] leverages automatic differentiation (autograd), fundamental to backpropagation, to solve differential equations using neural networks. Due to its simple yet powerful approach, PINNs have been widely adopted in scientific domains that rely on numerical methods. This section explores whether numerical precision issues also arise in PINNs and investigates related challenges through a literature survey.

First, Nakamura et al. [68] explicitly discussed the impact of numerical precision in scientific research, reporting that training PINNs with FP32 failed, whereas FP64 did not: *from a comprehensive standpoint, FP32 computation has a risk of failure for the present problem compared with FP64.* This work applies PINNs to a specific fluid dynamics problem involving surface tension modeling, which requires up to fourth-order derivatives, making it a specialized case of differential equations. Although this is a specific scenario, it is a real-world scientific study, demonstrating that precision issues can significantly impact the practical use of PINNs.

Meanwhile, Sharma and Shankar [91] were well aware of precision issues and leveraged this understanding to improve the methodology of PINNs. The key idea of their work is to replace certain autograd operations in PINNs with a specialized finite difference method, reducing the computational cost associated with autograd. Here, to compensate for the loss of accuracy introduced by finite difference approximations, the authors proposed using high-precision (FP64) training. As a result, the reduction in computational cost from bypassing autograd exceeds the overhead introduced by FP64 operations, leading to an overall speedup that makes their approach faster than a vanilla PINN in FP32. Beyond the fields of PINNs and scientific ML, this study introduces a novel perspective on utilizing high-precision models in neural network research.

Thus, in the context of PINNs, a comprehensive study is needed to systematically assess the impact of numerical precision issues on scientific research. Fortunately, many fields share similar types of differential equations, *e.g.*, Laplace equation in electrostatics and fluid dynamics, where it describes electric potential distribution and velocity potential in inviscid flow, respectively. By focusing on the precision challenges of commonly used differential equations and rigorously validating PINNs in this context, such research could have a substantial impact across multiple domains.

3.3 Challenges for large language models

The emergence of LLMs in scientific applications is accelerating, further raising concerns about numerical precision in such domains. To investigate these concerns, we examine both existing studies and empirical evidence that highlight precision-related challenges in LLM applications. The integration of LLMs in scientific domains follows two distinct approaches. The first involves direct inference without architectural modifications, where scientific data is transformed into natural language format for existing LLM architectures [84, 37, 59]. The second approach develops specialized architectures that combine domain-specific encoders with fine-tuned language models, preserving the intrinsic properties of scientific data while leveraging LLM capabilities [53, 74].

Regarding the first approach, unlike conventional scientific models, LLMs generate outputs based on tokens, which may compromise prediction accuracy. Numerous studies have demonstrated that LLMs struggle with symbolic tasks [110, 114], similar to their difficulties in numerical predictions. For instance, these models often fail to accurately count the occurrences of specific characters within words (*e.g.*, counting the letter 'r' in 'strawberry') or comparing the size of decimal numbers (*e.g.*, determining whether 3.9 is larger than 3.11[2]). This limitation stems from their fundamental architecture, where words are processed as sequences of tokens rather than as individual alphabetic characters or numbers. Although various studies [110, 46, 114, 15] have been proposed to address these challenges, symbolic manipulation remains a significant obstacle for LLMs. Consequently, their application in scientific tasks requires careful consideration and validation.

Another critical consideration in LLM deployment is the continuous increase in model size. For instance, the open-source Llama series demonstrates this trend clearly: LLaMA (65B parameters) grew to Llama-2 (70B) and further to Llama-3.1 (405B) [103, 104, 60], and more recently, DeepSeek-v3 has pushed this expansion even further, reaching 671B [23]. Such explosive growth in model sizes across LLMs has resulted in a substantial increase in computational costs for both training and inference. To mitigate the budget, researchers commonly employ parameter quantization techniques by reducing model precision to lower-bit formats [57, 25, 58], sometimes even 1-bit representations [112].

However, these optimization strategies fundamentally conflict with the stringent precision requirements of scientific computing applications, as emphasized throughout our analysis. This issue is particularly critical for the second approach, where domain-specific encoders, which are often derived from scientific ML models, serve as feature extractors. If quantization significantly reduces the precision of the extracted features, the LLM may fail to process them accurately, potentially degrading overall model performance. For example, Li et al. [53] employed UniMol [121], a model broadly categorized as an ML potential, as an encoder. Even if the encoder provides highly precise features, the LLM's lower precision representations may obscure this information, leading to inaccurate final predictions. This inherent trade-off between computational efficiency and numerical precision highlights the necessity of careful consideration when integrating LLM into scientific applications.

[2]Recent large language models exhibit systematic errors in decimal comparison due to tokenization artifacts. When comparing 3.9 and 3.11, models tokenize these as ['3', '.', '9'] and ['3', '.', '11'] respectively, leading to incorrect digit-wise comparison (9 vs. 11) rather than proper decimal evaluation. As of early 2025, while GPT-4 has resolved this specific case, Claude 3.7 continues to incorrectly identify 3.11 as larger than 3.9.

4 Suggestions for Advancing Scientific ML

Building upon previous discussions, we present key suggestions for the scientific ML community.

Benchmarking and reporting FP32 vs. FP64 results Scientific ML typically necessitate higher precision than general ML tasks to ensure reliability. While predictive accuracy is the primary focus, other factors such as training time, inference latency, and energy consumption remain significant constraints. Consequently, researchers should explicitly report the numerical precision used in their studies, conduct comparative analyses between the implementations of FP32 and FP64 where applicable, and publicly release FP64-trained models to improve reproducibility and facilitate collaborative research. To support meaningful evaluations, standardized benchmarks that capture precision sensitivity across diverse scientific tasks are essential. Such benchmarks would provide a consistent framework for quantifying trade-offs between numerical precision, computational efficiency, and reproducibility in scientific ML research.

Exploring high-precision models and mixed high-precision training Inspired by mixed-precision training [65], we propose extending this concept to high-precision training by identifying precision-sensitive layers and selectively training them using FP64 arithmetic. This approach mirrors conventional mixed-precision strategies that utilize reduced precision (*e.g.*, FP32 and FP16) for most network layers while maintaining higher precision for numerically sensitive operations such as batch normalization and softmax. This direction holds significance from an energy efficiency perspective, as FP64 training inherently consumes more energy than FP32. While scientific ML offers computational advantages over traditional scientific computing methods, energy consumption remains a persistent concern. Investigating novel model architectures and training techniques that preserve high numerical precision while enhancing energy efficiency will be essential for broader adoption of scientific ML.

