Allegro-Legato: Scalable, Fast, and Robust Neural-Network Quantum Molecular Dynamics via Sharpness-Aware Minimization

Hikaru Ibayashi[1](✉), Taufeq Mohammed Razakh[1], Liqiu Yang[1], Thomas Linker[1], Marco Olguin[2], Shinnosuke Hattori[3], Ye Luo[4], Rajiv K. Kalia[1], Aiichiro Nakano[1], Ken-ichi Nomura[1], and Priya Vashishta[1]

[1] Collaboratory for Advanced Computing and Simulations, University of Southern California, Los Angeles, CA 90089, USA
ibayashi@usc.edu

[2] Center for Advanced Research Computing, University of Southern California, Los Angeles, CA 90089, USA

[3] Advanced Research Laboratory, R&D Center, Sony Group Corporation, Atsugi Tec. 4-14-1 Asahi-cho, Atsugi-shi, Kanagawa 243-0014, Japan

[4] Argonne Leadership Computing Facility, Argonne National Laboratory, Lemont, IL 60439, USA

Abstract. Neural-network quantum molecular dynamics (NNQMD) simulations based on machine learning are revolutionizing atomistic simulations of materials by providing quantum-mechanical accuracy but orders-of-magnitude faster, illustrated by ACM Gordon Bell prize (2020) and finalist (2021). State-of-the-art (SOTA) NNQMD model founded on group theory featuring rotational equivariance and local descriptors has provided much higher accuracy and speed than those models, thus named Allegro (meaning fast). On massively parallel supercomputers, however, it suffers a fidelity-scaling problem, where growing number of unphysical predictions of interatomic forces prohibits simulations involving larger numbers of atoms for longer times. Here, we solve this problem by combining the Allegro model with sharpness aware minimization (SAM) for enhancing the robustness of model through improved smoothness of the loss landscape. The resulting Allegro-Legato (meaning fast and "smooth") model was shown to elongate the time-to-failure t_{failure}, without sacrificing computational speed or accuracy. Specifically, Allegro-Legato exhibits much weaker dependence of time-to-failure on the problem size, $t_{\text{failure}} \propto N^{-0.14}$ (N is the number of atoms) compared to the SOTA Allegro model ($t_{\text{failure}} \propto N^{-0.29}$), *i.e.*, systematically delayed time-to-failure, thus allowing much larger and longer NNQMD simulations without failure. The model also exhibits excellent computational scalability and GPU acceleration on the Polaris supercomputer at Argonne Leadership Computing Facility. Such scalable, accurate, fast and robust NNQMD models will likely find broad applications in NNQMD simulations on emerging exaflop/s computers, with a specific example of accounting for nuclear quantum effects in the dynamics of ammonia to lay a foundation of the green ammonia technology for sustainability.

A. Bhatele et al. (Eds.): ISC High Performance 2023, LNCS 13948, pp. 223–239, 2023.
https://doi.org/10.1007/978-3-031-32041-5_12

Keywords: Molecular dynamics · Equivariant neural network · Sharpness-aware minimization

1 Introduction

Neural-network quantum molecular dynamics (NNQMD) simulations based on machine learning are revolutionizing atomistic modeling of materials by following the trajectories of all atoms with quantum-mechanical accuracy at a drastically reduced computational cost [1]. NNQMD not only predicts accurate interatomic forces but also captures quantum properties such as electronic polarization [2] and electronic excitation [3], thus the 'Q' in NNQMD. NNQMD represents one of the most scalable scientific applications on the current high-end supercomputers, evidenced by ACM Gordon Bell prize winner in 2020 [4] and finalist in 2021 [5]. A more recent breakthrough in NNQMD is drastically improved accuracy of force prediction [6] over those previous models, which was achieved through rotationally equivariant neural networks based on a group theoretical formulation of tensor fields [7]. The state-of-the-art (SOTA) accuracy has now been combined with a record speed based on spatially localized descriptors in the latest NNQMD model named Allegro (meaning fast) [8].

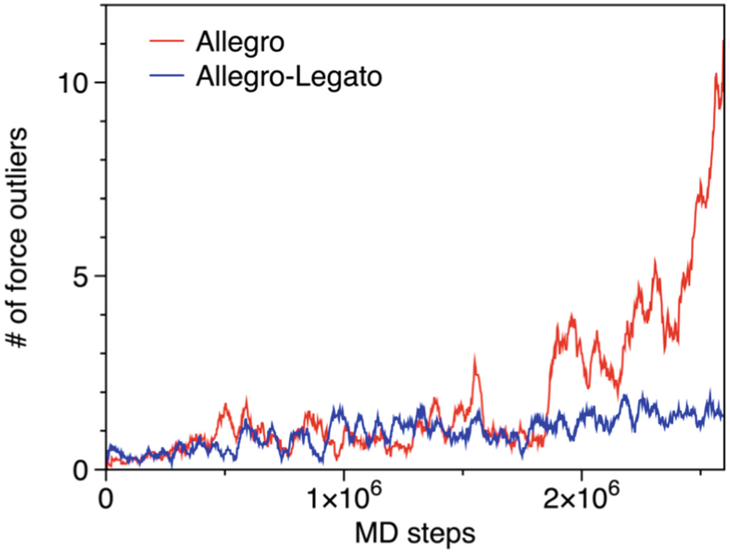

Fig. 1. Number of outliers in atomic force inference during NNQMD simulation: As the simulation progresses, the dynamic of atoms becomes unstable due to an increasing number of unphysically large force values (over 5σ) predicted by the original Allegro model. This resulted in the eventual failure after 2.6×10^6 MD steps (red). On the other hand, the proposed model (Allegro-Legato) maintains a nearly constant number of outliers and the simulation stable (blue). (Color figure online)

Despite its remarkable computational scalability, massively parallel NNQMD simulation faces a major unsolved issue known as *fidelity scaling* [9]. In large-scale NNQMD simulations, small prediction errors can propagate and lead to unphysical atomic forces that degrade the accuracy of atomic trajectory over time. These force outliers can even cause the simulation to terminate unexpectedly (Fig. 1). As simulations become spatially larger and temporarily longer, the number of unphysical force predictions is expected to scale proportionally, which could severely limit the fidelity of NNQMD simulations on new exascale supercomputing platforms, especially for the most exciting far-from-equilibrium applications [3, 10].