Collaboration with natural scientists Achieving meaningful progress in scientific ML requires interdisciplinary collaboration with with natural scientists. This is not merely a conceptual argument but a practical requirement, as ML researchers often lack the domain-specific intuition to determine the appropriate level of numerical precision for a given scientific task. For instance, research on ML potential is published in both traditional scientific journals and ML conferences, yet the evaluation criteria and priorities differ substantially between these communities [9, 48]. Strengthening the collaboration will help bridge this gap, ensuring that precision requirements align with both scientific validity and practical usability.

Integrating ML into traditional computational methods Rather than exclusively developing high-precision ML models, one of the alternative approaches is to integrate ML into traditional computational methods to achieve both accuracy and efficiency. One promising strategy is to employ ML models while acknowledging their inherent numerical limitations and using them to generate an approximate solution [3, 87, 69]. These ML-generated approximations subsequently serve as an initial guess for traditional computational methods, significantly accelerating convergence while preserving numerical precision. This hybrid approach presents a compelling solution for scientific applications where both computational speed and numerical accuracy are necessary.

5 Alternative Views

This section presents alternative views challenging our position and offers responses to these concerns.

Q1: Is the extra computation cost due to higher precision tolerable? The most straightforward negative impact of using higher precision is the increased computational burden. For example, on NVIDIA A100 and H100 GPUs, FP64 operations are approximately twice as slow as FP32 operations. While this overhead may be acceptable for training that takes only a few hours, it becomes prohibitive for large-scale models trained across multiple GPUs over longer periods spanning several weeks or months. In addition, certain classes of GPUs (*e.g.*, RTX A6000) feature intentionally constrained FP64 performance, with throughput ranging from 1/32 to 1/64 of their FP32 capabilities. This hardware limitation makes consistent development of scientific ML models using double precision computationally inefficient. Therefore, as discussed in Section 4, a systematic analysis of numerical

precision's impact on model accuracy becomes essential, enabling practitioners to selectively optimize the balance between accuracy and computational efficiency.

Q2: Is the issue really about numerical precision, or could it be a capacity limitation of the model? An alternative perspective suggests that observed inaccuracies originate from fundamental limitations in network architecture or training methodologies rather than numerical precision constraints. This viewpoint posits that neural networks may lack sufficient expressivity to solve a given task, regardless of precision considerations. To distinguish numerical issues from capacity concerns, we can employ numerical analysis tools including condition numbers and numerical sensitivity analysis (*e.g.*, interval arithmetic [33]), to determine whether errors arise from numerical instability. Since modern neural networks heavily rely on matrix operations, existing research on matrix sensitivity provides a robust analytical foundation. These insights can help clarify the relationship between numerical stability and model expressivity.

Q3: If certain scientific computing tasks are not sensitive to numerical precision, does it matter? While not all scientific tasks require high numerical precision, focus should be directed toward fields where high precision is essential, such as quantum chemistry, materials science, and nonlinear physics, where even slight inaccuracies can lead to significant deviations. Currently, there is still limited understanding of which tasks, models, and environments are most affected by numerical precision and what factors contribute to these sensitivities. A systematic analysis is necessary to identify precision-critical cases before making broad assumptions about acceptable precision levels. Until a clear understanding is established, a precision-aware approach should be considered, while relaxed conditions can be applied only to tasks that demonstrably insensitive to precision.

Certain scientific tasks may not require explicit consideration of numerical precision, particularly those where logical reasoning is more critical than numerical accuracy, such as tasks relying on LLMs. These include explaining or summarizing experimental results or literature [111], generating hypotheses for scientific research [49, 61], providing guidance for tasks where the methodology is not clearly defined (*e.g.*, retrosynthesis), and assisting scientific educations [16]. In such cases, the role of ML extends beyond numerical fidelity, emphasizing knowledge synthesis and interpretability.

Q4: Is it possible to design models that can avoid precision-related issues? Numerical instability in scientific computing frequently originates from precision-sensitive operations including numerical differentiation, integration, and eigendecomposition. Designing scientific ML models that avoid these operations and instead directly predict their outcomes can help mitigate such instability. ML potentials exemplify this approach by directly predicting energies from atomic structures, bypassing the numerically sensitive integration and eigendecomposition required in DFT. This perspective extends to examining whether individual neural network layers are numerically stable, similar to spectral normalization [66] which enforces Lipschitz continuity to stabilize training.

However, avoiding numerical instability through model design is not always practical, as scientists require understanding of underlying processes rather than just final predictions. This has led to ML models that mimic traditional scientific computations, such as NeuralODEs and DEQs [21, 6, 108], which explicitly model computational processes and align better with scientific domains emphasizing interpretability. While end-to-end approaches remain effective when predictive accuracy is the primary concern, many scientific domains continue to depend on numerical precision and computational understanding, making precision-related issues an important ongoing research area.

6 Conclusions

Scientific ML has become a major field in modern ML research, with the goal of developing models that contribute to scientific discovery. This position paper highlights the impact of precision issues, which can affect the practical usability of scientific ML models but have been largely overlooked. The precision issues in scientific ML are closely tied to ethical concerns regarding the reliability and explainability of scientific findings. In summary, our contribution lies in a practical step toward making scientific ML models more reliable, reducing the risk of misleading scientific insights due to numerical inaccuracies. If our simple yet easily actionable proposal becomes widely adopted in scientific ML research field, it can enhance the practicality and thereby accelerate scientific discovery.

References

[1] Maryam Abbasi, Beatriz P. Santos, Tiago C. Pereira, Raul Sofia, Nelson R. C. Monteiro, Carlos J. V. Simões, Rui M. M. Brito, Bernardete Ribeiro, José L. Oliveira, and Joel P. Arrais. Designing optimized drug candidates with generative adversarial network. *Journal of Cheminformatics*, 14(1):40, 2022. (Cited on page 2)

[2] M. P. Andersson and P. Uvdal. New scale factors for harmonic vibrational frequencies using the b3lyp density functional method with the triple- ζ basis set 6-311+g(d,p). *The Journal of Physical Chemistry A*, 109(12):2937–2941, 2005. (Cited on page 3, 19)

[3] Sohei Arisaka and Qianxiao Li. Principled acceleration of iterative numerical methods using machine learning. In *Proc. the International Conference on Machine Learning (ICML)*, 2023. (Cited on page 8)

[4] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N. Kinch, R. Dustin Schaeffer, Claudia Millán, Hahnbeom Park, Carson Adams, Caleb R. Glassman, Andy DeGiovanni, Jose H. Pereira, Andria V. Rodrigues, Alberdina A. van Dijk, Ana C. Ebrecht, Diederik J. Opperman, Theo Sagmeister, Christoph Buhlheller, Tea Pavkov-Keller, Manoj K. Rathinaswamy, Udit Dalwadi, Calvin K. Yip, John E. Burke, K. Christopher Garcia, Nick V. Grishin, Paul D. Adams, Randy J. Read, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021. (Cited on page 1)

[5] S. R. Bahn and K. W. Jacobsen. An object-oriented scripting interface to a legacy electronic structure code. *Comput. Sci. Eng.*, 4(3):56–66, 2002. (Cited on page 5, 22)

[6] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2019. (Cited on page 6, 9)

[7] Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2019. (Cited on page 2)

[8] Ilyes Batatia, Simon Batzner, Dávid Péter Kovács, Albert Musaelian, Gregor N. C. Simm, Ralf Drautz, Christoph Ortner, Boris Kozinsky, and Gábor Csányi. The design space of e(3)-equivariant atom-centred interatomic potentials. *Nature Machine Intelligence*, 7(1):56–67, 2025. (Cited on page 6)

[9] Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gabor Csanyi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (Cited on page 5, 8)

[10] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1):2453, 2022. (Cited on page 5, 6)

[11] Peter Bauer, Alan Thorpe, and Gilbert Brunet. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, 2015. (Cited on page 6)

[12] Axel D Becke. Density-functional exchange-energy approximation with correct asymptotic behavior. *Physical review A*, 38(6):3098, 1988. (Cited on page 3)

[13] Axel D. Becke. Density-functional thermochemistry. iii. the role of exact exchange. *The Journal of Chemical Physics*, 98(7):5648–5652, 04 1993. (Cited on page 3, 19)

[14] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, 2007. (Cited on page 5)

[15] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models. In *Proc. the AAAI Conference on Artificial Intelligence (AAAI)*, 2024. (Cited on page 7)

[16] Arne Bewersdorff, Christian Hartmann, Marie Hornberger, Kathrin Seßler, Maria Bannert, Enkelejda Kasneci, Gjergji Kasneci, Xiaoming Zhai, and Claudia Nerdel. Taking the next step with generative artificial intelligence: The transformative role of multimodal large language models in science education. *Learning and Individual Differences*, 118:102601, 2025. (Cited on page 9)

[17] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654, 1973. (Cited on page 6)

[18] Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 2014. (Cited on page 1)

[19] Humphrey John Moule Bowen and Leslie Ernest Sutton. *Tables of interatomic distances and configuration in molecules and ions*. Number 11 in Special publication. Chemical Society, 1958. (Cited on page 3)

[20] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016. (Cited on page 6)

[21] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2018. (Cited on page 6, 9)

[22] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2818–2829, 2023. (Cited on page 2)

[23] DeepSeek-AI. Deepseek-v3 technical report, 2024. (Cited on page 7)

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2009. (Cited on page 1)

[25] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3.int8(): 8-bit matrix multiplication for transformers at scale. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, pages 30318–30332, 2022. (Cited on page 2, 7)

[26] Xiang Fu, Brandon M Wood, Luis Barroso-Luque, Daniel S Levine, Meng Gao, Misko Dzamba, and C Lawrence Zitnick. Learning smooth and expressive interatomic potentials for physical property prediction. *arXiv preprint arXiv:2502.12147*, 2025. (Cited on page 5)

[27] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se(3)-transformers: 3d roto-translation equivariant attention networks. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2020. (Cited on page 5)

[28] Johannes Gasteiger, Chandan Yeshwanth, and Stephan Günnemann. Directional message passing on molecular graphs via synthetic coordinates. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, pages 15421–15433, 2021. (Cited on page 5)

[29] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L Chiarotti, Matteo Cococcioni, Ismaila Dabo, Andrea Dal Corso, Stefano de Gironcoli, Stefano Fabris, Guido Fratesi, Ralph Gebauer, Uwe Gerstmann, Christos Gougoussis, Anton Kokalj, Michele Lazzeri, Layla Martin-Samos, Nicola Marzari, Francesco Mauri, Riccardo Mazzarello, Stefano Paolini, Alfredo Pasquarello, Lorenzo Paulatto, Carlo

Sbraccia, Sandro Scandolo, Gabriele Sclauzero, Ari P Seitsonen, Alexander Smogunov, Paolo Umari, and Renata M Wentzcovitch. Quantum espresso: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39):395502, 2009. (Cited on page 3)

[30] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proc. the International Conference on Machine Learning (ICML)*, 2017. (Cited on page 5)

[31] Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018. (Cited on page 2)

[32] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2014. (Cited on page 1)

[33] T. Hickey, Q. Ju, and M. H. Van Emden. Interval arithmetic: From principles to implementation. *J. ACM*, 48(5):1038–1068, 2001. (Cited on page 9)

[34] Jacob Hilton, Jie Tang, and John Schulman. Scaling laws for single-agent reinforcement learning. *arXiv preprint arXiv:2301.13442*, 2023. (Cited on page 2)

[35] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2020. (Cited on page 1, 6)

[36] Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3D. In *Proc. the International Conference on Machine Learning (ICML)*, pages 8867–8887, 2022. (Cited on page 2)

[37] Ryan Jacobs, Maciej P Polak, Lane E Schultz, Hamed Mahdavi, Vasant Honavar, and Dane Morgan. Regression with large language models for materials and molecular property prediction. *arXiv preprint arXiv:2409.06080*, 2024. (Cited on page 7)

[38] Robert O Jones and Olle Gunnarsson. The density functional formalism, its applications and prospects. *Reviews of Modern Physics*, 61(3):689, 1989. (Cited on page 3)

[39] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021. (Cited on page 1)

[40] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. (Cited on page 2)

[41] Georgia Karagiorgi, Gregor Kasieczka, Scott Kravitz, Benjamin Nachman, and David Shih. Machine learning in the search for new fundamental physics. *Nature Reviews Physics*, 4(6):399–412, 2022. (Cited on page 1)

[42] Sungwon Kim, Juhwan Noh, Geun Ho Gu, Alan Aspuru-Guzik, and Yousung Jung. Generative adversarial networks for crystal structure prediction. *ACS Central Science*, 6(8):1412–1420, 2020. (Cited on page 2)

[43] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proc. the International Conference on Learning Representations (ICLR)*, 2014. (Cited on page 1)

[44] Emir Kocer, Tsz Wai Ko, and Jörg Behler. Neural network potentials: A concise overview of methods. *Annual Review of Physical Chemistry*, 73(Volume 73, 2022):163–186, 2022. (Cited on page 2, 5)

[45] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical review*, 140(4A):A1133, 1965. (Cited on page 3)