In this paper, we solve the fidelity-scaling issue taking a cue from a recent development in machine learning. Solving the fidelity-scaling issue requires robustness of the NNQMD model, *i.e.*, reduced number of unphysical force-prediction outliers when simulation trajectories encounter atomic configurations outside the training dataset. It has been observed that the robustness of a neural-network model can be enhanced by sharpness-aware minimization (SAM) [11]—a training algorithm that regularizes the sharpness of the model (*i.e.*, the curvature of the loss surface) along with its training loss. We thus apply SAM to train the fast Allegro model to smoothen its loss landscape, thereby enhancing its robustness. The resulting Allegro-Legato (meaning fast and "smooth") model is shown to increase the time-to-failure t_{failure}, *i.e.*, how many MD steps a NNQMD simulation can run under microcanonical ensemble, while maintaining the same inference speed and nearly equal accuracy. Specifically, Allegro-Legato exhibits much weaker dependence of time-to-failure on the problem size, $t_{\text{failure}} \propto N^{-0.14}$ (N is the number of atoms) compared to the SOTA Allegro model ($t_{\text{failure}} \propto N^{-0.29}$), thus allowing much larger and longer NNQMD simulations without failure. Along with this main contribution, we find that the fidelity-scalability of the NNQMD model correlates with sharpness of the model more than the number of parameters in the model.[1]

The fast and robust Allegro-Legato model has been implemented in our scalable parallel NNQMD code named RXMD-NN. We have achieved a weak-scaling parallel efficiency of 0.91 on 480 computing nodes, each with an AMD EPYC central processing unit (CPU) and four NVIDIA A100 graphics processing units (GPUs), of the Polaris supercomputer at Argonne Leadership Computing Facility (ALCF). The code has also achieved a 7.6-fold single-node performance acceleration using four GPUs over single 32-core CPU of Polaris.

Allegro-Legato allows much larger spatio-temporal scale NNQMD simulations than are otherwise possible. Unlike MD simulation with heat bath often used in "effective" long-time sampling of molecular configurations (*e.g.*, for protein folding), which disrupts dynamic trajectories, Allegro-Legato enables "true" long-time Hamiltonian dynamics that can be directly compared with fine vibrational modes observed in high-resolution spectroscopic experiments. Specifically, we can now satisfy the prohibitive computational demand of accounting for subtle nuclear quantum effects in the dynamics of ammonia based on path-integral molecular dynamics, which is essential for resolving a mystery in a recent high-resolution neutron-scattering experimental observation at Oak Ridge National Laboratory. Synergy between the most advanced neutron experiment and

[1] Code is available at github.com/ibayashi-hikaru/allegro-legato.

leadership-scale NNQMD simulation lays a foundation of the green ammonia-based fuel technology for achieving a sustainable society.

2 Method Innovation

This section first summarizes (1) NNQMD simulation method, along with the SOTA Allegro model, and (2) SAM for robust neural-network model training. We then present the key method innovation of SAM-enhanced Allegro model, Allegro-Legato, followed by its scalable parallel implementation.

2.1 Summary of Neural-Network Quantum Molecular Dynamics

Molecular dynamics (MD) simulation follows time evolution of the positions $\{\mathbf{r}_i | i = 1, \ldots, N\}$ (*i.e.*, trajectories) of N atoms,

$$m_i \frac{d^2}{dt^2}\mathbf{r}_i = \mathbf{f}_i = -\frac{\partial}{\partial \mathbf{r}_i} E(\{\mathbf{r}_i\}), \tag{1}$$

where m_i and $\mathbf{f}_i$ are the mass of the i-th atoms and the force acting on it, whereas E is the interatomic potential energy that is dictated by quantum mechanics (QM). In NNQMD, neural networks are trained to reproduce ground-truth QM values, $E(\{\mathbf{r}_i\}_t)$, for a set of atomic configurations $\left\{\{\mathbf{r}_i\}_t | t = 1, \ldots, N_{\text{training}}\right\}$ (N_{training} is the number of training configurations) [1–5]. In the SOTA Allegro model, the energy E is composed of pairwise embedding energies, E_{ij}, between atomic pairs (i, j) within a finite cutoff distance to preserve data locality [8]. Key to the high accuracy of Allegro is that all energy terms are group-theoretically equivariant with respect to rotation, inversion and translation, *i.e.*, to the Euclidean group $E(3)$ [6, 7]. This is achieved by representing the energy in terms of tensors up to rank ℓ and tensor products using their irreducible representations. In short, Allegro attains accuracy through group-theoretical equivariance and computational speed through data locality.

2.2 Summary of Sharpness-Aware Minimization

Neural networks are trained by minimizing the loss function $L(\mathbf{w})$ where $\mathbf{w}$ represents the weight parameters of the neural network. Design choice of optimization methods plays a crucial role in machine learning, as it impacts various factors such as convergence speed and generalization performance [12]. In particular, vulnerability to adversarial attacks is a problem unique to neural networks [13], which has actively been studied in various fields such as computer vision [14] and natural language processing [4]. Recent studies suggest that the fidelity-scalability in NNQMD can also be viewed as a robustness against "adversarial attacks" during large-scale simulations [15, 16], where atomic trajectories are "attacked" by the accumulated unphysical predictions, *i.e.*, "adversarial perturbations" throughout the long and large-scale simulation. Therefore, it is natural to expect that optimization methods for adversarial attack would enhance the fidelity-scalability in NNQMD.

Sharpness-aware minimization (SAM) is one of such robustness-enhancing methods proposed in the computer vision area [11]. The key component of SAM is that it minimizes "sharpness" of the model defined as

$$\max_{\|\boldsymbol{\epsilon}\|_2 \le \rho} \{L(\mathbf{w} + \boldsymbol{\epsilon}) - L(\mathbf{w})\}, \tag{2}$$

where ρ (the size of neighborhood) is a hyperparameter to define sharpness. While computing the sharpness directly is infeasible, it has been shown that minimizing $L(\mathbf{w}) + \max_{\|\boldsymbol{\epsilon}\|_2 \le \rho}\{L(\mathbf{w} + \boldsymbol{\epsilon}) - L(\mathbf{w})\}$ (training loss + sharpness) can be achieved through the following update rule:

$$\boldsymbol{w} = \boldsymbol{w} - \eta \nabla_{\boldsymbol{w}'} L\big(\boldsymbol{w}'\big)\Big|_{\boldsymbol{w}' = \boldsymbol{w} + \rho \frac{\nabla_{\boldsymbol{w}} L(\boldsymbol{w})}{\|\nabla_{\boldsymbol{w}} L(\boldsymbol{w})\|}} \; (\eta : \text{learning rate}), \tag{3}$$

which utilizes first-order derivatives, *i.e.*, $\nabla_{\mathbf{w}} L(\mathbf{w})$. This allows for the optimization of sharpness without the need for computationally expensive second-order derivatives.

2.3 Key Innovation: Allegro-Legato: SAM-Enhanced Allegro

As explained above, our hypothesis is that smoothened loss landscape through SAM enhances fidelity scaling of NNQMD. To quantitatively test this hypothesis, we incorporate SAM into the training of the Allegro NNQMD model [8], which entails SOTA accuracy and computational speed. We call the resulting SAM-enhanced Allegro model as Allegro-Legato (In music, Legato means "smooth without sudden breaking between notes").

To find an appropriate strength of sharpness regularization, SAM's hyper parameter ρ is tuned so as to provide the most robust model, *i.e.*, the longest time-to-failure, t_{failure}, in a small-scale simulation ($N = 432$). Table 1 shows the result of our grid search over $\rho \in \{0.001, 0.0025, 0.005, 0.01, 0.025, 0.05\}$, from which we found that $\rho = 0.005$ gives the longest t_{failure} in our setup. For the small-scale simulation test, we used LAMMPS, which is a widely used open-source MD simulation software (https://www.lammps.org). See Sect. 4.5 for the detailed training settings.

Table 1. SAM strength ρ *vs.* time-to-failure t_{failure}: We tune ρ by conducting a grid search in the range of 0.001 to 0.05. A model with $\rho = 0.005$ gives the largest t_{failure} with a small-scale simulation ($N = 432$).

ρ	0.001	0.0025	**0.005**	0.01	0.025	0.05
t_{failure}	4030	6420	**8480**	4760	4210	3780

2.4 RXMD-NN: Scalable Parallel Implementation of Allegro-Legato NNQMD

For large-scale testing of computational and fidelity scaling, we implement the proposed Allegro-Legato NNQMD model in our RXMD-NN software [3, 9], which is an extension of our scalable parallel reactive MD software, RXMD [17]. RXMD-NN employs a hierarchical divide-and-conquer scheme to realize "globally-scalable and local-fast" (or "globally-sparse and locally-dense") parallelization [18]: (1) globally scalable spatial decomposition that is best suited for massively parallel computing platforms; and (2) locally efficient linked-list decomposition and subsequent neighbor-list construction to achieve the $O(N)$ computational complexity. Interprocess communication is implemented using non-blocking application programming interfaces (APIs) of Message Passing Interface (MPI) library, and the communication pattern is designed to be lock-free with minimal internode-data exchange. While it is one of the most widely adapted strategies in large-scale MD applications, this is particularly suitable for NNQMD algorithm to take advantage of the modern high-performance computing (HPC) architecture, in which a few very powerful GPU cards do the heavy lifting by accelerating computationally demanding kernels while random memory access and out-of-order data processing are concurrently executed by many-core CPUs. In RXMD-NN, CPU is responsible for the adjacency-list construction in parallel. The constructed adjacency list, together with atom position and type information, is converted to PyTorch tensor object for force inference on GPUs. RXMD-NN allows to control the computational granularity, such as the number of atoms per domain and domains per node, to find an ideal balance between horizontal and vertical scalability to utilize available hardware resources.

PyTorch has become a standard Python library in machine learning community due to its APIs for complex model architectures that enables highly efficient training and inference on GPU. However, production platforms such as HPC clusters, mobile devices, and edge nodes often demand a set of requirements that Python is not designed for, *e.g.*, multithreading, low latency computing, and massively parallel distributed architectures. GPU Offloading of Allegro model is realized by TorchScript, which is statically typed intermediate representation to create serialized and optimizable ML models. The serialized model can be loaded from other programming language such as C++ allowing to be deployed in environments that are difficult for python codes to run without sacrificing multithreading and optimization opportunities.

3 Results

We test both fidelity and computational scalability of the proposed Allegro-Legato NNQMD model as implemented in the RXMD-NN code on a leadership-scale computing platform, Polaris, at Argonne Leadership Computing Facility (ALCF).

3.1 Experimental Platform

We conduct numerical experiments on the Polaris supercomputer at ALCF. It is a Hewlett Packard Enterprise (HPE) Apollo 6500 Gen 10+ based system consisting of two computing nodes per chassis, seven chassis per rack, and 40 racks, with a total of 560 nodes. Each Polaris node has one 2.8 GHz AMD EPYC Milan 7543P 32-core CPU with 512 GB of DDR4 RAM, four NVIDIA A100 GPUs with 40GB HBM2 memory per GPU, two 1.6 TB of SSDs in RAID0 and two Slingshot network endpoints. Polaris uses the NVIDIA A100 HGX platform to connect all 4 GPUs *via* NVLink, with a GPU interconnect bandwidth of 600 GB/s. Designed by Cray, the Slingshot interconnect is based on high radix 64-port switches arranged in dragonfly topology, offering adaptive routing, congestion control and bandwidth guarantees by assigning traffic classes to applications. Polaris is rated at a production peak performance of 44 petaflops with node-wise performance at 78 teraflops for double precision.