[46] Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (Cited on page 7)

[47] Dávid Péter Kovács, J. Harry Moore, Nicholas J. Browning, Ilyes Batatia, Joshua T. Horton, Yixuan Pu, Venkat Kapil, William C. Witt, Ioan-Bogdan Magdău, Daniel J. Cole, and Gábor Csányi. Mace-off: Short-range transferable machine learning force fields for organic molecules. *Journal of the American Chemical Society*, 2025. (Cited on page 22)

[48] Dávid Péter Kovács, Ilyes Batatia, Eszter Sára Arany, and Gábor Csányi. Evaluation of the mace force field architecture: From medicinal chemistry to materials science. *The Journal of Chemical Physics*, 159(4):044118, 2023. (Cited on page 8)

[49] Shrinidhi Kumbhar, Venkatesh Mishra, Kevin Coutinho, Divij Handa, Ashif Iquebal, and Chitta Baral. Hypothesis generation for materials discovery and design using goal-driven and constraint-guided llm agents. *arXiv preprint arXiv:2501.13299*, 2025. (Cited on page 9)

[50] Silvan Käser, Luis Itza Vazquez-Salazar, Markus Meuwly, and Kai Töpfer. Neural network potentials for chemistry: concepts, applications and prospects. *Digital Discovery*, 2:28–58, 2023. (Cited on page 2, 5)

[51] Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S. Yu. Large language models in law: A survey. *AI Open*, 5:181–196, 2024. (Cited on page 1)

[52] Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, Ivano E Castelli, Rune Christensen, Marcin Dułak, Jesper Friis, Michael N Groves, Bjørk Hammer, Cory Hargus, Eric D Hermes, Paul C Jennings, Peter Bjerre Jensen, James Kermode, John R Kitchin, Esben Leonhard Kolsbjerg, Joseph Kubal, Kristen Kaasbjerg, Steen Lysgaard, Jón Bergmann Maronsson, Tristan Maxson, Thomas Olsen, Lars Pastewka, Andrew Peterson, Carsten Rostgaard, Jakob Schiøtz, Ole Schütt, Mikkel Strange, Kristian S Thygesen, Tejs Vegge, Lasse Vilhelmsen, Michael Walter, Zhenhua Zeng, and Karsten W Jacobsen. The atomic simulation environment—a python library for working with atoms. *Journal of Physics: Condensed Matter*, 29(27):273002, 2017. (Cited on page 5, 22)

[53] Sihang Li, Zhiyuan Liu, Yanchen Luo, Xiang Wang, Xiangnan He, Kenji Kawaguchi, Tat-Seng Chua, and Qi Tian. Towards 3d molecule-text interpretation in language models. In *Proc. the International Conference on Learning Representations (ICLR)*, 2024. (Cited on page 7)

[54] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *Proc. the International Conference on Learning Representations (ICLR)*, 2023. (Cited on page 5)

[55] Jaechang Lim, Seongok Ryu, Jin Woo Kim, and Woo Youn Kim. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *Journal of Cheminformatics*, 10(1):31, 2018. (Cited on page 2)

[56] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017. (Cited on page 1)

[57] Fangxin Liu, Wenbo Zhao, Zhezhi He, Yanzhi Wang, Zongwu Wang Wang, Changzhi Dai, Xiaoyao Liang, and Li Jiang. Improving neural network efficiency via post-training quantization with adaptive floating-point. In *Proc. of the IEEE international conference on computer vision (ICCV)*, 2021. (Cited on page 7)

[58] Shih-yang Liu, Zechun Liu, Xijie Huang, Pingcheng Dong, and Kwang-Ting Cheng. LLM-FP4: 4-bit floating-point quantized transformers. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 592–605, 2023. (Cited on page 2, 7)

[59] Siyu Liu, Tongqi Wen, Beilin Ye, Zhuoyuan Li, and David J Srolovitz. Large language models for material property predictions: elastic constant tensor prediction and materials design. *arXiv preprint arXiv:2411.12280*, 2024. (Cited on page 7)

[60] AI @ Meta Llama Team. The llama 3 herd of models, 2024. (Cited on page 7)

[61] Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024. (Cited on page 9)

[62] Tristan Maxson, Ademola Soyemi, Benjamin W. J. Chen, and Tibor Szilvási. Enhancing the quality and reliability of machine learning interatomic potentials through better reporting practices. *The Journal of Physical Chemistry C*, 128(16):6524–6537, 2024. (Cited on page 6)

[63] James Clerk Maxwell. Viii. a dynamical theory of the electromagnetic field. *Philosophical Transactions of the Royal Society of London*, 155:459–512, 1865. (Cited on page 4)

[64] James Clerk Maxwell. On physical lines of force. *Philosophical magazine*, 90(S1):11–23, 2010. (Cited on page 4)

[65] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training. In *Proc. the International Conference on Learning Representations (ICLR)*, 2018. (Cited on page 2, 8)

[66] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proc. the International Conference on Learning Representations (ICLR)*, 2018. (Cited on page 9)

[67] Dane Morgan and Ryan Jacobs. Opportunities and challenges for machine learning in materials science. *Annual Review of Materials Research*, 50(1):71–103, 2020. (Cited on page 1)

[68] Yo Nakamura, Suguru Shiratori, Ryota Takagi, Michihiro Sutoh, Iori Sugihara, Hideaki Nagano, and Kenjiro Shimano. Physics-informed neural network applied to surface-tension-driven liquid film flows. *International Journal for Numerical Methods in Fluids*, 94(9):1359–1378, 2022. (Cited on page 6)

[69] Kevin J Napier. Improved initial guesses for numerical solutions of kepler's equation. *arXiv preprint arXiv:2411.15374*, 2024. (Cited on page 8)

[70] Oren Neumann and Claudius Gros. Scaling laws for a multi-agent reinforcement learning model. In *Proc. the International Conference on Learning Representations (ICLR)*, 2023. (Cited on page 2)

[71] Eric Nguyen, Michael Poli, Matthew G. Durrant, Brian Kang, Dhruva Katrekar, David B. Li, Liam J. Bartie, Armin W. Thomas, Samuel H. King, Garyk Brixi, Jeremy Sullivan, Madelena Y. Ng, Ashley Lewis, Aaron Lou, Stefano Ermon, Stephen A. Baccus, Tina Hernandez-Boussard, Christopher Ré, Patrick D. Hsu, and Brian L. Hie. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, 2024. (Cited on page 2)

[72] Ardavan F. Oskooi, David Roundy, Mihai Ibanescu, Peter Bermel, J.D. Joannopoulos, and Steven G. Johnson. Meep: A flexible free-software package for electromagnetic simulations by the fdtd method. *Computer Physics Communications*, 181(3):687–702, 2010. (Cited on page 4, 19)

[73] Ahmet Murat Ozbayoglu, Mehmet Ugur Gudelek, and Omer Berat Sezer. Deep learning for financial applications : A survey. *Applied Soft Computing*, 93:106384, 2020. (Cited on page 1)

[74] Jinyoung Park, Minseong Bae, Dohwan Ko, and Hyunwoo J. Kim. LLamo: Large language model-based molecular graph assistant. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2024. (Cited on page 7)

[75] Xingang Peng, Jiaqi Guan, Qiang Liu, and Jianzhu Ma. MolDiff: Addressing the atom-bond inconsistency problem in 3D molecule diffusion generation. In *Proc. the International Conference on Machine Learning (ICML)*, pages 27611–27629, 2023. (Cited on page 2)

[76] Max Planck. Zur theorie des gesetzes der energieverteilung im normalspektrum. *Berlin*, pages 237–245, 1900. (Cited on page 3)

[77] Steve Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1–19, 1995. (Cited on page 5)

[78] Ryan Poplin, Pi-Chuan Chang, David Alexander, Scott Schwartz, Thomas Colthurst, Alexander Ku, Dan Newburger, Jojo Dijamco, Nam Nguyen, Pegah T. Afshar, Sam S. Gross, Lizzie Dorfman, Cory Y. McLean, and Mark A. DePristo. A universal snp and small-indel variant caller using deep neural networks. *Nature Biotechnology*, 36(10):983–987, 2018. (Cited on page 2)

[79] Oleksii Prykhodko, Simon Viet Johansson, Panagiotis-Christos Kotsias, Josep Arús-Pous, Esben Jannik Bjerrum, Ola Engkvist, and Hongming Chen. A de novo molecular generation method using latent vector based generative adversarial network. *Journal of Cheminformatics*, 11(1):74, 2019. (Cited on page 2)

[80] A. Pukrittayakamee, M. Malshe, M. Hagan, L. M. Raff, R. Narulkar, S. Bukkapatnum, and R. Komanduri. Simultaneous fitting of a potential-energy surface and its corresponding force fields using feedforward neural networks. *The Journal of Chemical Physics*, 130(13):134101, 2009. (Cited on page 5)

[81] Maithra Raghu and Eric Schmidt. A survey of deep learning for scientific discovery. *arXiv preprint arXiv:2003.11755*, 2020. (Cited on page 1)

[82] 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. *Journal of Computational Physics*, 378:686–707, 2019. (Cited on page 2, 6)

[83] Xiaoli Ren, Xiaoyong Li, Kaijun Ren, Junqiang Song, Zichen Xu, Kefeng Deng, and Xiang Wang. Deep learning-based weather prediction: A survey. *Big Data Research*, 23:100178, 2021. (Cited on page 1)

[84] Andre Niyongabo Rubungo, Kangming Li, Jason Hattrick-Simpers, and Adji Bousso Dieng. Llm4mat-bench: Benchmarking large language models for materials property prediction. *arXiv preprint arXiv:2411.00177*, 2024. (Cited on page 7)

[85] Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018. (Cited on page 2)

[86] Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *Proc. the International Conference on Machine Learning (ICML)*, 2021. (Cited on page 5)

[87] Luca Saverio. Accelerating convergence of linear iterative solvers using machine learning. Master's thesis, Politecnico di Milano, 2023. (Cited on page 8)

[88] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009. (Cited on page 5)

[89] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (Cited on page 1)

[90] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2017. (Cited on page 5)

[91] Ramansh Sharma and Varun Shankar. Accelerated training of physics-informed neural networks (pinns) using meshless discretizations. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (Cited on page 7)

[92] Jingzhe Shi, Qinwei Ma, Huan Ma, and Lei Li. Scaling law for time series forecasting. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2024. (Cited on page 2)

[93] J. S. Smith, O. Isayev, and A. E. Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.*, 8:3192–3203, 2017. (Cited on page 5)

[94] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. (Cited on page 2)

[95] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, 2019. (Cited on page 1, 6)

[96] Walter A Strauss. *Partial differential equations: An introduction*. John Wiley & Sons, 2007. (Cited on page 6)

[97] Hyeong Chan Suh, Jaekak Yoo, Kangmo Yeo, Dong Hyeon Kim, Yo Seob Won, Taehoon Kim, Youngwoo Cho, Ki Kang Kim, Seung Mi Lee, Heejun Yang, Dong-Wook Kim, and Mun Seok Jeong. Probing nanoscale structural perturbation in a ws2 monolayer via explainable artificial intelligence. *Applied Physics Reviews*, 12(2):021406, 04 2025. (Cited on page 1)

[98] Qiming Sun, Timothy C. Berkelbach, Nick S. Blunt, George H. Booth, Sheng Guo, Zhendong Li, Junzi Liu, James D. McClain, Elvira R. Sayfutyarova, Sandeep Sharma, Sebastian Wouters, and Garnet Kin-Lic Chan. Pyscf: the python-based simulations of chemistry framework. *WIREs Computational Molecular Science*, 8(1):e1340, 2018. (Cited on page 3, 19)

[99] The Royal Swedish Academy of Sciences. The nobel prize in chemistry 2024, 2024. (Cited on page 1)

[100] Philipp Thölke and Gianni De Fabritiis. Equivariant transformers for neural network based molecular potentials. In *Proc. the International Conference on Learning Representations (ICLR)*, 2022. (Cited on page 5)

[101] Aidan P. Thompson, H. Metin Aktulga, Richard Berger, Dan S. Bolintineanu, W. Michael Brown, Paul S. Crozier, Pieter J. in 't Veld, Axel Kohlmeyer, Stan G. Moore, Trung Dac Nguyen, Ray Shan, Mark J. Stevens, Julien Tranchida, Christian Trott, and Steven J. Plimpton. Lammps - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, 271:108171, 2022. (Cited on page 5)

[102] Julian Tirado-Rives and William L. Jorgensen. Performance of b3lyp density functional methods for a large set of organic molecules. *Journal of Chemical Theory and Computation*, 4(2):297–306, 2008. (Cited on page 3)

[103] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. (Cited on page 7)

[104] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. (Cited on page 7)

[105] David J Tritton. *Physical fluid dynamics*. Springer Science & Business Media, 2012. (Cited on page 6)

[106] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2017. (Cited on page 5)

[107] Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. Ace 2005 multilingual training corpus, 2006. (Cited on page 1)