3.2 Fidelity-Scaling Results

For the fidelity-scaling test, we trained Allegro and Allegro-Legato with $\ell = 1$ and examined their robustness in terms of t_{failure}, *i.e.* the greater t_{failure}, the more robust. The parameters of MD simulation for the test are carefully chosen so that each MD simulation is expected to fail within a reasonable time but not immediately. While the constant-temperature ensemble method based on Nose-Hoover thermostat (*i.e.*, NVT ensemble) is used to study thermal-equilibrium properties, it could suppress and hidden unphysical model predictions by connecting atoms with an external thermostat. Microcanonical ensemble (NVE) method is the most rigorous test on the model robustness by simply integrating the equations of motion without an external control (also it has broader applicability to nonequilibrium processes). In each simulation instance, the liquid ammonia system is first thermalized at a temperature of 200 K using NVT ensemble for 1,000 steps. We subsequently switch the ensemble to NVE and continue the simulation until it fails to determine t_{failure} (see the arrow in Fig. 1). The time step Δt of 2 femto-seconds (fs) is chosen throughout the robustness test. For each system size, over ten independent simulation instances are averaged to measure t_{failure}.

Figure 2 shows t_{failure} as a function of the system size (*i.e.*, the total number of atoms, N) ranging from $N = 432$ to 27,648. Firstly, regardless of the system size, we observe a significant improvement in the averaged sustained MD simulation steps using Allegro-Legato model. We observe the greatest improvement of the simulation robustness in the largest system with $N = 27{,}648$, where 2.6-times longer MD simulation (14,600 steps) is achieved with SAM in Allegro-Legato than that with the original Allegro model (5,500 steps). In the MD simulation framework, even a single misprediction of atomic force can lead to catastrophe through chain reactions. An error in atomic force is integrated into its velocity, then into atom coordinates. Too large atomic displacement in a single MD step could result in unphysically strong collisions with other atoms, which propagate throughout the system within a few MD steps, known as "the billiard effect." MD simulations with large number of atoms or longer simulation time will inevitably suffer from higher probability of having such model mispredictions, thus fail faster than a smaller system. Our test demonstrates that SAM successfully improves the robustness of model prediction, realizing a stable MD simulation for greater time steps.

To quantify fidelity scaling, we define a fidelity-scaling exponent $N^{-\beta}$ through the scaling relation,

$$t_{\text{failure}} = \alpha N^{-\beta}, \tag{4}$$

where α is a prefactor. A smaller β value (*i.e.*, weaker fidelity scaling) indicates delayed time-to-failure, thus a capability to study larger spatiotemporal-scale processes accurately on massively parallel computers. The Allegro-Legato model has drastically improved fidelity scaling, $\beta_{\text{Allegro-Legato}} = 0.14 < \beta_{\text{Allegro}} = 0.29$ beyond statistical uncertainty (see the error bars in Fig. 2), thus systematically delaying time-to-failure.

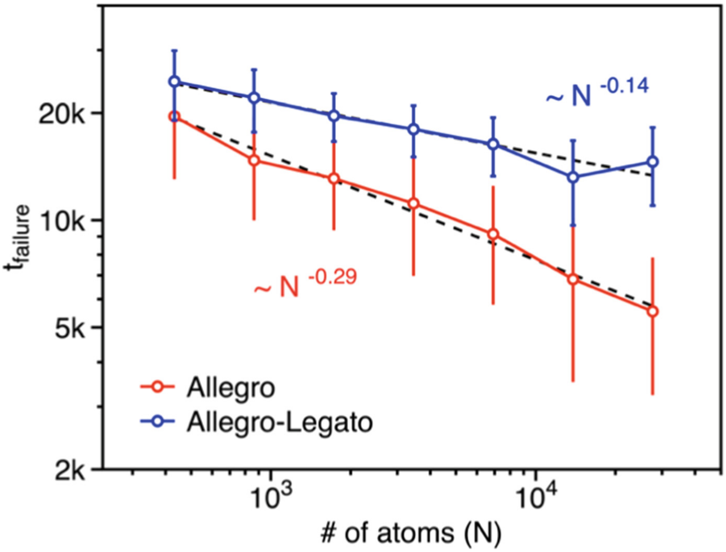

Fig. 2. Fidelity scaling of NNQMD simulation: Here, t_{failure} is measured using NVE ensemble with a timestep of 2 fs. Statistically improved t_{failure} is observed in even the smallest system size, which is further pronounced as the system size increases. The exponent of power law fitting shows nearly a factor of two reduction using Allegro-Legato model.

3.3 Computational-Scaling Results

We measure the wall-clock time per MD step with scaled workload—6,912P-atom ammonia system on P MD domains. In this test, each MD domain consists of 6,912 atoms that are offloaded to single GPU. In addition to the force inference, the execution time includes the adjacency list construction, data transfer between host and GPU memory, and internode communication via network fabric. Figure 3 shows wall-clock time as a function of P. By scaling the problem size linearly with the number of GPUs, the runtime increases only slightly, indicating an excellent scalability.

Here, we quantify the parallel efficiency by defining the speed of NNQMD algorithm as the product of the total number of atoms multiplied by the number of MD steps executed per second. The isogranular speedup is given by the speed on P MD domains relative to the speed of single domain as baseline. The parallel efficiency of weak scalability thus is obtained by the isogranular speedup divided by P. With the granularity of 6,912 atoms per domain, we have obtained an excellent weak-scaling efficiency, 0.91 for up to 13,271,040 atoms on 1,920 A100 GPUs. Despite the relatively large granularity of 6,912 atoms per domain, we obtained a fast time-to-solution of 3.46 s per MD step enabling 25,000 MD steps per day for production runs.

Figure 4 shows GPU acceleration of NNQMD algorithm on single Polaris node. The histogram presents the reduction in wall-clock time per MD step (averaged over 10 MD steps) using the runtime obtained with CPU only (32 cores with 32 threads) as baseline. Here, we examined: (1) three system sizes of N = 1,728, 6,912, and 13,824 ammonia

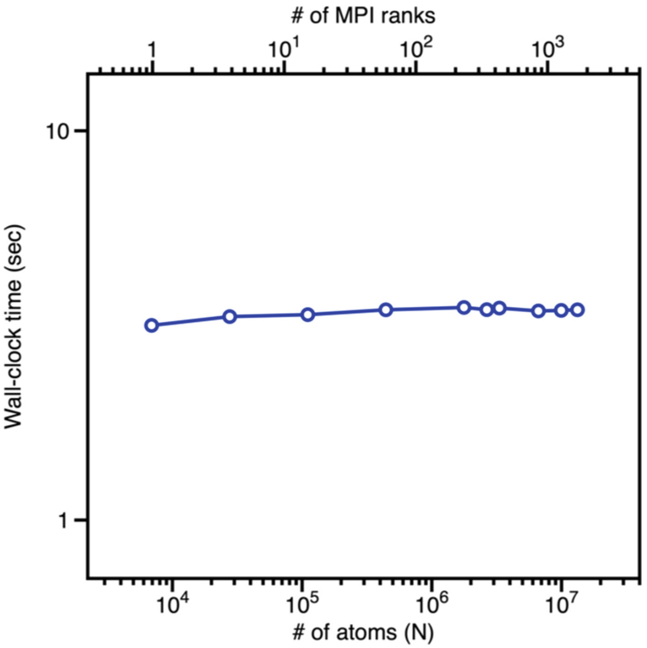

Fig. 3. Wall-clock time of the RXMD-NN code per MD step, with scaled workloads—6,912*P* atom ammonia liquid using *P* A100 GPUs (P = 1,…, 1,920).

atoms; and (2) three domain decompositions such as single, double and quadruple subdomains. Atoms in each domain are assigned to one GPU. With N = 1,728 system, we observe a marginal GPU acceleration up to 1.24× speedup, which has been substantially improved with greater system sizes. We have achieved a 7.6x speedup using N = 13,824 atom system with four subdomains.

4 Discussions

While SAM-enhanced Allegro model, Allegro-Legato, has achieved improved robustness over the SOTA Allegro model as shown in the previous section, we here discuss the imprecation of SAM training to other aspects such as accuracy and computational speed.

4.1 Simulation Time

First of all, MD simulation time is not affected by SAM since SAM only applies to the training stage but not the inference stage in MD simulation. Table 2 compares the simulation time per MD time step for the baseline Allegro model and the proposed Allegro-Legato model. Hereafter, we use the default value, $\ell = 1$, for the maximum tensor rank, thus the same number of parameters for the two models. The simulation time is identical for both models within the measurement uncertainty due to nondedicated access to the experimental platform.

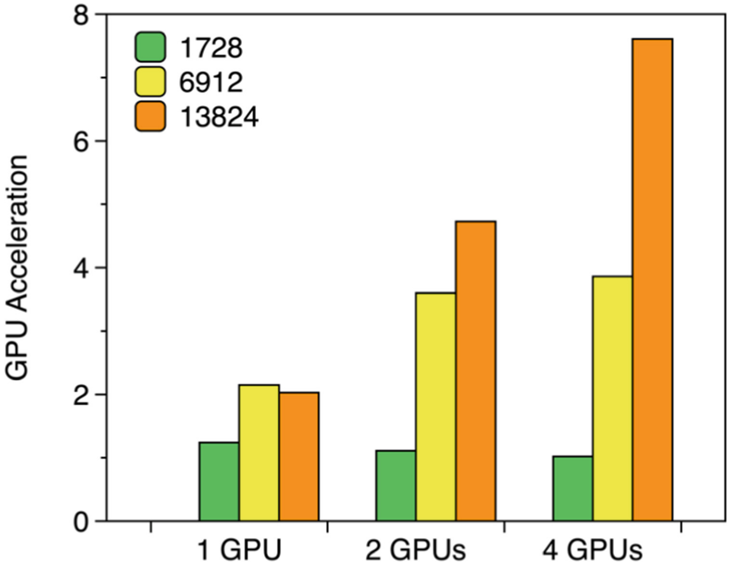

Fig. 4. GPU acceleration of NNQMD algorithm: Three system sizes of N = 1728, 6912 and 13,824 atoms are examined. The histogram presents the reduction in wall-clock time per MD step over the runtime with 32 CPU cores without GPU as reference. Detail of the benchmark platform as well as the GPU and CPU architectures are presented in the main text. We have achieved 7.6× speedup using four GPUs with N = 13,824 atoms.

As a comparison, Table 2 also shows the baseline Allegro model with two other tensor ranks, $\ell = 0$ and 2. Larger ℓ generates more accurate but larger models (*i.e.*, larger numbers of parameters) and hence incur longer simulation times. Based on the accuracy/computational-cost trade-off, production NNQMD simulations with the Allegro model typically use $\ell = 1$.

Table 2. Simulation-time comparison: As SAM only applies to the training stage and does not modify the size of architecture, the computational cost for simulation is not affected.

Model	# of parameters	Time/step (ms)
Allegro	133,544	916
Allegro-Legato	133,544	898
Reference Models		
Allegro ($\ell = 0$)	95,656	395
Allegro ($\ell = 2$)	183,720	2,580

4.2 Training Time

As mentioned in Sect. 2.2, SAM's shortcoming is that it requires more computation time than the base optimizer, because each epoch has to compute the first-order gradients twice. However, in our setting, SAM converges faster than the default optimizer, and thus the total training time is not significantly affected (Table 3). As references, we also measured the training time of Allegro models with different maximum tensor ranks, $\ell = 0$ and 2 and we observed that the training cost increases drastically for larger ℓ. In summary, Allegro-Legato improves the robustness of Allegro without incurring extra training cost.

Table 3. Training-time comparison: Although SAM takes longer per-epoch training time, it converges faster and thus does not significantly affect total training time. Compared to the reference training times of variations of Allegro models, the extra training cost of Allegro-Legato is negligible.