[108] Zun Wang, Chang Liu, Nianlong Zou, He Zhang, Xinran Wei, Lin Huang, Lijun Wu, and Bin Shao. Infusing self-consistency into density functional theory hamiltonian prediction via deep equilibrium models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, pages 89652–89681, 2024. (Cited on page 9)

[109] Sarah Webb et al. Deep learning for biology. *Nature*, 554(7693):555–557, 2018. (Cited on page 1)

[110] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2022. (Cited on page 7)

[111] Tong Xie, Yuwei Wan, Yixuan Liu, Yuchen Zeng, Wenjie Zhang, Chunyu Kit, Dongzhan Zhou, and Bram Hoex. Darwin 1.5: Large language models as materials science adapted learners. *arXiv preprint arXiv:2412.11970*, 2024. (Cited on page 9)

[112] Yuzhuang Xu, Xu Han, Zonghan Yang, Shuo Wang, Qingfu Zhu, Zhiyuan Liu, Weidong Liu, and Wanxiang Che. Onebit: Towards extremely low-bit large language models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2024. (Cited on page 2, 7)

[113] Takeshi Yanai, David P Tew, and Nicholas C Handy. A new hybrid exchange–correlation functional using the coulomb-attenuating method (cam-b3lyp). *Chemical Physics Letters*, 393(1):51–57, 2004. (Cited on page 3, 19)

[114] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Proc. the Advances in Neural Information Processing Systems (NeurIPS)*, 2023. (Cited on page 7)

[115] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on antennas and propagation*, 14(3):302–307, 1966. (Cited on page 3)

[116] Jaekak Yoo, Youngwoo Cho, Byeonggeun Jeong, Soo Ho Choi, Ki Kang Kim, Seong Chu Lim, Seung Mi Lee, Jaegul Choo, and Mun Seok Jeong. Explainable artificial intelligence approach to identify the origin of phonon-assisted emission in wse2 monolayer. *Advanced Intelligent Systems*, 5(7):2200463, 2023. (Cited on page 1)

[117] Jaekak Yoo, Youngwoo Cho, Dong Hyeon Kim, Jaeseok Kim, Tae Geol Lee, Seung Mi Lee, Jaegul Choo, and Mun Seok Jeong. Unraveling the role of raman modes in evaluating the degree of reduction in graphene oxide via explainable artificial intelligence. *Nano Today*, 57:102366, 2024. (Cited on page 1)

[118] Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, Roberto Sordillo, Lixin Sun, Jake Smith, Bichlien Nguyen, Hannes Schulz, Sarah Lewis, Chin-Wei Huang, Ziheng Lu, Yichi Zhou, Han Yang, Hongxia Hao, Jielan Li, Chunlei Yang, Wenjie Li, Ryota Tomioka, and Tian Xie. A generative model for inorganic materials design. *Nature*, Jan 2025. (Cited on page 2)

[119] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 12104–12113, 2022. (Cited on page 2)

[120] Qiang Zhang, Keyan Ding, Tianwen Lv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, et al. Scientific large language models: A survey on biological & chemical domains. *ACM Computing Surveys*, 2024. (Cited on page 2)

[121] Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, and Guolin Ke. Uni-mol: A universal 3d molecular representation learning framework. In *Proc. the International Conference on Learning Representations (ICLR)*, 2023. (Cited on page 7)

[122] Jian Zhou and Olga G. Troyanskaya. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, 12(10):931–934, 2015. (Cited on page 2)

[123] Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. A survey on model compression for large language models. *Transactions of the Association for Computational Linguistics*, 12:1556–1577, 2024. (Cited on page 2)

[124] Jan Zrimec, Xiaozhi Fu, Azam Sheikh Muhammad, Christos Skrekas, Vykintas Jauniskis, Nora K. Speicher, Christoph S. Börlin, Vilhelm Verendel, Morteza Haghir Chehreghani, Devdatt Dubhashi, Verena Siewers, Florian David, Jens Nielsen, and Aleksej Zelezniak. Controlling gene expression with deep generative design of regulatory dna. *Nature Communications*, 13(1):5099, 2022. (Cited on page 2)

A Details of computational methods

This section provides detailed information on the DFT and FDTD calculations.

A.1 Density functional theory calculation

Computational environment Quantum mechanical calculations were performed using PySCF version 2.7.0 [98]. To evaluate the impact of numerical precision, we conducted the same calculations using both single precision and double precision by declaring `np.float32` and `np.float64`, respectively. All simulations were conducted using two nodes of an AMD EPYC 7543 32-Core Processor.

Simulation setup The input water molecule consists of a single oxygen atom at (0.000000, 0.000000, 0.000000) and two hydrogen atoms at (0.757000, 0.586000, 0.000000) and (-0.757000, 0.586000, 0.000000), respectively. To compare the geometry optimization result of a water molecule based on different exchange functionals, we performed both Hartree-Fock calculations and density functional theory calculations using the B3LYP functional [13, 113]. For both methods, we employed the 6-311++G(d,p) basis set [2]. The default convergence tolerances for structural stabilization were set as follows: $|\Delta E| < 1.00 \times 10^{-6}$, RMS-Grad $< 3.00 \times 10^{-4}$, Max-Grad $< 4.50 \times 10^{-4}$, RMS-Disp $< 1.20 \times 10^{-3}$, and Max-Disp $< 1.80 \times 10^{-3}$.

A.2 Finite-difference time-domain calculation

Nonlinear material properties Kerr media were modeled with a second-order nonlinear susceptibility (χ^2) ranging from 10^{-12} to 10^{-2} and the refractive index was set to 1.65 to mimic the conventional nonlinear materials like beta barium borate. The nonlinear polarization of the material was expressed as:

$$P = \epsilon_0(\chi^{(1)}E + \chi^{(2)}E^2 + \chi^{(3)}E^3 + ...) \tag{1}$$

And the second-order nonlinear polarization term is represented as: $P^{(2)} = \epsilon_0\chi^{(2)}E^2$. Meep incorporates such nonlinear polarization terms into Maxwell's equations to simulate interactions between electromagnetic waves and the material in the time domain

$$\bigtriangledown \times H = \epsilon_0 \frac{\partial E}{\partial t} + \frac{\partial P}{\partial t} \tag{2}$$

$$\bigtriangledown \times E = -\mu_0 \frac{\partial H}{\partial t} \tag{3}$$

Simulation setup The simulation domain consisted of a 100 μm medium, a 1 μm thick boundary layer, and 2 μm buffer regions at both ends. The spatial resolution was user-defined to capture fine electromagnetic field characteristics. Kerr media were placed at the center of the domain, with χ^2 explicitly defined. The calculations were performed using Meep v1.29.0 [72], an open-source FDTD software, with both FP32 and FP64 precisions on a single core of an AMD Ryzen 5 8500G processor.