Model	Total time (hours)	Per-epoch time (seconds)	Epochs
Allegro	11.1	248	161
Allegro-Legato	13.6	433	113
Reference Models			
Allegro ($\ell = 0$)	4.4	127	127
Allegro ($\ell = 2$)	19.6	636	111

4.3 Model Accuracy

While faithful reproduction of system energy is necessary to properly guide model training, the most crucial to MD simulations is accurate force prediction. We obtained the validation error in atomic force as 15.9 (root mean-square error, RMSE) and 11.6 (mean absolute error, MAE) with Allegro-Legato ($\ell = 1$) model, and 14.7 (RMSE) and 10.7 (MAE) with the original Allegro model ($\ell = 1$), respectively. All error values are in a unit of meV/Å. Chmiela *et al.* recently provided a guideline that MAE required for reliable MD simulations is 1 kcal/mol/Å, which corresponds to 43.4 meV/Å [19]. Although Allegro-Legato incurs a slight increase in the force prediction error (about 8% in the liquid ammonia dataset) compared to the original Allegro model, the obtained force error is about a factor four smaller than the guideline for reliably performing MD simulations. Namely, Allegro-Legato improves the robustness without sacrificing accuracy.

4.4 Implicit Sharpness Regularization in Allegro

While we propose to explicitly control the sharpness of models, we found that one control parameter in the baseline Allegro model (*i.e.*, maximum rank of tensors to represent features) implicitly regulate the sharpness of the model. In Table 4, besides our Allegro-Legato model having smaller sharpness, Allegro $\ell = 1, 2$ models have significantly smaller sharpness and higher t_{failure} compared to Allegro $\ell = 0$ model. Namely, Allegro with higher ℓ implicitly regularizes sharpness, resulting in higher robustness (*i.e.*, larger t_{failure}), but with increasing computational cost. Allegro-Legato ($\ell = 1$) model achieves the same level of sharpness as Allegro ($\ell = 2$) model with much less computing time; see Tables 2 and 3.

Table 4. Implicit sharpness regularization by Allegro: While our Allegro-Legato model has smaller sharpness than Allegro, Allegro models with larger ℓ have progressively smaller sharpness. Here, we measure sharpness, $\underset{\|\boldsymbol{\epsilon}\|_2 \leq \rho}{max} \{L(\boldsymbol{w} + \boldsymbol{\epsilon}) - L(\boldsymbol{w})\}$, by taking maximum of 1,000 independent random samples around the 0.05-neighborhood of each minimum.

Model	Allegro ($\ell = 0$)	Allegro ($\ell = 1$)	Allegro ($\ell = 2$)	Allegro-Legato ($\ell = 1$)
Sharpness	5.0×10^{-4}	3.2×10^{-4}	9.8×10^{-5}	1.2×10^{-4}

Figure 5 visualizes the loss surface of Allegro ($\ell = 0$, 1, and 2) and Allegro-Legato ($\ell = 1$) models. The figure confirms: (1) progressive smoothening (*i.e.*, smaller sharpness) for larger ℓ within the Allegro model due to implicit regularization through accuracy but with increasing computational cost; and (2) explicit smoothening of Allegro-Legato through SAM over Allegro with the same ℓ without extra computational cost.

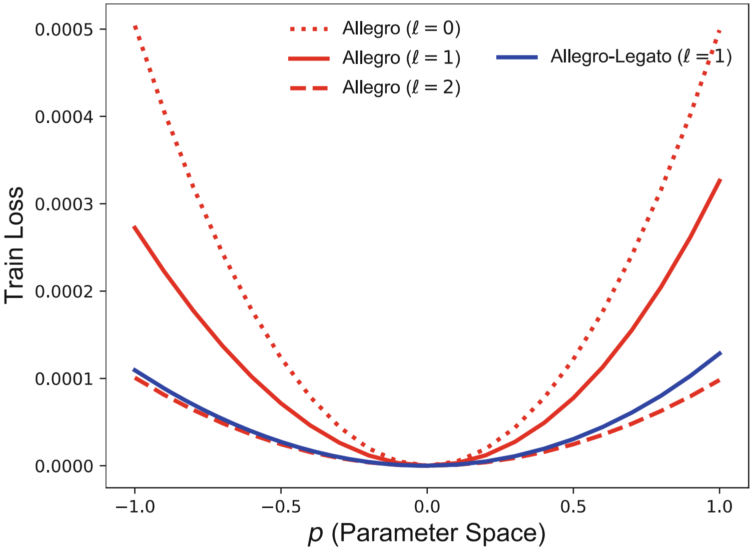

Fig. 5: Loss surface visualization: One dimensional visualization of loss surface of each model. Following the definition of sharpness (Eq. 2), we randomly sample a vector, $\boldsymbol{d}$, that gives the sharpness direction to compute $L(\boldsymbol{w} + p\boldsymbol{d})$ for $p \in [-1, 1]$.

4.5 Training Details

Lastly, we provide detailed training configuration for completeness (Table 5). For fair comparison, we used the default hyperparameters that are released as the SOTA model and SAM training uses the default optimizer as its base optimizer.

Table 5. Detailed training setting: All training setups in this paper adopt these parameters unless otherwise noted.

Material type	Liquid NH_3
Number of atoms per a configuration	432
# of training examples ($N_{training}$)	4,500
# of validation examples	500
$r_{\max}$ for cutoff	6.0
Maximum tensor rank (ℓ)	1
Batch size	4
Peak learning rate	2e−3
Learning rate decay	ReduceLROnPlateau
Learning rate scheduler patience	50
Learning rate scheduler factor	0.5
(Base) Optimizer	Adam
Adam's (β_1, β_2)	(0.9, 0.999)
Loss function	Per atom MSE
Loss coefficient (force, total energy)	(1.0, 1.0)
Stopping criterion	$\Delta L_{\text{validation}} \leq 3e - 3$ for 100 epochs

5 Applications

The improved robustness of the proposed Allegro-Legato model, while preserving the SOTA accuracy and computational speed of Allegro, enables large spatio-temporal scale NNQMD simulations on leadership-scale computers. A compelling example is the study of vibrational properties of ammonia. Development of dynamical models that accurately reproduce the vibrational spectra of molecular crystals and liquids is vital for predictions of their thermodynamic behavior, which is critical for their applications in energy, biological, and pharmaceutical systems [20]. In particular, there has been growing development of green ammonia-based fuel technologies for sustainable society over the past few years. Ammonia (NH_3) has a higher energy density than even liquid hydrogen, but ammonia can be stored at a much less energy-intensive –33 °C versus –253 °C, and thanks to a century of ammonia use in agriculture, a vast ammonia infrastructure already exists [21]. Over 180 million metric tons of ammonia is produced annually, and 120 ports are equipped with ammonia terminals [21]. Development of technologies based on ammonia will be reliant on our ability to understand and model the complex physical and chemical interactions that give rise to its unique properties.

There are multiple complicating factors that require careful considerations such as nuclear quantum effects (NQEs) and its coupling with vibrational anharmonicity when developing computational frameworks that accurately describe vibrational properties [20]. Standard first-principles calculations for vibrational properties only treat electrons quantum mechanically and vibrational properties can be determined by Fourier transform and matrix diagonalization of the unit-cell Hessian, which is at most on the order of a few 100 entries [22]. Evaluating the role of NQEs and its coupling with vibrational anharmonicity is done in the so-called path integral MD (PIMD) approach, which samples the quantum partition function for the entire quantum system [23, 24]. This requires long-time simulations of a large number of replicas of large MD systems that are harmonically coupled to each other as interacting ring-polymers, especially at low temperatures [23, 24]. The background of Fig. 6a shows a typical first principles-based simulation, where the atoms are treated classically and the electron charge density is treated quantum-mechanically to compute atomic forces, which is illustrated as blue iso-surfaces. In the foreground we have highlighted one NH_3 molecule from a PIMD simulation of the same atomic configuration, where each atom has 32 replicas that are harmonically coupled together. The computation of the replica simulations is embarrassingly parallel, with only fixed nearest replica communication, and the major cost is computing the energy and forces for the atoms within each replica simulation, which is typically done from first principles. However, our Allegro-Legato model with enhanced robustness allows for stable long-time MD simulations at near quantum accuracy, and thus can replace expensive first-principles calculations in the PIMD simulations, which would make accurate evaluation of ammonia's low energy inter-molecular vibrational modes intractable.

We have performed massively parallel PIMD simulations with our Allegro-Legato model, computing the energy and forces within each replica simulation to evaluate the phonon spectra for inter-molecular modes of ammonia. The Allegro-Legato model is found to produce the expected softening of high-energy modes at finite temperature with inclusion of nuclear quantum effects in comparison to standard matrix diagonalization within the harmonic approximation, which is illustrated in Fig. 6b. In particular, reduction of the energy of the vibrational modes in the 30–90 meV is consistent with high-end neutron experiments for the vibrational spectrum performed by the authors at Oak Ridge National Laboratory in the last summer (these results will be published elsewhere).

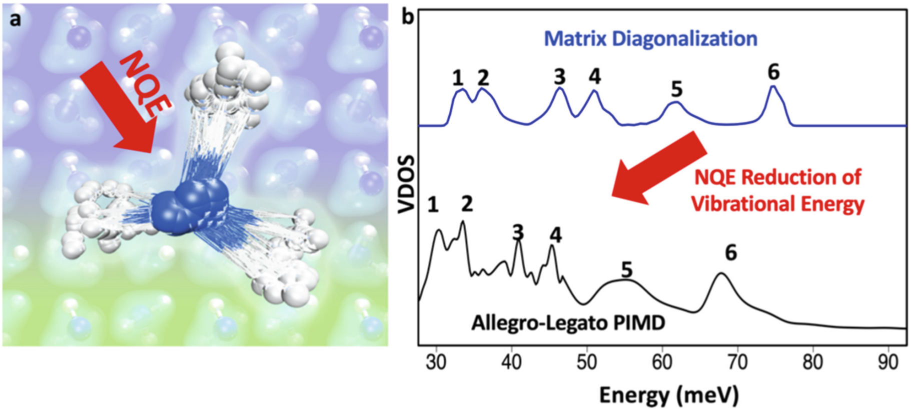

Fig. 6. Computed vibrational spectra of ammonia: (a) While typical first-principles simulation treats atoms classically and electrons quantum-mechanically, PIMD simulation uses multiple replicas of each atom to mimic nuclear quantum effect (NQE). (b) Top curve shows vibrational spectrum computed at zero temperature without NQE, while bottom at finite temperature with Allegro-Legato PIMD simulation. With the inclusion of NQE, Allegro-Legato PIMD correctly shows softening of high-energy inter-molecular modes expected at finite temperature and explains high-end neutron-scattering observations.

6 Related Work

There has been an explosion in the development and application of NNQMD simulations [1–3, 6, 8] and their scalable parallel implementation [4, 5]. On the other hand, it was only recently that the robustness of NNQMD was quantified in terms of time-to-failure t_{failure} [25] and its deteriorating reduction with the problem size (*i.e.*, fidelity-scaling problem) was pointed out [9]. This work is the first to: (1) formally quantify the fidelity scaling by introducing the fidelity-scaling exponent β through $t_{\text{failure}} \propto N^{-\beta}$ (N is the number of atoms); and (2) propose the solution to the fidelity-scaling problem using sharpness-aware minimization.

Robustness against adversarial attacks is a central and widely studied issue in machine learning [4, 13, 14]. Compared to typical adversarial attacks, it is nontrivial to generate adversarial perturbations for NNQMD. This is because the attack we consider is not only focused on the accuracy of the model, but also on the time to failure (t_{failure}) of the model, which can only be determined through long-time simulations [15, 16]. Generative adversarial network (GAN) is one possible approach for sampling molecular configurations in a learning-on-the-fly setting [26]. However, we remark that the real strength of MD simulation is its ability to compute dynamic correlations that can directly explain high-resolution spectroscopic experiments, which requires a long uninterrupted Hamiltonian trajectory, to which adversarial networks are generally not applicable. In this domain, Allegro-Legato thus provides a unique solution.