Source and monitor definition The source was defined as a Gaussian plane wave with a central wavelength of 1.55 μm and a bandwidth of 0.15. Both the source and monitors were positioned 1 μm outside the nonlinear medium, with the electric field oscillating along the x-axis. Simulations were executed to allow sufficient decay of the fields after the source was turned off to confirm accurate measurements.

Harmonic generation and analysis Using the Meep's add flux function, the optical flux outside the nonlinear medium was measured, and the transmitted power spectra of the fundamental frequency (ω) and harmonic components (2ω, 3ω) were calculated. The add flux function records the time-domain values of electric and magnetic fields at specific locations, then performs a Fourier transform to convert them into the frequency domain to compute flux. This process allows precise analysis of the intensity of each frequency component within the user-defined frequency range and intervals. The analysis frequency range extended from ω/2 to 3.5ω, encompassing all relevant frequency bands of interest. Flux measurements were particularly useful for understanding the interaction between

newly generated harmonic components and existing frequency components caused by the material's nonlinearity.

Results and reproducibility Simulation results demonstrated how the intensity and distribution of harmonic components varied with changes in χ^2. The nonlinear modeling capabilities of Meep enabled precise analysis of harmonic generation characteristics in nonlinear optical materials.

B Additional experimental results

In this section, we provide additional experimental results that supplement the results presented in the main text.

B.1 Atomic coordination difference between FP32 and FP64

A detailed examination of the atomic coordinates in Table 1 further highlights the differences. While the coordinates obtained from FP64 differ only by approximately 0.01 Å for oxygen and hydrogen atoms, FP32 results display considerable deviation. Notably, the FP32-calculated atomic positions deviate by up to 0.4 Å from those obtained using FP64, a significant difference considering that the O-H bond length itself is only 0.957 Å. In addition, the total energy difference between FP32 and FP64 calculations is approximately 1.1 Hartree (equivalent to 29.93 eV), which exceeds the formation energy of water (2.9 eV) by more than an order of magnitude. This clearly indicates that the FP32 result corresponds to a structure that is impossible to exist in reality. These results demonstrate that FP32 lacks the numerical precision necessary to achieve sufficient convergence tolerance in scientific computations. The failure of a simple molecular system such as water to reach an optimized structure under FP32 precision indicates its fundamental limitations in scientific calculations.

Table 1: Comparison of atomic coordinates and total energy for geometry-optimized water molecule at FP32 and FP64 precision levels. Both calculations used the 6-311++G(d,p) basis set. As indicated by an asterisk (*), FP32 calculations failed to converge for both HF and DFT methods, while FP64 results show compatibility between HF and DFT.

		HF 6-311++G(d,p)		**DFT** (B3LYP) 6-311++G(d,p)	
		FP32	FP64	FP32	FP64
Atomic coordinates (Å)	O_x	-0.000356*	0.000000	0.009524*	0.000000
	O_y	0.246311*	0.014028	0.578655*	0.000780
	O_z	0.000000*	0.000000	0.000000*	0.000000
	H_1x	0.453099*	0.752792	0.026584*	0.763642
	H_1y	0.534244*	0.578999	0.998814*	0.585902
	H_1z	0.000000*	0.000000	0.000000*	0.000000
	H_2x	-0.453725*	-0.752792	-0.024889*	-0.763642
	H_2y	0.534404*	0.578999	0.998054*	0.585902
	H_2z	0.000000*	0.000000	0.000000*	0.000000
Total energy (Ha)		-74.938*	-76.053	*N/A**	-76.458

**Not Converged*

B.2 Absorbed power density of a SiO_2 cylinder

To validate our FDTD workflow against a well-characterised linear system, we also simulated the absorption of a single silica (SiO_2) cylinder under normal-incidence plane-wave illumination.

Geometry and material A two-dimensional square domain of side length 20 μm was created, containing one infinitely long cylinder of radius 1.0 μm centred at the origin. SiO_2 was taken from the built-in SiO_2 material in Meep, so its frequency-dependent permittivity—and hence refractive index—were implicitly evaluated at the simulation's centre frequency. Vacuum surrounded the cylinder. Mirror symmetry along the y-axis halved the computational cost.

Source definition A continuous-wave Gaussian plane wave, polarized along $\hat{z}$ (out-of-plane E_z), impinged from the left boundary. The central wavelength was 1.0 μm with a 10 $\%$ fractional bandwidth, wide enough to sample the vicinity of the design wavelength while narrow enough to approximate monochromatic excitation.

Monitors and post-processing

- **Incident-flux monitor.** A line segment at $x = -2.0\mu m$ recorded the power of the incoming wave, serving as a reference for normalising absorption.
- **Absorbed-flux box.** A closed rectangular contour wrapped tightly around the cylinder. By integrating the Poynting vector over this surface, net absorbed power P_{abs} was obtained directly.
- **DFT field monitor.** A square region coincident with the flux box captured E_z and D_z fields in the frequency domain via Meep's discrete Fourier transform facility. Absorbed power density was then evaluated volumetrically as

$$p_{\text{abs}}(f) = 2\pi f \ \text{Im}\big(\overline{E_z}\, D_z\big) = 2\pi f \ \text{Im}(\varepsilon)\, |E_z|^2.$$

  where f is frequency, ε is the complex permittivity of SiO_2, and the overbar denotes complex conjugation. Integrating p_{abs} over the cylinder volume reproduced P_{abs} obtained from the flux box, providing a cross-check on numerical consistency.

Simulation parameters The same spatial resolution used in the nonlinear study (*i.e.* user-defined) was retained to capture sub-wavelength field variations near the curved surface. Perfectly matched layers of 2 μm thickness enclosed the domain to eliminate spurious reflections.

This linear-dielectric benchmark served two purposes: (*i*) it confirmed the accuracy of our absorption-post-processing pipeline before applying it to nonlinear scenarios, and (*ii*) it provided a reference scale against which to compare the additional harmonic-generation pathways introduced by the Kerr media.

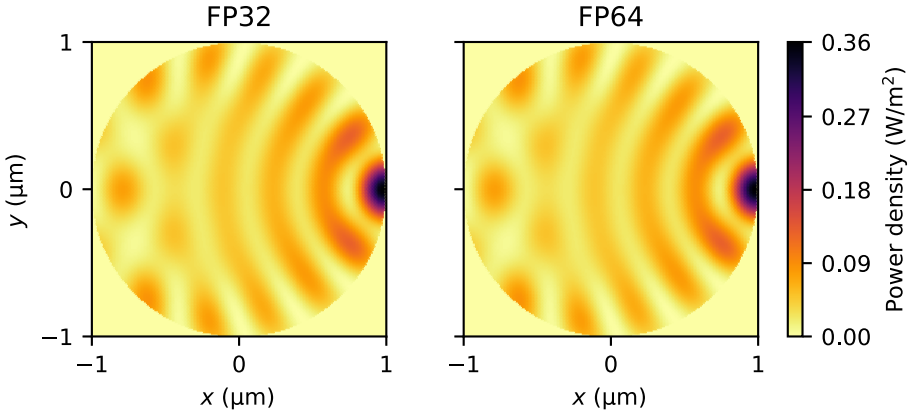

Figure 3: Absorbed power density of the SiO_2 cylinder computed with FP32 and FP64.

Results and discussion In double precision (FP64) the spatially averaged absorbed–power density over the silica cylinder is $\langle p_{\text{abs}} \rangle = 3.2475 \times 10^{-2}$ (W/m^2), with a standard deviation $\sigma = 3.4241 \times 10^{-2}$. Using single precision (FP32) we obtained $\langle p_{\text{abs}} \rangle = 3.2476 \times 10^{-2}$ and $\sigma = 3.4243 \times 10^{-2}$. The absolute differences between the two datasets are, respectively, $\Delta\langle p_{\text{abs}} \rangle = 1.96 \times 10^{-5}$ and $\Delta\sigma = 3.02 \times 10^{-5}$, which correspond to relative errors of 6.0×10^{-2} $\%$ and 8.8×10^{-2} $\%$. These discrepancies are two orders of magnitude smaller than the intrinsic statistical spread of the fields and therefore negligible for any practical analysis.

This near-identity arises because (*i*) SiO_2 is essentially loss-free at the operating wavelength, so the dynamic range of E_z and D_z is modest; and (*ii*) the perfectly matched layers efficiently remove outgoing waves, preventing late-time reflections that could amplify numerical noise.

We therefore conclude that, for linear absorption in a dielectric cylinder, Meep's single-precision kernel yields numerically indistinguishable results from double precision while offering lower memory usage and faster runtimes. This benchmark justifies the use of FP32 for the more demanding nonlinear Kerr-media simulations reported in the main text.

B.3 Toy experiments using MACE potential

Experimental setup Both toy experiments described in Section 3.1 were conducted under the same experimental setup. Among the various MACE model families, we used the `MACE-OFF23 medium` checkpoint [47]; the MACE codebase version was 0.3.10, and the ASE library [5, 52] version was 3.24.0. Each experiment was completed within 10 minutes on a single NVIDIA H100 GPU.

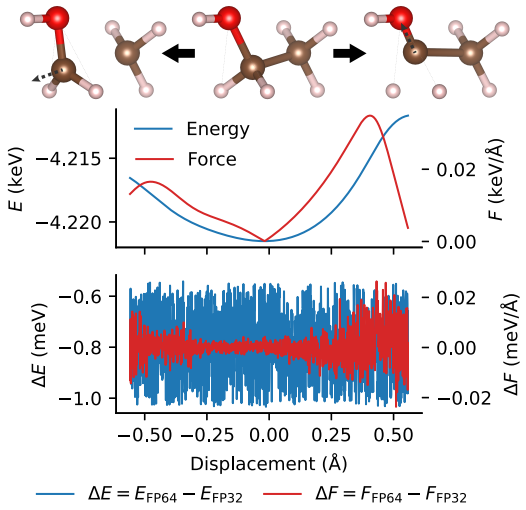

Figure 4: MACE model calculations showing energy (red) and force (blue) changes during carbon atom displacement in ethanol. FP32 versus FP64 precision reveals minimal deviations (1 meV energy, 0.02 meV/Å force).

Ethanol experiment We selected one of the carbon atoms in the ethanol molecule, specifically the one closest to the oxygen atom, and displaced it along a predefined direction while computing the corresponding energies and forces. The top panel of Figure 4 visualizes this setup. The original ethanol structure is shown in the center, with the displaced structures placed on either side. The direction of the carbon atom's movement is indicated by a dotted arrow.

The PES resulting from this displacement is shown in the middle panel. The energy (left y-axis) reaches a minimum when the displacement is zero, increases as the carbon atom moves away from its original position due to growing structural deformation, and then decreases again. The force (right y-axis) exhibits a similar trend, increasing from near zero and then decreasing as the displacement reverses.

The bottom panel shows the differences between FP32 and FP64 predictions for energy and force along the displacement path. The energy difference remains relatively uniform across the range. In contrast, the force difference is minimal near the equilibrium structure and increases gradually as the displacement moves the system farther from the optimal configuration.

Oseltamivir experiment We downloaded the 3D structure of oseltamivir from PubChem[3] and used it in our experiment. To compute the vibrational modes of a molecule, structural optimization is required prior to vibrational analysis. For this purpose, we performed geometry optimization using the MACE-OFF model and the ASE library, employing the BFGS optimizer with the maximum force threshold set to 0.01 eV/Å.

[3] https://pubchem.ncbi.nlm.nih.gov/compound/Oseltamivir

Table 2: Calculated vibrational modes of the oseltamivir molecule, the active ingredient in Tamiflu, using the MACE framework. Out of 30 total modes, four imaginary phonon modes were excluded, and only the 26 real phonon modes are presented.

Mode	Frequency (cm^{-1})		
	FP32	FP64	Δ
1	0.418	0.060	0.357
2	2.003	1.403	0.600
3	20.729	21.107	0.377
4	23.581	23.481	0.099
5	30.129	30.058	0.071
6	32.152	31.345	0.807
7	42.711	42.550	0.161
8	59.571	58.907	0.663
9	60.734	60.132	0.602
10	**77.851**	**76.113**	**1.738**
11	**80.128**	**78.696**	**1.431**
12	92.388	92.069	0.318
13	94.896	94.720	0.176
14	103.763	103.375	0.387
15	113.600	113.514	0.085
16	144.647	144.908	0.260
17	153.184	153.488	0.304
18	180.164	179.934	0.230
19	183.479	183.414	0.065
20	212.501	212.516	0.014
21	228.214	228.076	0.138
22	240.207	240.086	0.121
23	250.314	250.124	0.190
24	255.417	256.290	0.873
25	263.003	262.807	0.195
26	270.000	269.875	0.124

Following the optimization, we computed the vibrational modes using the ASE library as well. A total of 30 modes were obtained, among which four modes were identified as imaginary phonon modes. The remaining 26 real phonon modes are reported in Table 2.