7 Conclusion

We have introduced the proposed SAM-based solution to the fidelity-scaling problem into the Allegro NNQMD model,[8] which represents the state-of-the-art accuracy and speed. The resulting Allegro-Legato model has drastically improved fidelity scaling by exhibiting a significantly lower exponent, $\beta_{\text{Allegro}-\text{Legato}} = 0.14 < \beta_{\text{Allegro}} = 0.29$, thus systematically delaying time-to-failure. Such improved fidelity scaling is central to ensure that meaningful scientific knowledge is extracted from large-scale simulations on leadership-scale parallel computers. Our scalable parallel implementation of Allegro-Legato with excellent computational scaling and GPU acceleration combines accuracy, speed, robustness and scalability, thus allowing practical large spatiotemporal-scale NNQMD simulations for challenging applications on exascale computing platforms.

Acknowledgement. This work was supported as part of the Computational Materials Sciences Program funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, under award number DE-SC0014607. H.I. and K.N. were partially supported by an NSF grant, OAC-2118061. The simulations were performed at the Argonne Leadership Computing Facility under the DOE INCITE program, while scalable code development was supported by the Aurora ESP program. The authors acknowledge the Center for Advanced Research Computing at the University of Southern California for providing computing resources that have contributed to the research results reported within this publication. We are grateful to Dr. Makiko Hirata for valuable discussions regarding Allegro-Legato.

References

1. Behler, J.: Constructing high-dimensional neural network potentials: a tutorial review. Int. J. Quantum Chem. **115**(16), 1032–1050 (2015)
2. Krishnamoorthy, A., et al.: Dielectric constant of liquid water determined with neural network quantum molecular dynamics. Phys. Rev. Lett. **126**(21), 216403 (2021)
3. Linker, T., et al.: Exploring far-from-equilibrium ultrafast polarization control in ferroelectric oxides with excited-state neural network quantum molecular dynamics. Sci. Adv. **8**(12), eabk2625 (2022)
4. Jia, W., et al.: Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning. Proceedings of Supercomputing, vol. 5. ACM/IEEE (2020)
5. Nguyen-Cong, K., et al.: Billion atom molecular dynamics simulations of carbon at extreme conditions and experimental time and length scales. In: Proceedings of Supercomputing, vol. 4. IEEE/ACM (2021)
6. Batzner, S., et al.: E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. Nat. Commun. **13**, 2453 (2021)
7. Thomas, N., et al.: Tensor field networks: rotation-and translation-equivariant neural networks for 3D point clouds. arXiv:1802.08219 (2018)
8. Musaelian, A., et al.: Learning local equivariant representations for large-scale atomistic dynamics. arXiv:2204.05249 (2022)
9. Rajak, P., et al.: Ex-NNQMD: extreme-scale neural network quantum molecular dynamics. In: Proceedings of IPDPSW21, pp. 943–946. IEEE (2021)
10. Misawa, M., et al.: Application of first-principles-based artificial neural network potentials to multiscale-shock dynamics simulations on solid materials. J. Phys. Chem. Lett. **11**, 4536–4541 (2020)

11. Foret, P., Kleiner, A., Mobahi, H., Neyshabur, B.: Sharpness-aware minimization for efficiently improving generalization. In: Proceedings of International Conference on Learning Representations, ICLR, vol. 1839 (2021)
12. Schmidt, R.M., Schneider, F., Hennig, P.: Descending through a crowded valley - benchmarking deep learning optimizers. In: Proceedings of International Conference on Machine Learning, ICML, vol. 139, pp. 9367–9376 (2021)
13. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: Proceedings of International Conference on Learning Representations, ICLR (2015)
14. Shafahi, Aet al.: Adversarial training for free! In: Proceedings of NeurIPS, vol. 32 (2019)
15. Schwalbe-Koda, D., Tan, A.R., Gómez-Bombarelli, R.: Differentiable sampling of molecular geometries with uncertainty-based adversarial attacks. Nat. Commun. **12**(1), 5104 (2021)
16. Cubuk, E.D., Schoenholz, S.S.: Adversarial forces of physical models. In: Proceedings of NeurIPS-MLPS (2020)
17. Nomura, K., Kalia, R.K., Nakano, A., Rajak, P., Vashishta, P.: RXMD: a scalable reactive molecular dynamics simulator for optimized time-to-solution. SoftwareX **11**, 100389 (2020)
18. Nomura, K., et al.: Metascalable quantum molecular dynamics simulations of hydrogen-on-demand. In: Proceedings of SC14, pp. 661–673. IEEE/ACM (2014)
19. Chmiela, S., Sauceda, H.E., Müller, K.-R., Tkatchenko, A.: Towards exact molecular dynamics simulations with machine-learned force fields. Nat. Commun. **9**(1), 3887 (2018)
20. Hoja, J., Reilly, A.M., Tkatchenko, A.: First-principles modeling of molecular crystals: structures and stabilities, temperature and pressure. WIREs Comput. Mol. Sci. **7**(1), e1294 (2017)
21. Chehade, G., Dincer, I.: Progress in green ammonia production as potential carbon-free fuel. Fuel **299**, 120845 (2021)
22. Togo, A., Tanaka, I.: First principles phonon calculations in materials science. Scripta Mater. **108**, 1–5 (2015)
23. Feynman, R.P., Hibbs, A.R.: Quantum Mechanics and Path Integrals. McGraw-Hill (1965)
24. Rossi, M., Ceriotti, M., Manolopoulos, D.E.: How to remove the spurious resonances from ring polymer molecular dynamics. J. Chem. Phys. **140**(23), 234116 (2014)
25. Fu, X., et al.: Forces are not enough: benchmark and critical evaluation for machine learning force fields with molecular simulations. arXiv:2210.07237 (2022)
26. Jinnouchi, R., Miwa, K., Karsai, F., Kresse, G., Asahi, R.: On-the-fly active learning of interatomic potentials for large-scale atomistic simulations. J. Phys. Chem. Lett. **11**(17), 6946–6955 (2020)